

Fall 12-1-2004

A Centralized Storage System for Catering Information

Matthew D. Johnson
Dakota State University

Follow this and additional works at: <https://scholar.dsu.edu/theses>

Recommended Citation

Johnson, Matthew D., "A Centralized Storage System for Catering Information" (2004). *Masters Theses*. 62.
<https://scholar.dsu.edu/theses/62>

This Thesis is brought to you for free and open access by Beadle Scholar. It has been accepted for inclusion in Masters Theses by an authorized administrator of Beadle Scholar. For more information, please contact repository@dsu.edu.

A Centralized Storage System for Catering Information

by
Matthew D. Johnson

A project submitted in partial fulfillment of the requirements for the

Master of Science in Information Systems

Dakota State University

2004



MSIS
PROJECT APPROVAL FORM

Student Name: Matthew David Johnson

Expected Graduation Date: December 19, 2004

Master's Project Title: A Centralized Storage System for Catering Information

Date Project Plan Approved: _____

Date Project Coordinator Notified and Grade Submitted: _____

Approvals/Signatures:

Student: Matthew David Johnson

Date: 12-21-04

Faculty supervisor: Mr. Shan

Date: 12-29-04

Committee member: Shari Zuber

Date: 12-29-04

Committee member: Steve [unclear]

Date: 12/30/04

*To Ensure Certification of Completion:
Student must bring or send the original to the Graduate Programs Office.
Copies on acid free paper go to the library with the reports for binding.*

Original to Graduate Programs Office
Acid-free copies with written reports to library
Copies to: Project supervisor and committee and to MSIS Coordinator

Project Abstract:

A Centralized Storage System for Catering Information by Matthew D. Johnson

**Chairperson of the Supervisory Committee: Professor Ronghua Shan
Department of Information Systems**

Hy-Vee has always prided itself at being the best around in customer service. It has been the cornerstone of our company for the last 75 years and guides the way that we conduct every part of our business.

With the incredible expansion of our catering business over the last decade, it became evident that we were not providing the level of service that our clients had come to expect of us. This lack of service came in several ways but all were traced back to a simple lack of adequate recordkeeping.

A review of the company's method for storing catering information showed us that the two methods most widely in use were both error prone and required much more labor than was really reasonable to expend on it.

After carefully reviewing the needs of the company and the desires of the end users, it was decided that the best possible route that we could take to rectify this situation was to develop a new computerized storage system that could help us improve our customer service level while eliminating extra work for our employees and mountains of paperwork piled up in our stores.

Acknowledgments

I would like to extend my thanks to Doctor Ronghua Shan for his patience and guidance during the development and completion of this project. His help went well above the call of duty on several occasions and without his support this project would have never reached completion on time. I would also like to extend thanks to the remaining members of my committee, Dr. Zehai Zhou and Professor Stephen Krebsbach for their input into the final project.

My sincere thanks go out to my coworkers and supervisors at Hy-Vee. You have put up with a seemingly endless stream of questions, surveys, test models, and training. Without all of the support that you gave to me and the input that you were able to bring to the table, this project would only be a passing thought in the back of my mind. You helped bring it to life.

To my family and friends I can only say "Thank You". You have always been there supporting me in my schooling and giving me that extra boost when needed. Your support has meant so much to me over the last few years and I may never be able to show you how much.

And finally to my wonderful wife, I can say nothing but "Thank You" and "I Love You". Your patience and understanding, love and support over the last few years has truly been outstanding. You may never understand the huge help you have been to me when deadlines loomed or frustration over problems took the best of me. You never let me forget that I could do this and that you would be with me on the journey every step of the way. Thank you from the bottom of my heart.

Table of Contents:

Title Page	i
MSIS Project Approval Page	ii
Abstract	iii
Acknowledgments	iv
Table of Contents	v
List of Tables	vii
List of Figures	viii
1. Introduction	1
2. Statement of Problem	3
2.1 Review of Current Situation	3
2.1.1 Memory Method	3
2.1.2 Order Form Method	3
2.2 Review of Existing Industry Solutions	4
3. Objectives	6
3.1 Deliverables	6
3.1.1 A Completed set of Databases.....	7
3.1.2 A Working Program	7
3.1.3 A Written Report.....	7
4. Scope of Action	8
4.1 Review Existing Systems	8
4.2 Review of Available Commercial Alternatives	9
4.3 System Design Standards Establishment	10
4.4 Database Design and Construction.....	11

4.4.1 Database Design	11
4.4.2 Database Construction	14
4.5 Program Design and Construction	15
4.5.1 Program Design	15
4.5.1.1 Main Screen Design	16
4.5.1.2 Add Customer Screen Design	17
4.5.1.3 Update Customer Screen Design	18
4.5.1.4 Search Customer Screen Design	19
4.5.1.5 New Event Screen Design	20
4.5.1.6 Event Update Screen Design	21
4.5.1.7 Search Event Screen Design	22
4.5.2 Program Construction	23
5. Results and Conclusions	26
Appendices	
A – Work Breakdown Structure	A-1
B – Gantt Chart	B-1
C – About Form Code	C-1
D – Add Event Form Code	D-1
E – Add Customer Form Code	E-1
F – Cater Information Form Code	F-1
G – Customer Information Form Code	G-1
H – Event Information Form Code	H-1
I – Search Customer Form Code	I-1
J – Search Event Form Code	J-1
K – Design Limitations Sheet	K-1
L – End User Survey	L-1
M – End User Survey Results	M-1
N – Existing Company Program Screen Shots	N-1

List of Tables

Table 1: Client Database Design..... 12

Table 2: Event Database Design 13

List of Figures

Figure 1: Database Relationship Diagram.....	14
Figure 2: Main Startup Screen	17
Figure 3: Add Customer Screen.....	18
Figure 4: Update Customer Information Screen.....	19
Figure 5: Customer Search Screen.....	20
Figure 6: New Event Screen	21
Figure 7: Update Event Screen	22
Figure 8: Search Events Screen	23
Figure 9: Invoice Program Main Menu	N-1
Figure 10: Invoice Program Invoice Detail Screen	N-2
Figure 11: Invoice Program Search Screen	N2
Figure 12: CARS Program Main Startup Screen.....	N-3
Figure 13: CARS Program New Order Screen.....	N-4
Figure 14: CARS Program Order Details Screen.....	N-5
Figure 15: New Item Order Screen	N-5
Figure 16: Online Catalog Order Form.....	N-6

1. Introduction

The idea behind this project came out of a discussion I had last year with one of my supervisors at work. Over the course of the past ten years we have expanded our catering offerings from the simple chicken and beans that we had become famous for to offering full service catering at any level. This increase in business has led to a tremendous increase in terms of both sales and customer count.

Along with the increase in business from our catering division came a new set of problems that we were unprepared to face. We at Hy-Vee have always prided ourselves at giving outstanding customer service. It is the motto that we run the company by. The problem that we were faced with was that our service level for our catering business was not up to the level that the company and our clients expect. We were losing information on some of our regular clients too frequently for comfort along with being faced with people calling us year after year wanting to do the same event as in the past and having no idea what it was that we had done for them previously. We knew that if we were going to make sure our business didn't suffer from our poor recordkeeping systems, we needed to make a change.

The solution to our problem was easy enough to envision. We had been using computer based systems for years to handle other aspects of our business and thought that it would be a natural fit for us to continue in this way to solve our catering problem. A review of the available commercial system for sale showed

us that it would be far better for us to create our own system from both a financial standpoint and from a usability point of view.

By looking at the key information that we needed to track and the design of existing store systems, I was able to create a program that allows users to enter information about both clients and catering events. They are able to update the information at anytime and can search through the records based on some selected criteria.

The end result of this program is a system that has allowed for us to better track our catering and relieve much of the problems that have been plaguing us in the past.

2. Statement of Problem

Each of the company's 180+ kitchens offer some sort of catering at a variety of service levels. These can range from the very small to events with tens of thousands of people in attendance. Each of these events is important to the company and as such we strive to keep records of all of the various events that we serve at for future reference and to better understand our business.

2.1 Current Methods of Recordkeeping

The problem that we were faced with is that there was no central method of storing information on these events. Currently there are two ways used in stores, both of which are severely limited in both effectiveness and easy of use.

2.1.1 Memory Method

The first of the two methods that are used is to rely on the memory of the person who either took the order or the person who filled the order. This method is by far the least preferred method of storage of records. Not only are they dependent solely on the memory of that person, there is no way of checking back on an order once that person has left, be it for the day or for good.

2.1.2 Order Form Method

The second widely used method of recordkeeping for our records involves keeping the hard copy of the order that we use in preparing for the event. While this eliminates the problems of someone not remembering an order, it does bring up a whole new set of problems. Among these are the physical space needed to

store such records, the safekeeping of these records, and the lack of an ability to quickly reference any past orders.

As one can see, both of the systems that we are employing today are at best antiquated and totally useless in an age of information technology. We, as a company, have always prided ourselves on being on the forefront of the industry and need to change the current method to reflex that position.

2.2 Review of Existing Industry Solutions

A review of the current situation in the industry shows that there is little in terms of software that will allow us to track our orders.

When I was reviewing software that is currently being used in other businesses I found several similarities between most of them. In almost every case, the current software is designed more for the pricing and management of caterers. This means that the program allows the user to enter the guest count and a menu while the software calculates the amount of food needed as well as an inventory control list. This is perfect for a company that does only caterings or one that needs to have a better way of tracking inventory. This is not our case.

The other thing that the available software has going against it for the company to put into use is the cost factor. The prices for some of these packages can run from a few hundred dollars to several thousand dollars for the base package plus an additional couple hundred dollars per site. This translates into a huge financial investment for the company and one that we are not ready to make.

By designing and constructing our own package, we are going to be able to incorporate the features that we need and want without having to pay for extras that are just not going to get used. We are also able to save the cost of purchasing the commercial software and are going to be able to provide any needed updates in a faster fashion than otherwise obtainable.

3. Objectives

The objectives of this project followed the goals that I laid out in the planning phase fairly closely. Those goals were:

1. To develop a better method of tracking event information.
2. To develop a client information storage system that allows for better and faster access to information.
3. To provide for a process for reporting event information based on user inputted criteria.
4. To develop an easy to understand user interface that will allow the end user to meet the above listed goals.

These goals were identified by both my managers and myself working with each other. These represented what we thought were the best goals needed to fulfill our business demands while at the same time enabling me to put into practice the skills that I learned from the MSIS program.

All of the goals that were identified were to be met with the final set of deliverables and this proved to be the case

3.1 Deliverables

As I mentioned above, the successful completion of this project resulted in the release of the final set of deliverables.

3.1.1 A completed set of databases to store client and event information.

These two databases are the background of the new system. They enable us to store the information about both our customers and the individual catering events. These are Microsoft Access databases. There are two separate databases; one for clients and one for events.

3.1.2 A working program that allows people to work with the above databases.

The program is obviously the most important part of the project from the standpoint of the end-user. It was with them in mind that the design and construction took place. The program was developed to be easy to use and run with minimal effect on system resources to better performance. The program was developed using Microsoft VisualBasic.NET.

3.1.3 A completed and approved project report.

This is the report that you are currently reading. It is done to allow people to see the final result of my work and to fulfill the requirements for my MSIS degree.

4. Scope of Action

This project proved to be more of a challenge than I originally thought it would be. The work to develop a new system and the processes that go in to it was definitely more involving than it first appeared.

4.1 Review Existing Systems

The first step was to look at the current system and decide what changes were needed to make it an effective part of our business. From the start it was quite easy to see that a new way of doing the same old thing just wouldn't cut it. We needed to do more than just update the system.

We needed a way to make this job easier and more convenient for the users while still providing the information needed to carry out our jobs. To this end we decided to take advantage of the technology that we use everyday in other areas of the company and explore using a computer based system to store the records.

The existing two systems for storing information did have some elements included with them that we thought were valuable enough to incorporate into the new systems design.

The information on the paper forms was concise and included only what we needed to know to complete the job. It also was able to provide us with a standard format for order taking. Prior to the creation of these forms six years ago, orders were taken on whatever piece of paper was convenient.

4.2 Review of Available Commercial Alternatives

After I decided that a computerized storage system was the way to go, I began by looking at what was available to us from the commercial market. We have looked at using commercial products before in our food service operations so this wasn't an option that was way out of line.

There are quite literally hundreds of software packages out there that can be used for restaurant and catering management. I began my look for a commercial product by reviewing several of the products recommended in a few of the foodservice trade magazines that we receive at the store. This provided me with enough information to pick out three versions that had been mentioned with what seemed like the basic features that we needed. These three included the following:

- CaterWare
- CaterEdge
- Efficient Frontiers

As I said earlier, all three of these products were picked from trade publications. After a more extensive review of these three products I realized that a commercial product wasn't going to work in this case. This proved to be the wrong direction for us to take.

It was cost prohibitive for us to employ one of these systems. Cost estimates for a basic system came in at a low figure of \$2500 for the basic software package and an additional \$200 per site that employed it. That came to

over \$38,000 for a system that wasn't all we were looking for and had some features included that we would never use.

During the search for a commercial product it was also able to see that the products out there were not what we needed. All of the products that I looked at were either much more than we needed or was lacking in some of the basic features we wanted.

The only other solution that I could come up with to solve our problem was to design and construct a new system myself.

4.3 System Design Standards Establishment

The design and construction phase of the project began with sitting down with my manager as well as some of the other higher staff members. While I lead the discussion, they gave me ideas toward what they would like to see the new system involve. These ideas were discussed among everyone before deciding if they were worth including in the final design specifications.

The first item that we discussed was what information we saw as important enough to store in the system. We all agreed that the program needed to allow us to have the basic contact information about the clients that we work with. We also needed all of the information about the different events that we did. This included the basic who, what, where, and when of each event. We also discussed the need for us to keep payment information for each of the events. We decided that while we should know what we charged the client, we did not need to have information about payment such as store charge records or

credit card receipts. Those are handled separately by the bookkeeping department.

To go along with the basic information that we thought was needed was the overall theme of simplicity for the program. The users liked the design of the current programs that they use and wanted the new program to try and match that style. They also wanted me to try and keep the information from the current cater guide in the same format as possible. It was believed that this would allow for easier mitigation from the current to the new system.

In the end, the catering manager was allowed to give his final approval to all of the suggestions that were presented to him before I proceeded with the project. These approved design specs became the basis for the design limitation sheet that can be found in Appendix K.

4.4 Database Design and Construction

Once the design specs were approved by the catering manager and finalized, I set out to actually build the project. The logical starting point for me was the design and construction of the databases.

4.4.1 Database Design

The design of the database was my primary concern starting out. I had a series of requirements for myself concerning the design.

First, it must have enough data fields available to the program so that all of the needed information could be stored.

Second, it needed to be organized in a fashion that would be easy to understand from both the standpoint of the program as well as the technical support staff that might some day need to upgrade the system.

Finally, it should be convenient for the corporate tech staff to incorporate into our existing network structure.

The final decision was made that we should have two tables included in the database. The first table would be used to store the information on the individual clients that order catering events from us. This table (called Clients) was to be designed basically like a contact management system would. The following fields were included in the table;

Table 1: Client Database Design

Field Name	Data Type	Description
CustomerID	AutoNumber	Stores the unique ID number for each client. This is the primary key for the table.
ContactFirstName	Text	The first name of the contact person.
ContactLastName	Text	The last name of the contact person.
CompanyorDept	Text	The name of the company the client works for. This is an optional field.
BillingAddress	Text	The billing address for a business client. The home address of private individuals.
City	Text	The city the billing address is located in.
State	Text	The state the billing address is located in.
Zip Code	Text	The postal code for the billing address.
Home Phone	Text	
Bus. Phone	Text	
Cellular Phone	Text	
Notes	Memo	

The second of the two tables would be the Events table. This table would be where the information on each individually catered event is stored. The fields

chosen to be part of the table were largely taken off of our existing paper forms.

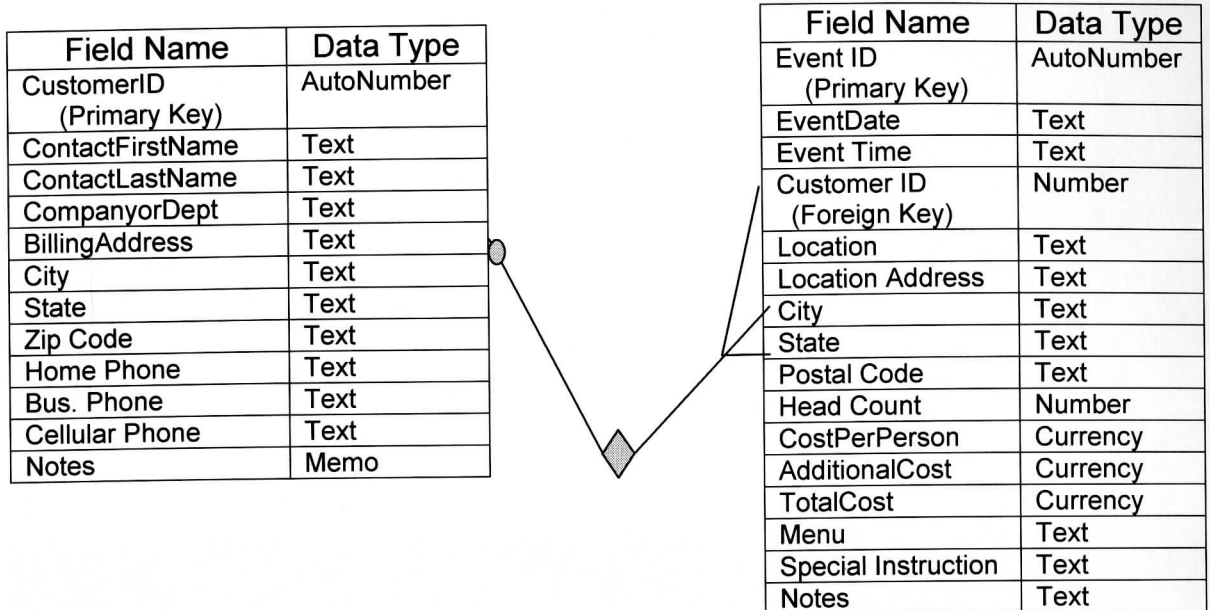
The following is the field listing for the Event database.

Table 2: Event Database Fields.

Field Name	Data Type	Description
Event ID	AutoNumber	Stores the unique ID number for each client. This is the primary key for the table.
EventDate	Text	Stores the date the event takes place on.
Event Time	Text	Stores the time of the event.
Customer ID	Number	A foreign key from the Client table.
Location	Text	The name of the building the event is being held at.
Location Address	Text	The address of the event location,
City	Text	The city the location is in.
State	Text	The state the location is in.
Postal Code	Text	The zip code for the event location
Head Count	Number	Stores the number of people coming to an event.
CostPerPerson	Currency	The cost per person for the event
AdditionalCost	Currency	Any additional cost not included in the pre person cost.
TotalCost	Currency	The total cost of the event.
Menu	Text	
Special Instruction	Text	Any special notes
Notes	Text	Any additional notes about the event.

The two tables are tied together with each other by the use of a foreign key system. The customer ID is the main link between the two tables. The tables share a one-to-many relationship between each other as well. This means that each event has to have one client, but each client can have one or more events. This relationship is shown in Figure 1.

Figure 1: Database Relationship Schema



4.4.2 Database Construction

When the design of the database was completed I turned my attention to the actual construction of it.

I choose to build the database using Microsoft Access. This decision was make for two reasons. First was the availability of the product. Not only did I have access to the program on my personal system at home, but it was also available at the store level and the corporate level already. To use another database would have required me to obtain another program and learn to use that database. For these reasons, Access was a perfect choice.

During the construction of the database is where I did end up making a change from the original proposal that I submitted for this project. This is the one area where I made a change from my original proposal.

In the proposal I submitted to my committee, I was planning on using one database to store all of the information. After I constructed the first example of the database, I received a suggestion from one of the managers I know in our corporate IT department. He suggested that I spit the two fields in to two separate databases. The reasoning behind this came down to storage space.

It is believed that we wouldn't want to remove a client's information for an extended period of years. On the other hand information on the various events we perform every year is only needed to be accessed for no more than three years. It was decided that the events database would needed more frequent archiving and maintenance and it was easier to meet these needs by splitting the database into two.

4.5 Program Design and Construction

With the completion of the databases behind me I turned my attention to the design and construction of the actual program that would be used by the employees.

4.5.1 Program Design

The design of the actual program was my favorite part of this project. I liked being able to start with what was a black screen and add and subtract things until I had the final product ahead of me.

I started the overall design of the program itself by looking at what I needed to accomplish. I knew that I had three basic activities that related to both

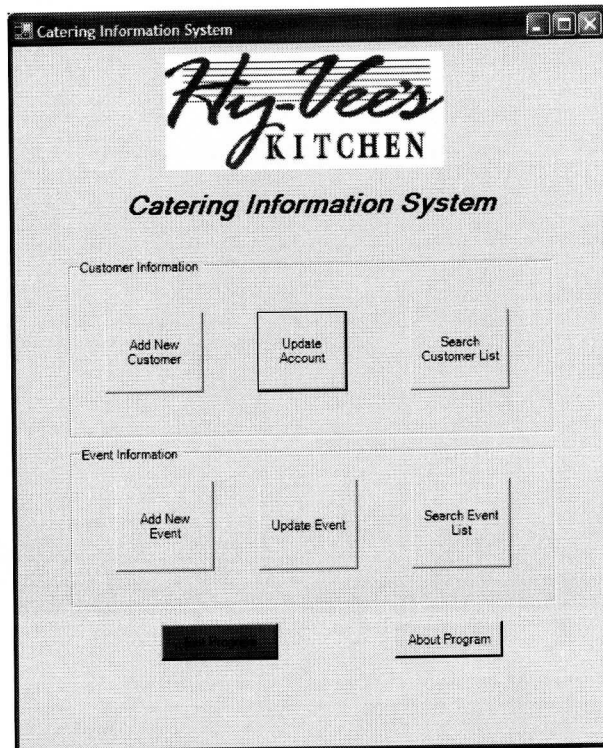
clients and events. I also knew that I wanted to separate out working with both clients and the events as much as possible.

I decided that the easiest way to deal with the construction would be to use a separate windows form for each of the three functions for both main parts. There would be a window for adding new clients or for creating a new event. There would also have to be a separate set of forms that would allow a user to update the information on either the client or any of the events. And lastly there was going to need to be some way of searching through the database for information

4.5.1.1 Main Screen Design

The first part of the program that I designed was the startup screen. This is the first thing that is seen when the user loads up the program. I wanted to make this as easy as possible for people to understand and use. For this reason, I elected to use only a series of buttons that would open other windows that would allow the user to conduct various tasks.

Figure 2: Main Startup Screen

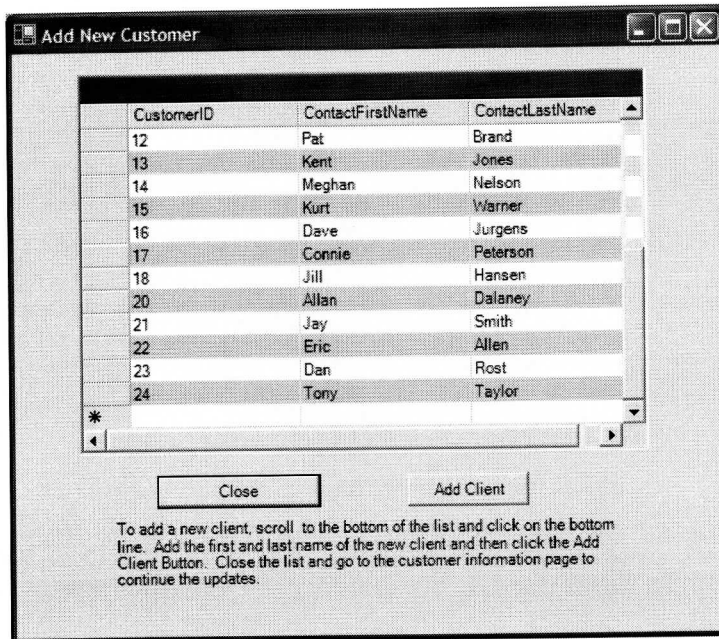


The screen includes buttons to deals with both clients and events. It also includes a button to exit the program as well as one that will bring up the About Program Screen.

4.5.1.2 Add Customer Screen Design

Adding a new client is the second most often done tasks in terms of catering and recordkeeping. I wanted this part of the system to be easy for the user to interface with. My goal is allow the person to enter only the name of the client at this time. All of the other information will be added later.

Figure 3: Add Customer Screen



The instructions for adding a person to the database were listed right on the page. This was done to hopefully cut down on the amount of retraining that is needed.

4.5.1.3 Update Customer Information Screen

This was one of easiest screens for me to design. The information that we wanted to have collected was fairly straight forward. For this screen, I basic created an electronic copy of the contact section of our paper forms.

Figure 4: Update Customer Information Screen

Customer Information

Customer Information

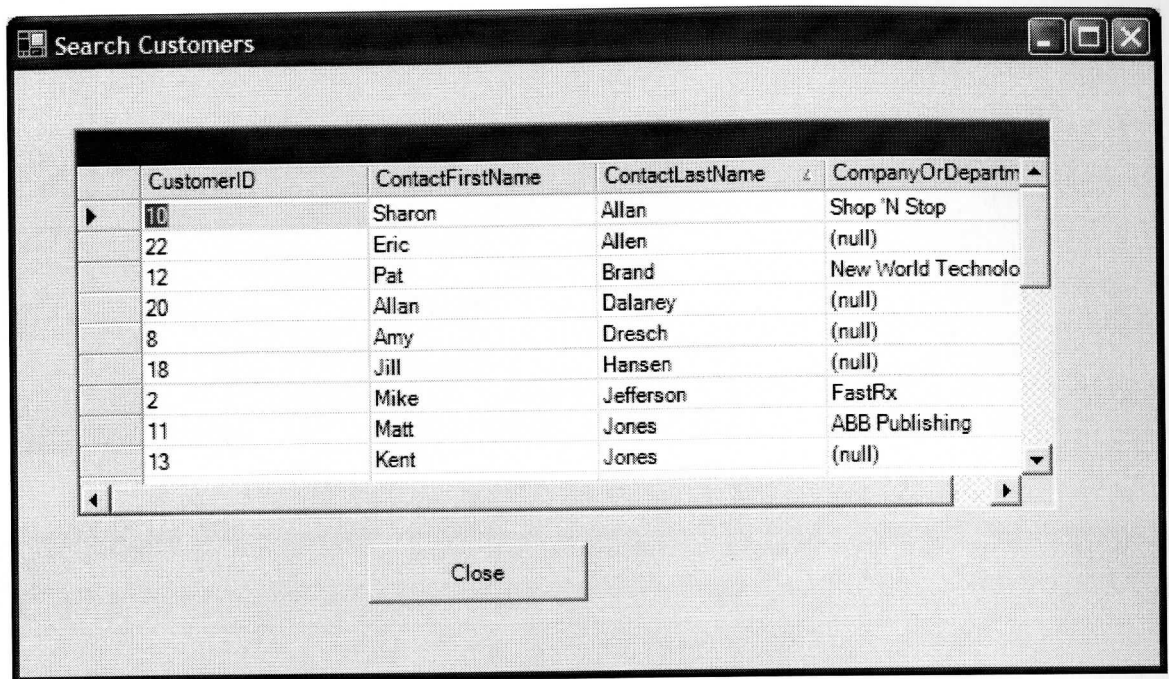
Customer #	9	First Name	Mark	Last Name	Majors
		Company	CCD Research		
Phone # 1	605-388-9823	Address	2384 S. Indiana Ave.		
Phone # 2	605-322-6473	City	Sioux Falls	State	SD
				Zip	57102

First Previous Next Last Update Records

4.5.1.4 Search Customer Screen Design

The design of the Customer Search screen was very easy to do. Based on the design of the Data Grid in Visual Basic, I only had to add a few command buttons to make it work. This screen was more about functionality than it was about artistic design.

Figure 5: Customer Search Screen

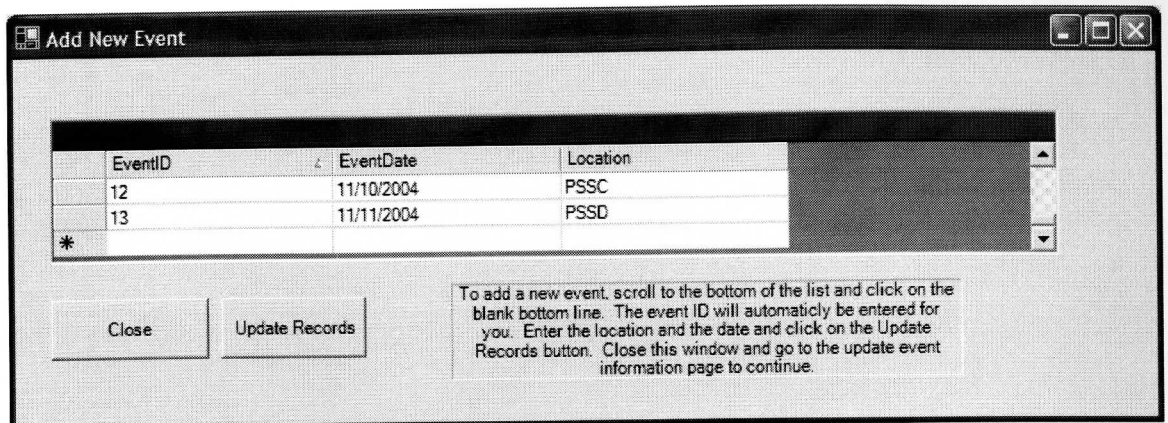


4.5.1.5 New Event Screen Design

The design of the new event screen was basically dictated by the Data Grid that was used to access the database. I wanted to keep the design free from any unnecessary elements. This whole program is about functionality.

As it was with the New Customer screen, the instructions on how to enter a new event are listed right on the screen.

Figure 6: New Event Screen



4.5.1.6 Event Update Screen Design

This proved to be the hardest of all the screens for me to design. I wanted to keep things simple, but needed some sort of order to go with it. What I finally settled on were a series of text boxes grouped together based on their purpose.

Figure 7: Update Event Screen

The screenshot shows a software window titled "Event Information" with a sub-header "Catering Event Information". At the top right of the sub-header are navigation buttons: "First", "Previous", "Next", "Last", and "Update Records". The main area is divided into two sections: "Event Information" and "Payment Information".

Event Information:

Event #	1	Location Name	Oncology Assoc.		
Date	10/23/2004 12:00:	Location Address	MB2		
Service Time	12/1/2004 7:30:00	City	Sioux Falls	State	SD
Customer Number		Zip	57105	Head Count	25
Menu	Chicken, Potatoes and Gravy, Peas and Carrots, Slaw, Cookies, Pop				
Special Instructions	Leave Extra foil pans				
Notes					

Payment Information:

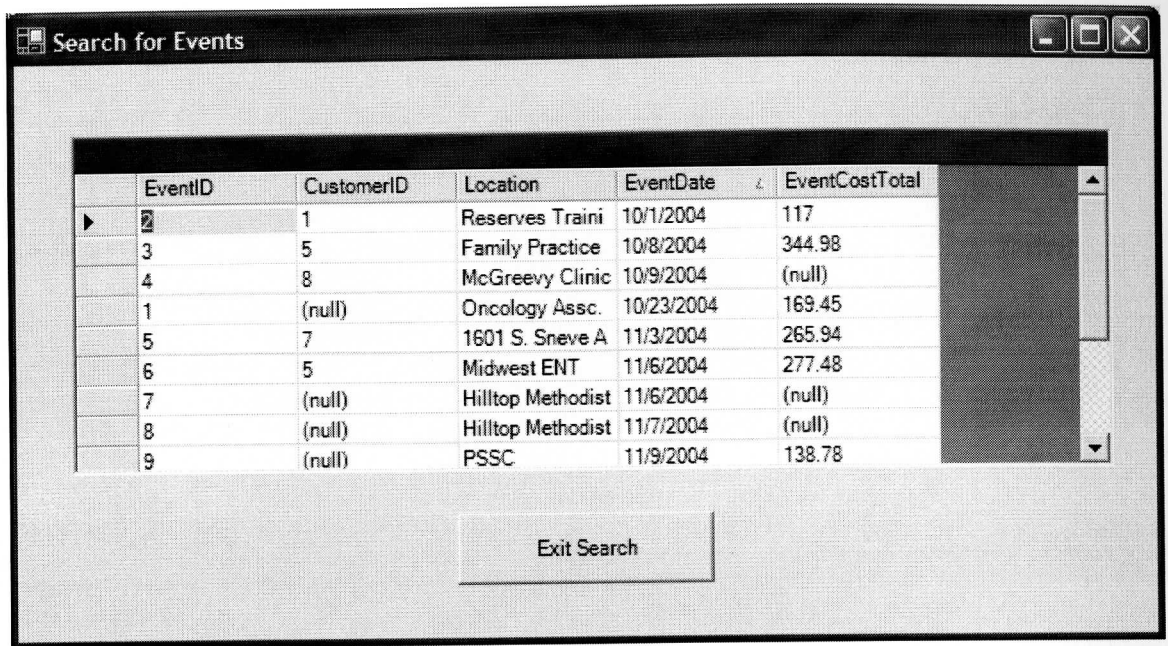
Cost Per Person	5.9	Payment Type		Exp. Date	**/**
Additional Charges	25	CC Number	See Secure Card File	Check #	
Total Cost	169.45	Store Charge Number			

The taskbar at the bottom shows the application name "tart" and several open windows: "Document1...", "Cater Trade...", "Control Panel", "Add or Rem...", "Catering Inf...", and "Event Infor...".

4.5.1.7 Search Event Screen Design

The design for the event search screen followed the same pattern as the Customer Search screen did. The two are for the most part the same page, just accessing different data.

Figure 8: Search Event Screen



4.5.2 Program Construction

After the database was established and the programs design was mostly done, I turned my attention to the actually construction of the program itself. This proved to be the most challenging part of the whole project. While the design of the program was fairly easy to do, the construction proved to be my main sticking point. This is due to largely to my inexperience with Visual Basic.NET.

I started the construction by setting the forms up with the correct titles and such. This was all basic stuff that needed to be done at some point and I choose to start here. At least that way I have a feeling of accomplishment to go on.

The next step I took was the programming on the main startup page to interact with the other pages. I set the commands for the six functions buttons that would allow the user to work with either clients or events. This was all

simple two line command programming. I also added the code to take the user to the 'About Program" form and to exit the program when the right button was clicked on.

The next area that I worked on with the programming was the new customer form. This was when my problems started. The latest version of Visual Basic became very different from the previous version when it came to working with data. Most of the old programming that I had learned was not going to do me any good at this point.

Originally, the New Customer form was supposed to be tied into the same form that you use to update people's information. This was the original intent of the form. The problem that started running into was that the database wouldn't save the changes to the information that I would change. It would give me an error that says it could not update null fields, even when there weren't any.

I worked on this problem the better part of a week with little success. After consulting several different sources both online and in print, I found myself with little choice but to redesign the system in order for it to work the way I needed it to.

With the decision to move forward with the project using the new method of entering new contacts and events decided, I turned my attention to the update functions of both. These two forms were the ones that I knew I had to get working for the project to be a success. I was able to get the forms to update and save those updates to the database after a few days of trying several things.

Upon getting the update forms working for both the clients and events, I turn to the last section I had to go. The search forms were done together and were coded the same way. I was originally intending to make this part more interactive with the user, allowing them to enter the values they would like to search for.

After reading the results of the survey that I conducted among catering managers, I decided that this would be just as easy to accomplish to their standards by simply using a Data Grid that included the fields one would want to search by. It proved to be an easy and simple way for the end user to quickly conduct the search they want without being too confusing.

Once the program was up and working, I installed it on a standalone computer in our store for testing purposes. Training on how to use the system was provided to the users along the way. We ran the system for one week and then gathered feedback from the various users. This feedback was used to make some modifications to the original design of the system. These modifications included some simple changes in appearance of the forms as well as the addition of color in a few areas.

The system was updated and allowed to run again for another week. The feedback at this time was positive all around and this version of the program became the final program.

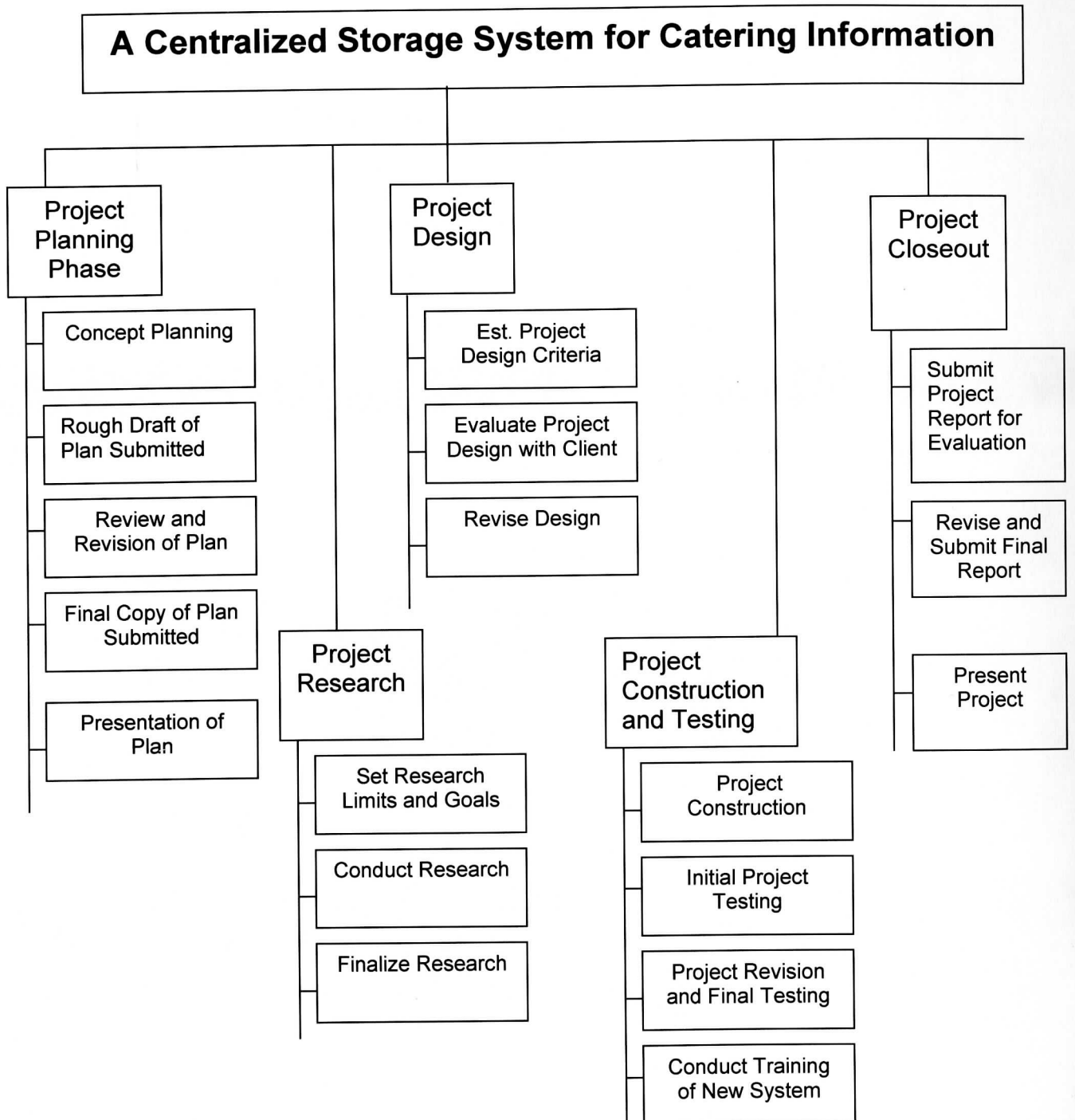
5. Results and Conclusions

The end result of the work that I did on this project proved to be the complete and working program that does exactly what it was supposed to. We are now able to help our repeat clients easier and faster than before. We have cut down on the amount of paperwork that we need to store every month and are able to track our business with a level that we have never had before. We were also able to save the company several thousands of dollars by creating our own system versus purchasing commercial software.

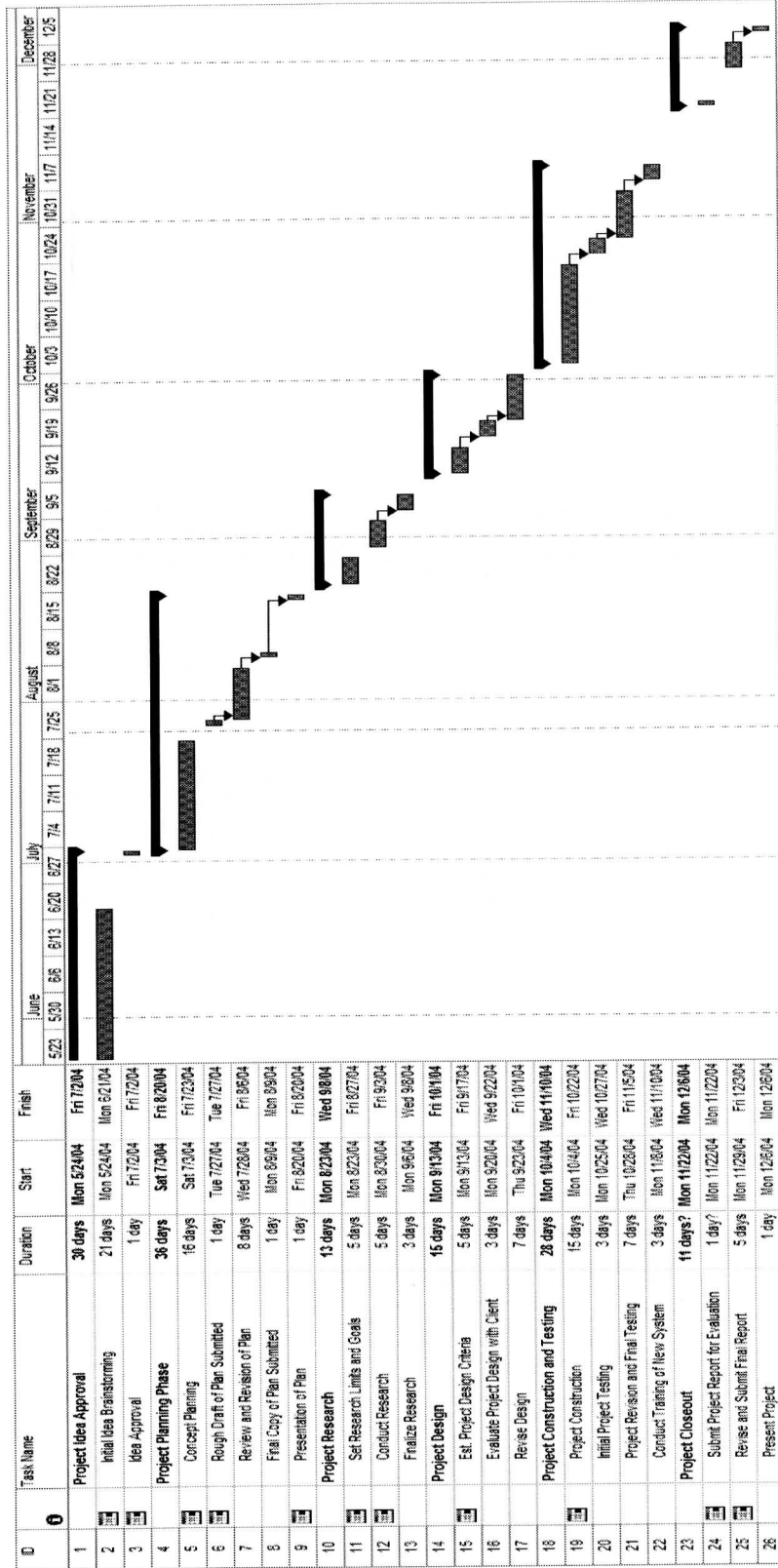
I feel that the completed project will prove useful to our stores for some time to come. On a related note, there are already some plans for improvements that I am working on.

This project has also allowed me to fully put to use the skills that I have learned during the course of the MSIS program that I have taken from DSU. I will admit that I at first thought that some of the course I have taken would not be of any use to me down the road. They were just the ones that were needed to give me some background. This project proved me wrong. All of the material that I studied during the last few years proved useful in this project.

Appendix A - Work Breakdown Structure



Appendix B - Gantt Chart



Appendix C - About Form Code

```
Option Explicit On
Imports VB = Microsoft.VisualBasic

Public Class frmAbout
    Inherits System.Windows.Forms.Form

#Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overrides Sub Dispose(ByVal disposing As
Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form
Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents picIcon As System.Windows.Forms.PictureBox
    Friend WithEvents cmdOK As System.Windows.Forms.Button
    Friend WithEvents lblTitle As System.Windows.Forms.Label
    Friend WithEvents lblVersion As System.Windows.Forms.Label
    Friend WithEvents lblDescription As System.Windows.Forms.Label
    Friend WithEvents lblCopyright As System.Windows.Forms.Label
    Friend WithEvents lblDisclaimer As System.Windows.Forms.Label
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()
        Dim resources As System.Resources.ResourceManager = New
System.Resources.ResourceManager(GetType(frmAbout))
        Me.picIcon = New System.Windows.Forms.PictureBox
        Me.lblTitle = New System.Windows.Forms.Label
        Me.lblVersion = New System.Windows.Forms.Label
        Me.lblDescription = New System.Windows.Forms.Label
        Me.lblCopyright = New System.Windows.Forms.Label
        Me.lblDisclaimer = New System.Windows.Forms.Label
```

```

Me.cmdOK = New System.Windows.Forms.Button
Me.SuspendLayout()
'
'picIcon
'
Me.picIcon.Image = CType(resources.GetObject("picIcon.Image"),
System.Drawing.Image)
Me.picIcon.Location = New System.Drawing.Point(16, 16)
Me.picIcon.Name = "picIcon"
Me.picIcon.Size = New System.Drawing.Size(48, 48)
Me.picIcon.TabIndex = 0
Me.picIcon.TabStop = False
'
'lblTitle
'
Me.lblTitle.Font = New System.Drawing.Font("Microsoft Sans
Serif", 12.0!, System.Drawing.FontStyle.Regular,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.lblTitle.Location = New System.Drawing.Point(88, 8)
Me.lblTitle.Name = "lblTitle"
Me.lblTitle.Size = New System.Drawing.Size(280, 32)
Me.lblTitle.TabIndex = 1
Me.lblTitle.Text = "Hy-Vee Catering Information System"
'
'lblVersion
'
Me.lblVersion.Font = New System.Drawing.Font("Microsoft Sans
Serif", 10.0!, System.Drawing.FontStyle.Regular,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.lblVersion.Location = New System.Drawing.Point(88, 56)
Me.lblVersion.Name = "lblVersion"
Me.lblVersion.Size = New System.Drawing.Size(272, 24)
Me.lblVersion.TabIndex = 2
Me.lblVersion.Text = "Version 1.0"
'
'lblDescription
'
Me.lblDescription.BorderStyle =
System.Windows.Forms.BorderStyle.Fixed3D
Me.lblDescription.Location = New System.Drawing.Point(16, 96)
Me.lblDescription.Name = "lblDescription"
Me.lblDescription.Size = New System.Drawing.Size(352, 80)
Me.lblDescription.TabIndex = 3
Me.lblDescription.Text = "This program has been developed to
allow the Hy-Vee Catering Department a method " & _
    "to more easily track information about their various clients
and catered events." & _
    " This program also represents the final project for the
Dakota State University" & _
    " MSIS Program for the programmer"
Me.lblDescription.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
'
'lblCopyright
'

```

```

        Me.lblCopyright.Font = New System.Drawing.Font("Microsoft Sans
Serif", 11.25!, System.Drawing.FontStyle.Regular,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblCopyright.Location = New System.Drawing.Point(8, 192)
        Me.lblCopyright.Name = "lblCopyright"
        Me.lblCopyright.Size = New System.Drawing.Size(360, 40)
        Me.lblCopyright.TabIndex = 4
        Me.lblCopyright.Text = "Programmer: Matthew D. Johnson
MSIS Candidate - Dakota" & _
        " State University"
        Me.lblCopyright.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
    '
    'lblDisclaimer
    '
        Me.lblDisclaimer.BorderStyle =
System.Windows.Forms.BorderStyle.Fixed3D
        Me.lblDisclaimer.Font = New System.Drawing.Font("Microsoft Sans
Serif", 8.25!, System.Drawing.FontStyle.Regular,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.lblDisclaimer.Location = New System.Drawing.Point(8, 248)
        Me.lblDisclaimer.Name = "lblDisclaimer"
        Me.lblDisclaimer.Size = New System.Drawing.Size(360, 64)
        Me.lblDisclaimer.TabIndex = 5
        Me.lblDisclaimer.Text = "Disclaimer: This software is the
sole work of its programmer. The programmer r" & _
        "eserves all rights to said program. No unauthorized use is
permitted. This p" & _
        "rogram is offered WITHOUT warrenty of any kind."
    '
    'cmdOK
    '
        Me.cmdOK.BackColor = System.Drawing.Color.Yellow
        Me.cmdOK.Font = New System.Drawing.Font("Microsoft Sans Serif",
8.25!, System.Drawing.FontStyle.Bold,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.cmdOK.Location = New System.Drawing.Point(120, 320)
        Me.cmdOK.Name = "cmdOK"
        Me.cmdOK.Size = New System.Drawing.Size(104, 32)
        Me.cmdOK.TabIndex = 6
        Me.cmdOK.Text = "OK"
    '
    'frmAbout
    '
        Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
        Me.ClientSize = New System.Drawing.Size(376, 366)
        Me.Controls.Add(Me.cmdOK)
        Me.Controls.Add(Me.lblDisclaimer)
        Me.Controls.Add(Me.lblCopyright)
        Me.Controls.Add(Me.lblDescription)
        Me.Controls.Add(Me.lblVersion)
        Me.Controls.Add(Me.lblTitle)
        Me.Controls.Add(Me.picIcon)
        Me.Name = "frmAbout"
        Me.Text = "About Catering Information System"
        Me.ResumeLayout(False)

```

End Sub

#End Region

```
*****  
'  
' Form Name: frmAbout  
'  
' Purpose: This form is used to display information about  
' the program and the programmer.  
'  
' Updated: November 10, 2004  
' Updated By: Matthew D. Johnson  
'  
*****
```

```
Private Sub cmdOK_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles cmdOK.Click  
    'This command closes the window when called.  
    Close()  
End Sub  
End Class
```

Appendix D - Add Event Form Code

```
Public Class frmAddEvent
    Inherits System.Windows.Forms.Form

    #Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overrides Sub Dispose(ByVal disposing As
Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form
Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents OleDbDataAdapter1 As
System.Data.OleDb.OleDbDataAdapter
    Friend WithEvents OleDbSelectCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbInsertCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbUpdateCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbDeleteCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbConnection1 As
System.Data.OleDb.OleDbConnection
    'Friend WithEvents DsEvents1 As WindowsApplication1.DsEvents
    Friend WithEvents DataGridView1 As System.Windows.Forms.DataGridView
    Friend WithEvents btnClose As System.Windows.Forms.Button
    Friend WithEvents btnUpdate As System.Windows.Forms.Button
    Friend WithEvents Label1 As System.Windows.Forms.Label
    Friend WithEvents OleDbDataAdapter2 As
System.Data.OleDb.OleDbDataAdapter
    Friend WithEvents OleDbSelectCommand2 As
System.Data.OleDb.OleDbCommand
```



```

    Friend WithEvents OleDbInsertCommand2 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbUpdateCommand2 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbDeleteCommand2 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents DsEvents21 As WindowsApplication1.DsEvents2
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()
    Me.OleDbDeleteCommand1 = New System.Data.OleDb.OleDbCommand
    Me.OleDbConnection1 = New System.Data.OleDb.OleDbConnection
    Me.OleDbInsertCommand1 = New System.Data.OleDb.OleDbCommand
    Me.OleDbSelectCommand1 = New System.Data.OleDb.OleDbCommand
    Me.OleDbUpdateCommand1 = New System.Data.OleDb.OleDbCommand
    Me.DataGrid1 = New System.Windows.Forms.DataGrid
    Me.DsEvents21 = New WindowsApplication1.DsEvents2
    Me.btnClose = New System.Windows.Forms.Button
    Me.btnUpdate = New System.Windows.Forms.Button
    Me.Label1 = New System.Windows.Forms.Label
    Me.OleDbDataAdapter2 = New System.Data.OleDb.OleDbDataAdapter
    Me.OleDbDeleteCommand2 = New System.Data.OleDb.OleDbCommand
    Me.OleDbInsertCommand2 = New System.Data.OleDb.OleDbCommand
    Me.OleDbSelectCommand2 = New System.Data.OleDb.OleDbCommand
    Me.OleDbUpdateCommand2 = New System.Data.OleDb.OleDbCommand
    CType(Me.DataGrid1,
System.ComponentModel.ISupportInitialize).BeginInit()
    CType(Me.DsEvents21,
System.ComponentModel.ISupportInitialize).BeginInit()
    Me.SuspendLayout()
    '
    'OleDbDeleteCommand1
    '
    Me.OleDbDeleteCommand1.CommandText = "DELETE FROM Events WHERE
(EventID = ?) AND (CustomerID = ? OR ? IS NULL AND Custo" & _
    "merID IS NULL) AND (HeadCount = ?) AND (Location = ? OR ? IS
NULL AND Location I" & _
    "S NULL) "
    Me.OleDbDeleteCommand1.Connection = Me.OleDbConnection1
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventID", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID1",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_HeadCount",
System.Data.OleDb.OleDbType.Boolean, 2,

```

```

System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "HeadCount", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location1",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
    ,
    'OleDbConnection1
    ,
    Me.OleDbConnection1.ConnectionString = "Jet OLEDB:Global
Partial Bulk Ops=2;Jet OLEDB:Registry Path=;Jet OLEDB:Database L" & _
"ocking Mode=1;Jet OLEDB:Database Password=;Data
Source=""C:\catering\Events.mdb"";" & _
"Password=;Jet OLEDB:Engine Type=5;Jet OLEDB:Global Bulk
Transactions=1;Provider=" & _
""Microsoft.Jet.OLEDB.4.0"";Jet OLEDB:System database=;Jet
OLEDB:SFP=False;Extende" & _
"d Properties=;Mode=Share Deny None;Jet OLEDB:New Database
Password=;Jet OLEDB:Cr" & _
"eate System Database=False;Jet OLEDB:Don't Copy Locale on
Compact=False;Jet OLEDB" & _
"B:Compact Without Replica Repair=False;User ID=Admin;Jet
OLEDB:Encrypt Database=" & _
"False"
    ,
    'OleDbInsertCommand1
    ,
    Me.OleDbInsertCommand1.CommandText = "INSERT INTO
Events(CustomerID, Location, HeadCount) VALUES (?, ?, ?)"
    Me.OleDbInsertCommand1.Connection = Me.OleDbConnection1
    Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("CustomerID",
System.Data.OleDb.OleDbType.Integer, 0, "CustomerID"))
    Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Location",
System.Data.OleDb.OleDbType.VarWChar, 255, "Location"))
    Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("HeadCount",
System.Data.OleDb.OleDbType.Boolean, 2, "HeadCount"))
    ,
    'OleDbSelectCommand1
    ,
    Me.OleDbSelectCommand1.CommandText = "SELECT EventID,
CustomerID, Location, HeadCount FROM Events"
    Me.OleDbSelectCommand1.Connection = Me.OleDbConnection1
    ,
    'OleDbUpdateCommand1
    ,
    Me.OleDbUpdateCommand1.CommandText = "UPDATE Events SET
CustomerID = ?, Location = ?, HeadCount = ? WHERE (EventID = ?)" & _

```

```

" AND (CustomerID = ? OR ? IS NULL AND CustomerID IS NULL) AND
(HeadCount = ?) AN" & _
"D (Location = ? OR ? IS NULL AND Location IS NULL)"
Me.OleDbUpdateCommand1.Connection = Me.OleDbConnection1
Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("CustomerID",
System.Data.OleDb.OleDbType.Integer, 0, "CustomerID"))
Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Location",
System.Data.OleDb.OleDbType.VarWChar, 255, "Location"))
Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("HeadCount",
System.Data.OleDb.OleDbType.Boolean, 2, "HeadCount"))
Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventID", System.Data.DataRowVersion.Original, Nothing))
Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID1",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_HeadCount",
System.Data.OleDb.OleDbType.Boolean, 2,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "HeadCount", System.Data.DataRowVersion.Original, Nothing))
Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location1",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
,
'DataGrid1
,
Me.DataGrid1.DataMember = "Events"
Me.DataGrid1.DataSource = Me.DsEvents21
Me.DataGrid1.HeaderForeColor =
System.Drawing.SystemColors.ControlText
Me.DataGrid1.Location = New System.Drawing.Point(24, 40)
Me.DataGrid1.Name = "DataGrid1"
Me.DataGrid1.PreferredColumnWidth = 150
Me.DataGrid1.Size = New System.Drawing.Size(664, 96)
Me.DataGrid1.TabIndex = 0
,
'DsEvents21

```

```

    Me.DsEvents21.DataSetName = "DsEvents2"
    Me.DsEvents21.Locale = New
System.Globalization.CultureInfo("en-US")
    'btnClose
    '
    Me.btnClose.Location = New System.Drawing.Point(24, 160)
    Me.btnClose.Name = "btnClose"
    Me.btnClose.Size = New System.Drawing.Size(104, 40)
    Me.btnClose.TabIndex = 2
    Me.btnClose.Text = "Close"
    '
    'btnUpdate
    '
    Me.btnUpdate.Location = New System.Drawing.Point(136, 160)
    Me.btnUpdate.Name = "btnUpdate"
    Me.btnUpdate.Size = New System.Drawing.Size(96, 40)
    Me.btnUpdate.TabIndex = 3
    Me.btnUpdate.Text = "Update Records"
    '
    'Labell
    '
    Me.Labell.BorderStyle =
System.Windows.Forms.BorderStyle.Fixed3D
    Me.Labell.Location = New System.Drawing.Point(288, 152)
    Me.Labell.Name = "Labell"
    Me.Labell.Size = New System.Drawing.Size(336, 64)
    Me.Labell.TabIndex = 4
    Me.Labell.Text = "To add a new event, scroll to the bottom of
the list and click on the blank botto" & _
    "m line. The event ID will automaticly be entered for you.
Enter the location a" & _
    "nd the date and click on the Update Records button. Close
this window and go to" & _
    " the update event information page to continue."
    Me.Labell.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
    '
    'OleDbDataAdapter2
    '
    Me.OleDbDataAdapter2.DeleteCommand = Me.OleDbDeleteCommand2
    Me.OleDbDataAdapter2.InsertCommand = Me.OleDbInsertCommand2
    Me.OleDbDataAdapter2.SelectCommand = Me.OleDbSelectCommand2
    Me.OleDbDataAdapter2.TableMappings.AddRange(New
System.Data.Common.DataTableMapping() {New
System.Data.Common.DataTableMapping("Table", "Events", New
System.Data.Common.DataColumnMapping() {New
System.Data.Common.DataColumnMapping("EventID", "EventID"), New
System.Data.Common.DataColumnMapping("EventDate", "EventDate"), New
System.Data.Common.DataColumnMapping("Location", "Location")}}})
    Me.OleDbDataAdapter2.UpdateCommand = Me.OleDbUpdateCommand2
    '
    'OleDbDeleteCommand2
    '
    Me.OleDbDeleteCommand2.CommandText = "DELETE FROM Events WHERE
(EventID = ?) AND (EventDate = ? OR ? IS NULL AND EventD" & _

```

```

"ate IS NULL) AND (Location = ? OR ? IS NULL AND Location IS
NULL)"
    Me.OleDbDeleteCommand2.Connection = Me.OleDbConnection1
    Me.OleDbDeleteCommand2.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventID", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand2.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventDate",
System.Data.OleDb.OleDbType.DBDate, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventDate", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand2.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventDate1",
System.Data.OleDb.OleDbType.DBDate, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventDate", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand2.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand2.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location1",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
    ,
    'OleDbInsertCommand2
    ,
    Me.OleDbInsertCommand2.CommandText = "INSERT INTO
Events(EventDate, Location) VALUES (?, ?)"
    Me.OleDbInsertCommand2.Connection = Me.OleDbConnection1
    Me.OleDbInsertCommand2.Parameters.Add(New
System.Data.OleDb.OleDbParameter("EventDate",
System.Data.OleDb.OleDbType.DBDate, 0, "EventDate"))
    Me.OleDbInsertCommand2.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Location",
System.Data.OleDb.OleDbType.VarWChar, 255, "Location"))
    ,
    'OleDbSelectCommand2
    ,
    Me.OleDbSelectCommand2.CommandText = "SELECT EventID,
EventDate, Location FROM Events"
    Me.OleDbSelectCommand2.Connection = Me.OleDbConnection1
    ,
    'OleDbUpdateCommand2
    ,
    Me.OleDbUpdateCommand2.CommandText = "UPDATE Events SET
EventDate = ?, Location = ? WHERE (EventID = ?) AND (EventDate " & _
"= ? OR ? IS NULL AND EventDate IS NULL) AND (Location = ? OR ?
IS NULL AND Locat" & _
"ion IS NULL)"
    Me.OleDbUpdateCommand2.Connection = Me.OleDbConnection1

```

```

        Me.OleDbUpdateCommand2.Parameters.Add(New
System.Data.OleDb.OleDbParameter("EventDate",
System.Data.OleDb.OleDbType.DBDate, 0, "EventDate"))
        Me.OleDbUpdateCommand2.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Location",
System.Data.OleDb.OleDbType.VarWChar, 255, "Location"))
        Me.OleDbUpdateCommand2.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventID", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand2.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventDate",
System.Data.OleDb.OleDbType.DBDate, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventDate", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand2.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventDate1",
System.Data.OleDb.OleDbType.DBDate, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventDate", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand2.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand2.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location1",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
    ,
    ' frmAddEvent
    ,
    Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
    Me.ClientSize = New System.Drawing.Size(752, 238)
    Me.Controls.Add(Me.Label1)
    Me.Controls.Add(Me.btnUpdate)
    Me.Controls.Add(Me.btnClose)
    Me.Controls.Add(Me.DataGrid1)
    Me.Name = "frmAddEvent"
    Me.Text = "Add New Event"
    CType(Me.DataGrid1,
System.ComponentModel.ISupportInitialize).EndInit()
    CType(Me.DsEvents21,
System.ComponentModel.ISupportInitialize).EndInit()
    Me.ResumeLayout(False)

End Sub

#End Region

! *****
!
!   Form Name:   frmAddEvent
!
! *****

```

```

' Purpose: This form is used to add a new event to the
' database.
'
' Updated: November 10, 2004
' Updated By: Matthew D. Johnson
'
'*****

Private Sub btnClose_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnClose.Click
    'The command is called when called from the form. It is used
to close the window.
    Close()
End Sub

Private Sub frmAddEvent_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load
    'This command is called when the form loads. It is used to
fill the dataset with the information
    'from the database.
    DsEvents21.Clear()
    OleDbDataAdapter2.Fill(DsEvents21)
End Sub

Private Sub btnUpdate_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnUpdate.Click
    'The command is used to update the database after the user has
made changes to the form.
    Dim Current As Short
    Current = Me.BindingContext(DsEvents21, "Events").Position
    Me.BindingContext(DsEvents21, "Events").Position = 0
    Me.BindingContext(DsEvents21, "Events").Position = Current
    OleDbDataAdapter2.Update(DsEvents21)
    MsgBox("Update Complete", MsgBoxStyle.Information, "Update
Confirmation")
End Sub
End Class

```

Appendix E - Add Customer Form Code

```
Public Class frmAddCust
    Inherits System.Windows.Forms.Form

#Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overloads Overrides Sub Dispose(ByVal disposing As
Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form
Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents OleDbDataAdapter1 As
System.Data.OleDb.OleDbDataAdapter
    Friend WithEvents OleDbSelectCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbInsertCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbUpdateCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbDeleteCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbConnection1 As
System.Data.OleDb.OleDbConnection
    Friend WithEvents DataGrid1 As System.Windows.Forms.DataGrid
    Friend WithEvents DsCustomers1 As WindowsApplication1.dsCustomers
    Friend WithEvents btnClose As System.Windows.Forms.Button
    Friend WithEvents btnUpdate As System.Windows.Forms.Button
    Friend WithEvents lblDirections As System.Windows.Forms.Label
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()
        Me.OleDbDataAdapter1 = New System.Data.OleDb.OleDbDataAdapter
        Me.OleDbDeleteCommand1 = New System.Data.OleDb.OleDbCommand
        Me.OleDbConnection1 = New System.Data.OleDb.OleDbConnection
```



```

Me.OleDbInsertCommand1 = New System.Data.OleDb.OleDbCommand
Me.OleDbSelectCommand1 = New System.Data.OleDb.OleDbCommand
Me.OleDbUpdateCommand1 = New System.Data.OleDb.OleDbCommand
Me.DsCustomers1 = New WindowsApplication1.dsCustomers
Me.DataGrid1 = New System.Windows.Forms.DataGrid
Me.btnClose = New System.Windows.Forms.Button
Me.btnUpdate = New System.Windows.Forms.Button
Me.lblDirections = New System.Windows.Forms.Label
CType(Me.DsCustomers1,
System.ComponentModel.ISupportInitialize).BeginInit()
CType(Me.DataGrid1,
System.ComponentModel.ISupportInitialize).BeginInit()
Me.SuspendLayout()
,
'OleDbDataAdapter1
,
Me.OleDbDataAdapter1.DeleteCommand = Me.OleDbDeleteCommand1
Me.OleDbDataAdapter1.InsertCommand = Me.OleDbInsertCommand1
Me.OleDbDataAdapter1.SelectCommand = Me.OleDbSelectCommand1
Me.OleDbDataAdapter1.TableMappings.AddRange(New
System.Data.Common.DataTableMapping() {New
System.Data.Common.DataTableMapping("Table", "Customers", New
System.Data.Common.DataColumnMapping() {New
System.Data.Common.DataColumnMapping("CustomerID", "CustomerID"), New
System.Data.Common.DataColumnMapping("ContactFirstName",
"ContactFirstName"), New
System.Data.Common.DataColumnMapping("ContactLastName",
"ContactLastName")}}})
Me.OleDbDataAdapter1.UpdateCommand = Me.OleDbUpdateCommand1
,
'OleDbDeleteCommand1
,
Me.OleDbDeleteCommand1.CommandText = "DELETE FROM Customers
WHERE (CustomerID = ?) AND (ContactFirstName = ? OR ? IS NU" & _
"LL AND ContactFirstName IS NULL) AND (ContactLastName = ? OR ?
IS NULL AND Conta" & _
"ctLastName IS NULL)"
Me.OleDbDeleteCommand1.Connection = Me.OleDbConnection1
Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactFirstName",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactFirstName", System.Data.DataRowVersion.Original,
Nothing))
Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactFirstName1",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactFirstName", System.Data.DataRowVersion.Original,
Nothing))
Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactLastName",

```

```

System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactLastName", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactLastName1",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactLastName", System.Data.DataRowVersion.Original,
Nothing))
    ,
    'OleDbConnection1
    ,
    Me.OleDbConnection1.ConnectionString = "Jet OLEDB:Global
Partial Bulk Ops=2;Jet OLEDB:Registry Path=;Jet OLEDB:Database L" & _
"ocking Mode=1;Jet OLEDB:Database Password=;Data
Source="C:\catering\Client.mdb";" & _
"Password=;Jet OLEDB:Engine Type=5;Jet OLEDB:Global Bulk
Transactions=1;Provider=" & _
""Microsoft.Jet.OLEDB.4.0"";Jet OLEDB:System database=;Jet
OLEDB:SFP=False;Extende" & _
"d Properties=;Mode=Share Deny None;Jet OLEDB:New Database
Password=;Jet OLEDB:Cr" & _
"eate System Database=False;Jet OLEDB:Don't Copy Locale on
Compact=False;Jet OLEDB" & _
"B:Compact Without Replica Repair=False;User ID=Admin;Jet
OLEDB:Encrypt Database=" & _
"False"
    ,
    'OleDbInsertCommand1
    ,
    Me.OleDbInsertCommand1.CommandText = "INSERT INTO
Customers(ContactFirstName, ContactLastName) VALUES (?, ?)"
    Me.OleDbInsertCommand1.Connection = Me.OleDbConnection1
    Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("ContactFirstName",
System.Data.OleDb.OleDbType.VarWChar, 30, "ContactFirstName"))
    Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("ContactLastName",
System.Data.OleDb.OleDbType.VarWChar, 50, "ContactLastName"))
    ,
    'OleDbSelectCommand1
    ,
    Me.OleDbSelectCommand1.CommandText = "SELECT CustomerID,
ContactFirstName, ContactLastName FROM Customers"
    Me.OleDbSelectCommand1.Connection = Me.OleDbConnection1
    ,
    'OleDbUpdateCommand1
    ,
    Me.OleDbUpdateCommand1.CommandText = "UPDATE Customers SET
ContactFirstName = ?, ContactLastName = ? WHERE (CustomerID " & _
"= ?) AND (ContactFirstName = ? OR ? IS NULL AND
ContactFirstName IS NULL) AND (C" & _
"ontactLastName = ? OR ? IS NULL AND ContactLastName IS NULL)"
    Me.OleDbUpdateCommand1.Connection = Me.OleDbConnection1

```

```

        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("ContactFirstName",
System.Data.OleDb.OleDbType.VarWChar, 30, "ContactFirstName"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("ContactLastName",
System.Data.OleDb.OleDbType.VarWChar, 50, "ContactLastName"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactFirstName",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactFirstName", System.Data.DataRowVersion.Original,
Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactFirstName1",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactFirstName", System.Data.DataRowVersion.Original,
Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactLastName",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactLastName", System.Data.DataRowVersion.Original,
Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactLastName1",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactLastName", System.Data.DataRowVersion.Original,
Nothing))
    ,
    'DsCustomers1
    ,
    Me.DsCustomers1.DataSetName = "dsCustomers"
    Me.DsCustomers1.Locale = New
System.Globalization.CultureInfo("en-US")
    ,
    'DataGrid1
    ,
    Me.DataGrid1.Anchor =
CType(((System.Windows.Forms.AnchorStyles.Top Or
System.Windows.Forms.AnchorStyles.Bottom) _
Or System.Windows.Forms.AnchorStyles.Left) _
Or System.Windows.Forms.AnchorStyles.Right),
System.Windows.Forms.AnchorStyles)
    Me.DataGrid1.BackColor =
System.Drawing.SystemColors.InactiveBorder
    Me.DataGrid1.DataMember = "Customers"
    Me.DataGrid1.DataSource = Me.DsCustomers1
    Me.DataGrid1.HeaderForeColor =
System.Drawing.SystemColors.ControlText
    Me.DataGrid1.Location = New System.Drawing.Point(48, 16)

```

```

Me.DataGrid1.Name = "DataGrid1"
Me.DataGrid1.PreferredColumnWidth = 125
Me.DataGrid1.Size = New System.Drawing.Size(416, 280)
Me.DataGrid1.TabIndex = 0
'
'btnClose
'
Me.btnClose.Location = New System.Drawing.Point(104, 312)
Me.btnClose.Name = "btnClose"
Me.btnClose.Size = New System.Drawing.Size(120, 24)
Me.btnClose.TabIndex = 2
Me.btnClose.Text = "Close"
'
'btnUpdate
'
Me.btnUpdate.Location = New System.Drawing.Point(288, 312)
Me.btnUpdate.Name = "btnUpdate"
Me.btnUpdate.Size = New System.Drawing.Size(88, 24)
Me.btnUpdate.TabIndex = 3
Me.btnUpdate.Text = "Add Client"
'
'lblDirections
'
Me.lblDirections.Location = New System.Drawing.Point(72, 344)
Me.lblDirections.Name = "lblDirections"
Me.lblDirections.Size = New System.Drawing.Size(376, 64)
Me.lblDirections.TabIndex = 4
Me.lblDirections.Text = "To add a new client, scroll to the
bottom of the list and click on the bottom li" & _
"ne. Add the first and last name of the new client and then
click the Add Client" & _
" Button. Close the list and go to the customer information
page to continue the" & _
" updates."
'
'frmAddCust
'
Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
Me.ClientSize = New System.Drawing.Size(520, 430)
Me.Controls.Add(Me.lblDirections)
Me.Controls.Add(Me.btnUpdate)
Me.Controls.Add(Me.btnClose)
Me.Controls.Add(Me.DataGrid1)
Me.Name = "frmAddCust"
Me.Text = "Add New Customer"
CType(Me.DsCustomers1,
System.ComponentModel.ISupportInitialize).EndInit()
CType(Me.DataGrid1,
System.ComponentModel.ISupportInitialize).EndInit()
Me.ResumeLayout(False)

```

End Sub

#End Region

'*****'

```

' Form Name: frmAddCust
'
' Purpose: This form is used to add a new customer to the
' database.
'
' Updated: November 10, 2004
' Updated By: Matthew D. Johnson
'
'*****

Private Sub btnClose_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnClose.Click
    'This procedure is called when the user clicks on the Close
button. It closes the window.
    Close()
End Sub

Private Sub frmChooseCust_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load
    'This set of procedures are called when the form loads into
memory.
    'In this case, the form Data Adapter is first cleared and then
filled from the Data Set.
    DsCustomers1.Clear()
    OleDbDataAdapter1.Fill(DsCustomers1)
End Sub

Private Sub btnUpdate_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnUpdate.Click
    'This procedure is called when the user clicks on the Add
Customer button.
    'The procedure then updates the database based on the changes
that the user has made.
    Dim Current As Short
    Current = Me.BindingContext(DsCustomers1, "Customers").Position
    Me.BindingContext(DsCustomers1, "Customers").Position = 0
    Me.BindingContext(DsCustomers1, "Customers").Position = Current
    OleDbDataAdapter1.Update(DsCustomers1)
    MsgBox("Update Complete", MsgBoxStyle.Information, "Update
Confirmation")
End Sub
End Class

```

Appendix F - Catering Information Form Code

```
Public Class frmCaterInfoSys
    Inherits System.Windows.Forms.Form

    #Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overrides Sub Dispose(ByVal disposing As
Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form
Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents PictureBox1 As System.Windows.Forms.PictureBox
    Friend WithEvents lblCaterLabel As System.Windows.Forms.Label
    Friend WithEvents GroupBox1 As System.Windows.Forms.GroupBox
    Friend WithEvents GroupBox2 As System.Windows.Forms.GroupBox
    Friend WithEvents btnExit As System.Windows.Forms.Button
    Friend WithEvents BtnAbout As System.Windows.Forms.Button
    Friend WithEvents btnUpdateCust As System.Windows.Forms.Button
    Friend WithEvents btnAddCustomer As System.Windows.Forms.Button
    Friend WithEvents btnSearchCust As System.Windows.Forms.Button
    Friend WithEvents cmdUpdateEvents As System.Windows.Forms.Button
    Friend WithEvents cmdAddEvent As System.Windows.Forms.Button
    Friend WithEvents btnSearchEvents As System.Windows.Forms.Button
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()
        Dim resources As System.Resources.ResourceManager = New
System.Resources.ResourceManager(GetType(frmCaterInfoSys))
        Me.PictureBox1 = New System.Windows.Forms.PictureBox
        Me.lblCaterLabel = New System.Windows.Forms.Label
        Me.GroupBox1 = New System.Windows.Forms.GroupBox
        Me.btnSearchCust = New System.Windows.Forms.Button
        Me.btnAddCustomer = New System.Windows.Forms.Button
```

```

Me.btnUpdateCust = New System.Windows.Forms.Button
Me.GroupBox2 = New System.Windows.Forms.GroupBox
Me.btnSearchEvents = New System.Windows.Forms.Button
Me.cmdAddEvent = New System.Windows.Forms.Button
Me.cmdUpdateEvents = New System.Windows.Forms.Button
Me.btnExit = New System.Windows.Forms.Button
Me.BtnAbout = New System.Windows.Forms.Button
Me.GroupBox1.SuspendLayout()
Me.GroupBox2.SuspendLayout()
Me.SuspendLayout()
'
'PictureBox1
'
Me.PictureBox1.Image =
CType(resources.GetObject("PictureBox1.Image"), System.Drawing.Image)
Me.PictureBox1.Location = New System.Drawing.Point(136, 8)
Me.PictureBox1.Name = "PictureBox1"
Me.PictureBox1.Size = New System.Drawing.Size(248, 104)
Me.PictureBox1.SizeMode =
System.Windows.Forms.PictureBoxSizeMode.StretchImage
Me.PictureBox1.TabIndex = 0
Me.PictureBox1.TabStop = False
'
'lblCaterLabel
'
Me.lblCaterLabel.Font = New System.Drawing.Font("Microsoft Sans
Serif", 18.0!, CType((System.Drawing.FontStyle.Bold Or
System.Drawing.FontStyle.Italic), System.Drawing.FontStyle),
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.lblCaterLabel.Location = New System.Drawing.Point(32, 120)
Me.lblCaterLabel.Name = "lblCaterLabel"
Me.lblCaterLabel.Size = New System.Drawing.Size(464, 48)
Me.lblCaterLabel.TabIndex = 21
Me.lblCaterLabel.Text = "Catering Information System"
Me.lblCaterLabel.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
'
'GroupBox1
'
Me.GroupBox1.Controls.Add(Me.btnSearchCust)
Me.GroupBox1.Controls.Add(Me.btnAddCustomer)
Me.GroupBox1.Controls.Add(Me.btnUpdateCust)
Me.GroupBox1.Location = New System.Drawing.Point(48, 192)
Me.GroupBox1.Name = "GroupBox1"
Me.GroupBox1.Size = New System.Drawing.Size(432, 160)
Me.GroupBox1.TabIndex = 22
Me.GroupBox1.TabStop = False
Me.GroupBox1.Text = "Customer Information"
'
'btnSearchCust
'
Me.btnSearchCust.Location = New System.Drawing.Point(304, 48)
Me.btnSearchCust.Name = "btnSearchCust"
Me.btnSearchCust.Size = New System.Drawing.Size(88, 72)
Me.btnSearchCust.TabIndex = 11
Me.btnSearchCust.Text = "Search Customer List"
'

```

```

'btnAddCustomer
,
Me.btnAddCustomer.Location = New System.Drawing.Point(32, 48)
Me.btnAddCustomer.Name = "btnAddCustomer"
Me.btnAddCustomer.Size = New System.Drawing.Size(88, 72)
Me.btnAddCustomer.TabIndex = 10
Me.btnAddCustomer.Text = "Add New Customer"
,
'btnUpdateCust
,
Me.btnUpdateCust.Location = New System.Drawing.Point(168, 48)
Me.btnUpdateCust.Name = "btnUpdateCust"
Me.btnUpdateCust.Size = New System.Drawing.Size(80, 72)
Me.btnUpdateCust.TabIndex = 9
Me.btnUpdateCust.Text = "Update Account"
,
'GroupBox2
,
Me.GroupBox2.Controls.Add(Me.btnSearchEvents)
Me.GroupBox2.Controls.Add(Me.cmdAddEvent)
Me.GroupBox2.Controls.Add(Me.cmdUpdateEvents)
Me.GroupBox2.Location = New System.Drawing.Point(48, 360)
Me.GroupBox2.Name = "GroupBox2"
Me.GroupBox2.Size = New System.Drawing.Size(432, 144)
Me.GroupBox2.TabIndex = 43
Me.GroupBox2.TabStop = False
Me.GroupBox2.Text = "Event Information"
,
'btnSearchEvents
,
Me.btnSearchEvents.Location = New System.Drawing.Point(304, 32)
Me.btnSearchEvents.Name = "btnSearchEvents"
Me.btnSearchEvents.Size = New System.Drawing.Size(88, 80)
Me.btnSearchEvents.TabIndex = 62
Me.btnSearchEvents.Text = "Search Event List"
,
'cmdAddEvent
,
Me.cmdAddEvent.Location = New System.Drawing.Point(40, 32)
Me.cmdAddEvent.Name = "cmdAddEvent"
Me.cmdAddEvent.Size = New System.Drawing.Size(88, 80)
Me.cmdAddEvent.TabIndex = 61
Me.cmdAddEvent.Text = "Add New Event"
,
'cmdUpdateEvents
,
Me.cmdUpdateEvents.Location = New System.Drawing.Point(168, 32)
Me.cmdUpdateEvents.Name = "cmdUpdateEvents"
Me.cmdUpdateEvents.Size = New System.Drawing.Size(88, 80)
Me.cmdUpdateEvents.TabIndex = 60
Me.cmdUpdateEvents.Text = "Update Event"
,
'btnExit
,
Me.btnExit.BackColor = System.Drawing.Color.Red
Me.btnExit.Location = New System.Drawing.Point(128, 520)
Me.btnExit.Name = "btnExit"

```



```

Me.btnExit.Size = New System.Drawing.Size(104, 32)
Me.btnExit.TabIndex = 61
Me.btnExit.Text = "Exit Program"
'
'BtnAbout
'
Me.BtnAbout.BackColor = System.Drawing.Color.Yellow
Me.BtnAbout.Location = New System.Drawing.Point(336, 520)
Me.BtnAbout.Name = "BtnAbout"
Me.BtnAbout.Size = New System.Drawing.Size(96, 32)
Me.BtnAbout.TabIndex = 62
Me.BtnAbout.Text = "About Program"
'
'frmCaterInfoSys
'
Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
Me.ClientSize = New System.Drawing.Size(528, 630)
Me.Controls.Add(Me.BtnAbout)
Me.Controls.Add(Me.btnExit)
Me.Controls.Add(Me.lblCaterLabel)
Me.Controls.Add(Me.PictureBox1)
Me.Controls.Add(Me.GroupBox1)
Me.Controls.Add(Me.GroupBox2)
Me.Name = "frmCaterInfoSys"
Me.Text = "Catering Information System"
Me.GroupBox1.ResumeLayout(False)
Me.GroupBox2.ResumeLayout(False)
Me.ResumeLayout(False)

```

End Sub

#End Region

```

'*****
'
'   Form Name:   frmCaterInfoSystem
'
'   Purpose:     This form is the startup form for the program.
'               This form allows users to choose what they want
'               to do with the customer and event databases,
'               such as add, update, or search.
'
'   Updated:     November 10, 2004
'   Updated By:  Matthew D. Johnson
'
'*****

```

```

Private Sub btnAddCustomer_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnAddCustomer.Click
'Procedure that is called when the user clicks on the Add
Customer button.
'Clicking on this will call the Add Customer form.
Dim objFrmAddCust As New frmAddCust
objFrmAddCust.Show()

```

End Sub

```
Private Sub cmdUpdateEvents_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles cmdUpdateEvents.Click
    'Procedure that is called when the user clicks on the Update
Event button.
    'Clicking on this will call the Event Information form.
    Dim objFrmEventInfo As New frmEventInfo
    objFrmEventInfo.Show()

```

```
End Sub
```

```
Private Sub btnUpdateCust_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnUpdateCust.Click
    'Procedure that is called when the user clicks on the Update
Customer button.
    'Clicking on this will call the Customer Information form.
    Dim objFrmCustInfo As New frmCustInfo
    objFrmCustInfo.Show()

```

```
End Sub
```

```
Private Sub cmdAddEvent_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles cmdAddEvent.Click
    'Procedure that is called when the user clicks on the Add Event
button.
    'Clicking on this will call the Add Event form.
    Dim objfrmAddEvent As New frmAddEvent
    objfrmAddEvent.Show()

```

```
End Sub
```

```
Private Sub btnExit_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExit.Click
    'This procedure is called when the user clicks on the Exit
button.
    'Calling this procedure will close the program.
    Close()

```

```
End Sub
```

```
Private Sub BtnAbout_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles BtnAbout.Click
    'This procedure is called when the user clicks on the About
Program button.
    'This displays the About form containing informaiton about the
program and the programmer.
    Dim objFrmAbout As New frmAbout
    objFrmAbout.Show()

```

```
End Sub
```

```
Private Sub btnSearchCust_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles btnSearchCust.Click
    'Procedure that is called when the user clicks on the Search
Customer button.
    'Clicking on this will call the Search Customer form.

```

```
        Dim objfrmSearchCust As New frmSearchCust
        objfrmSearchCust.Show()
    End Sub

    Private Sub btnSearchEvents_Click(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles btnSearchEvents.Click
        'Procedure that is called when the user clicks on the Search
    Events button.
        'Clicking on this will call the Search Events form.
        Dim objfrmSearchEvent As New frmSearchEvent
        objfrmSearchEvent.Show()
    End Sub
End Class
```

Appendix G - Customer Information Form Code

```
Public Class frmCustInfo
    Inherits System.Windows.Forms.Form

    '*****
    '
    ' Form Name: frmCustInfo
    '
    ' Purpose: This form is used to review and update the information
    ' about an customer. It is made up of several different
    ' text boxes and labels. It also contains controls
    ' for navigating through the records.
    '
    ' Updated: November 10, 2004
    ' Updated By: Matthew D. Johnson
    '
    '*****

#Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overrides Sub Dispose(ByVal disposing As
Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form
Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents txtCustCompany As System.Windows.Forms.TextBox
    Friend WithEvents lblCustCompany As System.Windows.Forms.Label
    Friend WithEvents txtCustZip As System.Windows.Forms.TextBox
    Friend WithEvents lblCustZip As System.Windows.Forms.Label
    Friend WithEvents txtCustState As System.Windows.Forms.TextBox
    Friend WithEvents lblCustState As System.Windows.Forms.Label

```

```

Friend WithEvents txtCustCity As System.Windows.Forms.TextBox
Friend WithEvents lblCustCity As System.Windows.Forms.Label
Friend WithEvents txtCustAddress As System.Windows.Forms.TextBox
Friend WithEvents lblCustAddress As System.Windows.Forms.Label
Friend WithEvents txtCustLastName As System.Windows.Forms.TextBox
Friend WithEvents lblCustLastName As System.Windows.Forms.Label
Friend WithEvents txtCustFirstName As System.Windows.Forms.TextBox
Friend WithEvents lblCustFirstName As System.Windows.Forms.Label
Friend WithEvents txtCustPhone2 As System.Windows.Forms.TextBox
Friend WithEvents txtCustPhone1 As System.Windows.Forms.TextBox
Friend WithEvents txtCustNum As System.Windows.Forms.TextBox
Friend WithEvents lblPhone2 As System.Windows.Forms.Label
Friend WithEvents lblPhone1 As System.Windows.Forms.Label
Friend WithEvents lblCustNumber As System.Windows.Forms.Label
Friend WithEvents GroupBox1 As System.Windows.Forms.GroupBox
Friend WithEvents Label1 As System.Windows.Forms.Label
Friend WithEvents OleDbDataAdapter1 As
System.Data.OleDb.OleDbDataAdapter
Friend WithEvents OleDbSelectCommand1 As
System.Data.OleDb.OleDbCommand
Friend WithEvents OleDbUpdateCommand1 As
System.Data.OleDb.OleDbCommand
Friend WithEvents OleDbDeleteCommand1 As
System.Data.OleDb.OleDbCommand
Friend WithEvents OleDbConnection1 As
System.Data.OleDb.OleDbConnection
Friend WithEvents DsClients1 As WindowsApplication1.DsClients
Friend WithEvents btnPrevious As System.Windows.Forms.Button
Friend WithEvents btnNext As System.Windows.Forms.Button
Friend WithEvents btnLast As System.Windows.Forms.Button
Friend WithEvents btnFirst As System.Windows.Forms.Button
Friend WithEvents btnNew As System.Windows.Forms.Button
Friend WithEvents btnClose As System.Windows.Forms.Button
Friend WithEvents btnUpdate As System.Windows.Forms.Button
Friend WithEvents btnCopy As System.Windows.Forms.Button
Public WithEvents OleDbInsertCommand1 As
System.Data.OleDb.OleDbCommand
<System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()
Me.txtCustCompany = New System.Windows.Forms.TextBox
Me.DsClients1 = New WindowsApplication1.DsClients
Me.lblCustCompany = New System.Windows.Forms.Label
Me.txtCustZip = New System.Windows.Forms.TextBox
Me.lblCustZip = New System.Windows.Forms.Label
Me.txtCustState = New System.Windows.Forms.TextBox
Me.lblCustState = New System.Windows.Forms.Label
Me.txtCustCity = New System.Windows.Forms.TextBox
Me.lblCustCity = New System.Windows.Forms.Label
Me.txtCustAddress = New System.Windows.Forms.TextBox
Me.lblCustAddress = New System.Windows.Forms.Label
Me.txtCustLastName = New System.Windows.Forms.TextBox
Me.lblCustLastName = New System.Windows.Forms.Label
Me.txtCustFirstName = New System.Windows.Forms.TextBox
Me.lblCustFirstName = New System.Windows.Forms.Label
Me.txtCustPhone2 = New System.Windows.Forms.TextBox
Me.txtCustPhone1 = New System.Windows.Forms.TextBox
Me.txtCustNum = New System.Windows.Forms.TextBox

```

```

Me.lblPhone2 = New System.Windows.Forms.Label
Me.lblPhone1 = New System.Windows.Forms.Label
Me.lblCustNumber = New System.Windows.Forms.Label
Me.GroupBox1 = New System.Windows.Forms.GroupBox
Me.btnClose = New System.Windows.Forms.Button
Me.Label1 = New System.Windows.Forms.Label
Me.OleDbDataAdapter1 = New System.Data.OleDb.OleDbDataAdapter
Me.OleDbDeleteCommand1 = New System.Data.OleDb.OleDbCommand
Me.OleDbConnection1 = New System.Data.OleDb.OleDbConnection
Me.OleDbInsertCommand1 = New System.Data.OleDb.OleDbCommand
Me.OleDbSelectCommand1 = New System.Data.OleDb.OleDbCommand
Me.OleDbUpdateCommand1 = New System.Data.OleDb.OleDbCommand
Me.btnPrevious = New System.Windows.Forms.Button
Me.btnNext = New System.Windows.Forms.Button
Me.btnLast = New System.Windows.Forms.Button
Me.btnFirst = New System.Windows.Forms.Button
Me.btnNew = New System.Windows.Forms.Button
Me.btnUpdate = New System.Windows.Forms.Button
Me.btnCopy = New System.Windows.Forms.Button
CType(Me.DsClients1,
System.ComponentModel.ISupportInitialize).BeginInit()
Me.SuspendLayout()
'
'txtCustCompany
'
Me.txtCustCompany.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsClients1,
"Customers.CompanyOrDepartment"))
Me.txtCustCompany.Location = New System.Drawing.Point(296, 104)
Me.txtCustCompany.Name = "txtCustCompany"
Me.txtCustCompany.Size = New System.Drawing.Size(408, 20)
Me.txtCustCompany.TabIndex = 42
Me.txtCustCompany.Text = ""
'
'DsClients1
'
Me.DsClients1.DataSetName = "DsClients"
Me.DsClients1.Locale = New
System.Globalization.CultureInfo("en-US")
'
'lblCustCompany
'
Me.lblCustCompany.Location = New System.Drawing.Point(216, 104)
Me.lblCustCompany.Name = "lblCustCompany"
Me.lblCustCompany.Size = New System.Drawing.Size(72, 24)
Me.lblCustCompany.TabIndex = 41
Me.lblCustCompany.Text = "Company"
Me.lblCustCompany.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
'
'txtCustZip
'
Me.txtCustZip.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsClients1,
"Customers.PostalCode"))
Me.txtCustZip.Location = New System.Drawing.Point(592, 184)
Me.txtCustZip.Name = "txtCustZip"

```

```

Me.txtCustZip.Size = New System.Drawing.Size(112, 20)
Me.txtCustZip.TabIndex = 40
Me.txtCustZip.Text = ""
,
'lblCustZip
,
Me.lblCustZip.Location = New System.Drawing.Point(536, 184)
Me.lblCustZip.Name = "lblCustZip"
Me.lblCustZip.Size = New System.Drawing.Size(48, 24)
Me.lblCustZip.TabIndex = 39
Me.lblCustZip.Text = "Zip"
Me.lblCustZip.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
,
'txtCustState
,
Me.txtCustState.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsClients1,
"Customers.StateOrProvince"))
Me.txtCustState.Location = New System.Drawing.Point(504, 184)
Me.txtCustState.Name = "txtCustState"
Me.txtCustState.Size = New System.Drawing.Size(24, 20)
Me.txtCustState.TabIndex = 38
Me.txtCustState.Text = ""
,
'lblCustState
,
Me.lblCustState.Location = New System.Drawing.Point(440, 184)
Me.lblCustState.Name = "lblCustState"
Me.lblCustState.Size = New System.Drawing.Size(56, 24)
Me.lblCustState.TabIndex = 37
Me.lblCustState.Text = "State"
Me.lblCustState.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
,
'txtCustCity
,
Me.txtCustCity.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsClients1, "Customers.City"))
Me.txtCustCity.Location = New System.Drawing.Point(296, 184)
Me.txtCustCity.Name = "txtCustCity"
Me.txtCustCity.Size = New System.Drawing.Size(128, 20)
Me.txtCustCity.TabIndex = 36
Me.txtCustCity.Text = ""
,
'lblCustCity
,
Me.lblCustCity.Location = New System.Drawing.Point(216, 184)
Me.lblCustCity.Name = "lblCustCity"
Me.lblCustCity.Size = New System.Drawing.Size(72, 24)
Me.lblCustCity.TabIndex = 35
Me.lblCustCity.Text = "City"
Me.lblCustCity.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
,
'txtCustAddress
,

```

```

        Me.txtCustAddress.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsClients1,
"Customers.BillingAddress"))
        Me.txtCustAddress.Location = New System.Drawing.Point(296, 144)
        Me.txtCustAddress.Name = "txtCustAddress"
        Me.txtCustAddress.Size = New System.Drawing.Size(408, 20)
        Me.txtCustAddress.TabIndex = 34
        Me.txtCustAddress.Text = ""
        '
        'lblCustAddress
        '
        Me.lblCustAddress.Location = New System.Drawing.Point(216, 144)
        Me.lblCustAddress.Name = "lblCustAddress"
        Me.lblCustAddress.Size = New System.Drawing.Size(72, 24)
        Me.lblCustAddress.TabIndex = 33
        Me.lblCustAddress.Text = "Address"
        Me.lblCustAddress.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
        '
        'txtCustLastName
        '
        Me.txtCustLastName.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsClients1,
"Customers.ContactLastName"))
        Me.txtCustLastName.Location = New System.Drawing.Point(528, 72)
        Me.txtCustLastName.Name = "txtCustLastName"
        Me.txtCustLastName.Size = New System.Drawing.Size(176, 20)
        Me.txtCustLastName.TabIndex = 32
        Me.txtCustLastName.Text = ""
        '
        'lblCustLastName
        '
        Me.lblCustLastName.Location = New System.Drawing.Point(432, 72)
        Me.lblCustLastName.Name = "lblCustLastName"
        Me.lblCustLastName.Size = New System.Drawing.Size(88, 24)
        Me.lblCustLastName.TabIndex = 31
        Me.lblCustLastName.Text = "Last Name"
        Me.lblCustLastName.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
        '
        'txtCustFirstName
        '
        Me.txtCustFirstName.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsClients1,
"Customers.ContactFirstName"))
        Me.txtCustFirstName.Location = New System.Drawing.Point(296,
72)
        Me.txtCustFirstName.Name = "txtCustFirstName"
        Me.txtCustFirstName.Size = New System.Drawing.Size(128, 20)
        Me.txtCustFirstName.TabIndex = 30
        Me.txtCustFirstName.Text = ""
        '
        'lblCustFirstName
        '
        Me.lblCustFirstName.Location = New System.Drawing.Point(216,
72)
        Me.lblCustFirstName.Name = "lblCustFirstName"

```



```

Me.lblCustFirstName.Size = New System.Drawing.Size(72, 24)
Me.lblCustFirstName.TabIndex = 29
Me.lblCustFirstName.Text = "First Name"
Me.lblCustFirstName.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
,
    'txtCustPhone2
,
    Me.txtCustPhone2.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsClients1,
"Customers.CellularPhone"))
    Me.txtCustPhone2.Location = New System.Drawing.Point(96, 184)
    Me.txtCustPhone2.Name = "txtCustPhone2"
    Me.txtCustPhone2.Size = New System.Drawing.Size(96, 20)
    Me.txtCustPhone2.TabIndex = 28
    Me.txtCustPhone2.Text = ""
,
    'txtCustPhone1
,
    Me.txtCustPhone1.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsClients1,
"Customers.BusPhone"))
    Me.txtCustPhone1.Location = New System.Drawing.Point(96, 144)
    Me.txtCustPhone1.Name = "txtCustPhone1"
    Me.txtCustPhone1.Size = New System.Drawing.Size(96, 20)
    Me.txtCustPhone1.TabIndex = 27
    Me.txtCustPhone1.Text = ""
,
    'txtCustNum
,
    Me.txtCustNum.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsClients1,
"Customers.CustomerID"))
    Me.txtCustNum.Location = New System.Drawing.Point(96, 72)
    Me.txtCustNum.Name = "txtCustNum"
    Me.txtCustNum.Size = New System.Drawing.Size(96, 20)
    Me.txtCustNum.TabIndex = 26
    Me.txtCustNum.Text = ""
,
    'lblPhone2
,
    Me.lblPhone2.Location = New System.Drawing.Point(24, 184)
    Me.lblPhone2.Name = "lblPhone2"
    Me.lblPhone2.Size = New System.Drawing.Size(64, 24)
    Me.lblPhone2.TabIndex = 25
    Me.lblPhone2.Text = "Phone # 2"
,
    'lblPhone1
,
    Me.lblPhone1.Location = New System.Drawing.Point(24, 144)
    Me.lblPhone1.Name = "lblPhone1"
    Me.lblPhone1.Size = New System.Drawing.Size(64, 24)
    Me.lblPhone1.TabIndex = 24
    Me.lblPhone1.Text = "Phone # 1"
,
    'lblCustNumber
,

```

```

Me.lblCustNumber.Location = New System.Drawing.Point(24, 72)
Me.lblCustNumber.Name = "lblCustNumber"
Me.lblCustNumber.Size = New System.Drawing.Size(64, 16)
Me.lblCustNumber.TabIndex = 23
Me.lblCustNumber.Text = "Customer #"
,
'GroupBox1
,
Me.GroupBox1.Location = New System.Drawing.Point(8, 56)
Me.GroupBox1.Name = "GroupBox1"
Me.GroupBox1.Size = New System.Drawing.Size(720, 160)
Me.GroupBox1.TabIndex = 43
Me.GroupBox1.TabStop = False
Me.GroupBox1.Text = "Customer Information"
,
'btnClose
,
Me.btnClose.BackColor = System.Drawing.Color.Red
Me.btnClose.Font = New System.Drawing.Font("Microsoft Sans
Serif", 9.75!, System.Drawing.FontStyle.Regular,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.btnClose.Location = New System.Drawing.Point(648, 16)
Me.btnClose.Name = "btnClose"
Me.btnClose.Size = New System.Drawing.Size(80, 32)
Me.btnClose.TabIndex = 45
Me.btnClose.Text = "Close"
,
'Label1
,
Me.Label1.Font = New System.Drawing.Font("Microsoft Sans
Serif", 14.25!, System.Drawing.FontStyle.Regular,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
Me.Label1.Location = New System.Drawing.Point(32, 8)
Me.Label1.Name = "Label1"
Me.Label1.Size = New System.Drawing.Size(384, 32)
Me.Label1.TabIndex = 46
Me.Label1.Text = "Customer Information"
,
'OleDbDataAdapter1
,
Me.OleDbDataAdapter1.DeleteCommand = Me.OleDbDeleteCommand1
Me.OleDbDataAdapter1.InsertCommand = Me.OleDbInsertCommand1
Me.OleDbDataAdapter1.SelectCommand = Me.OleDbSelectCommand1
Me.OleDbDataAdapter1.TableMappings.AddRange(New
System.Data.Common.DataTableMapping() {New
System.Data.Common.DataTableMapping("Table", "Customers", New
System.Data.Common.DataColumnMapping() {New
System.Data.Common.DataColumnMapping("CustomerID", "CustomerID"), New
System.Data.Common.DataColumnMapping("ContactFirstName",
"ContactFirstName"), New
System.Data.Common.DataColumnMapping("ContactLastName",
"ContactLastName"), New
System.Data.Common.DataColumnMapping("CompanyOrDepartment",
"CompanyOrDepartment"), New
System.Data.Common.DataColumnMapping("CellularPhone", "CellularPhone"),
New System.Data.Common.DataColumnMapping("BusPhone", "BusPhone"), New
System.Data.Common.DataColumnMapping("BillingAddress",

```

```

"BillingAddress"), New System.Data.Common.DataColumnMapping("City",
"City"), New System.Data.Common.DataColumnMapping("StateOrProvince",
"StateOrProvince"), New
System.Data.Common.DataColumnMapping("PostalCode", "PostalCode"))}}}
    Me.OleDbDataAdapter1.UpdateCommand = Me.OleDbUpdateCommand1
    ,
    'OleDbDeleteCommand1
    ,
    Me.OleDbDeleteCommand1.CommandText = "DELETE FROM Customers
WHERE (CustomerID = ?) AND (BillingAddress = ? OR ? IS NULL" & _
    " AND BillingAddress IS NULL) AND (BusPhone = ? OR ? IS NULL
AND BusPhone IS NULL" & _
    ") AND (CellularPhone = ? OR ? IS NULL AND CellularPhone IS
NULL) AND (City = ? OR ? IS NULL AND City IS NULL) AND (CompanyOrDepartment = ? OR
? IS NULL AND Comp" & _
    "anyOrDepartment IS NULL) AND (ContactFirstName = ? OR ? IS
NULL AND ContactFirst" & _
    "Name IS NULL) AND (ContactLastName = ? OR ? IS NULL AND
ContactLastName IS NULL)" & _
    " AND (PostalCode = ? OR ? IS NULL AND PostalCode IS NULL) AND
(StateOrProvince =" & _
    " ? OR ? IS NULL AND StateOrProvince IS NULL)"
    Me.OleDbDeleteCommand1.Connection = Me.OleDbConnection1
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_BillingAddress",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "BillingAddress", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_BillingAddress1",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "BillingAddress", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_BusPhone",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "BusPhone", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_BusPhone1",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "BusPhone", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CellularPhone",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CellularPhone", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CellularPhone1",
System.Data.OleDb.OleDbType.VarWChar, 30,

```

```

System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CellularPhone", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_City",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "City", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_City1",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "City", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CompanyOrDepartment",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CompanyOrDepartment", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CompanyOrDepartment1",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CompanyOrDepartment", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactFirstName",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactFirstName", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactFirstName1",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactFirstName", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactLastName",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactLastName", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactLastName1",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactLastName", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_PostalCode",
System.Data.OleDb.OleDbType.VarWChar, 20,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "PostalCode", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_PostalCode1",
System.Data.OleDb.OleDbType.VarWChar, 20,

```

```

System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "PostalCode", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_StateOrProvince",
System.Data.OleDb.OleDbType.VarWChar, 20,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "StateOrProvince", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_StateOrProvince1",
System.Data.OleDb.OleDbType.VarWChar, 20,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "StateOrProvince", System.Data.DataRowVersion.Original,
Nothing))
    ,
    'OleDbConnection1
    ,
    Me.OleDbConnection1.ConnectionString = "Jet OLEDB:Global
Partial Bulk Ops=2;Jet OLEDB:Registry Path=;Jet OLEDB:Database L" & _
"ocking Mode=1;Jet OLEDB:Database Password=;Data
Source=""C:\catering\Client.mdb"";" & _
"Password=;Jet OLEDB:Engine Type=5;Jet OLEDB:Global Bulk
Transactions=1;Provider=" & _
""Microsoft.Jet.OLEDB.4.0"";Jet OLEDB:System database=;Jet
OLEDB:SFP=False;Extende" & _
"d Properties=;Mode=Share Deny None;Jet OLEDB:New Database
Password=;Jet OLEDB:Cr" & _
"eate System Database=False;Jet OLEDB:Don't Copy Locale on
Compact=False;Jet OLEDB" & _
"B:Compact Without Replica Repair=False;User ID=Admin;Jet
OLEDB:Encrypt Database=" & _
"False"
    ,
    'OleDbInsertCommand1
    ,
    Me.OleDbInsertCommand1.CommandText = "INSERT INTO Customers
(CustomerID, ContactFirstName, ContactLastName, CompanyOrDe" & _
"partment, CellularPhone, BusPhone, BillingAddress, City,
StateOrProvince, Postal" & _
"Code) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?)"
    Me.OleDbInsertCommand1.Connection = Me.OleDbConnection1
    Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("ContactFirstName",
System.Data.OleDb.OleDbType.VarWChar, 30, "ContactFirstName"))
    Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("ContactLastName",
System.Data.OleDb.OleDbType.VarWChar, 50, "ContactLastName"))
    Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("CompanyOrDepartment",
System.Data.OleDb.OleDbType.VarWChar, 50, "CompanyOrDepartment"))
    Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("CellularPhone",
System.Data.OleDb.OleDbType.VarWChar, 30, "CellularPhone"))
    Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("BusPhone",
System.Data.OleDb.OleDbType.VarWChar, 30, "BusPhone"))

```

```

        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("BillingAddress",
System.Data.OleDb.OleDbType.VarWChar, 255, "BillingAddress"))
        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("City",
System.Data.OleDb.OleDbType.VarWChar, 50, "City"))
        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("StateOrProvince",
System.Data.OleDb.OleDbType.VarWChar, 20, "StateOrProvince"))
        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("PostalCode",
System.Data.OleDb.OleDbType.VarWChar, 20, "PostalCode"))
    ,
    'OleDbSelectCommand1
    ,
        Me.OleDbSelectCommand1.CommandText = "SELECT CustomerID,
ContactFirstName, ContactLastName, CompanyOrDepartment, Cellul" & _
        "arPhone, BusPhone, BillingAddress, City, StateOrProvince,
PostalCode FROM Custom" & _
        "ers"
        Me.OleDbSelectCommand1.Connection = Me.OleDbConnection1
    ,
    'OleDbUpdateCommand1
    ,
        Me.OleDbUpdateCommand1.CommandText = "UPDATE Customers SET
ContactFirstName = ?, ContactLastName = ?, CompanyOrDepartme" & _
        "nt = ?, CellularPhone = ?, BusPhone = ?, BillingAddress = ?,
City = ?, StateOrPr" & _
        "ovince = ?, PostalCode = ? WHERE (CustomerID = ?) AND
(BillingAddress = ? OR ? I" & _
        "S NULL AND BillingAddress IS NULL) AND (BusPhone = ? OR ? IS
NULL AND BusPhone I" & _
        "S NULL) AND (CellularPhone = ? OR ? IS NULL AND CellularPhone
IS NULL) AND (City" & _
        " = ? OR ? IS NULL AND City IS NULL) AND (CompanyOrDepartment =
? OR ? IS NULL AN" & _
        "D CompanyOrDepartment IS NULL) AND (ContactFirstName = ? OR ?
IS NULL AND Contac" & _
        "tFirstName IS NULL) AND (ContactLastName = ? OR ? IS NULL AND
ContactLastName IS" & _
        " NULL) AND (PostalCode = ? OR ? IS NULL AND PostalCode IS
NULL) AND (StateOrProv" & _
        "ince = ? OR ? IS NULL AND StateOrProvince IS NULL)"
        Me.OleDbUpdateCommand1.Connection = Me.OleDbConnection1
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("ContactFirstName",
System.Data.OleDb.OleDbType.VarWChar, 30, "ContactFirstName"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("ContactLastName",
System.Data.OleDb.OleDbType.VarWChar, 50, "ContactLastName"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("CompanyOrDepartment",
System.Data.OleDb.OleDbType.VarWChar, 50, "CompanyOrDepartment"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("CellularPhone",
System.Data.OleDb.OleDbType.VarWChar, 30, "CellularPhone"))

```

```

        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("BusPhone",
System.Data.OleDb.OleDbType.VarWChar, 30, "BusPhone"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("BillingAddress",
System.Data.OleDb.OleDbType.VarWChar, 255, "BillingAddress"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("City",
System.Data.OleDb.OleDbType.VarWChar, 50, "City"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("StateOrProvince",
System.Data.OleDb.OleDbType.VarWChar, 20, "StateOrProvince"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("PostalCode",
System.Data.OleDb.OleDbType.VarWChar, 20, "PostalCode"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_BillingAddress",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "BillingAddress", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_BillingAddress1",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "BillingAddress", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_BusPhone",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "BusPhone", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_BusPhone1",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "BusPhone", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CellularPhone",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CellularPhone", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CellularPhone1",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CellularPhone", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_City",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "City", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_City1",

```



```

System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "City", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CompanyOrDepartment",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CompanyOrDepartment", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CompanyOrDepartment1",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CompanyOrDepartment", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactFirstName",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactFirstName", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactFirstName1",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactFirstName", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactLastName",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactLastName", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactLastName1",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactLastName", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_PostalCode",
System.Data.OleDb.OleDbType.VarWChar, 20,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "PostalCode", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_PostalCode1",
System.Data.OleDb.OleDbType.VarWChar, 20,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "PostalCode", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_StateOrProvince",
System.Data.OleDb.OleDbType.VarWChar, 20,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "StateOrProvince", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_StateOrProvince1",

```



```

System.Data.OleDb.OleDbType.VarWChar, 20,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "StateOrProvince", System.Data.DataRowVersion.Original,
Nothing))
    ,
    'btnPrevious
    ,
    Me.btnPrevious.BackColor = System.Drawing.Color.Aqua
    Me.btnPrevious.Location = New System.Drawing.Point(88, 240)
    Me.btnPrevious.Name = "btnPrevious"
    Me.btnPrevious.Size = New System.Drawing.Size(72, 32)
    Me.btnPrevious.TabIndex = 48
    Me.btnPrevious.Text = "Previous"
    ,
    'btnNext
    ,
    Me.btnNext.BackColor = System.Drawing.Color.Aqua
    Me.btnNext.Location = New System.Drawing.Point(168, 240)
    Me.btnNext.Name = "btnNext"
    Me.btnNext.Size = New System.Drawing.Size(80, 32)
    Me.btnNext.TabIndex = 49
    Me.btnNext.Text = "Next"
    ,
    'btnLast
    ,
    Me.btnLast.BackColor = System.Drawing.Color.Lime
    Me.btnLast.Location = New System.Drawing.Point(256, 240)
    Me.btnLast.Name = "btnLast"
    Me.btnLast.Size = New System.Drawing.Size(72, 32)
    Me.btnLast.TabIndex = 50
    Me.btnLast.Text = "Last"
    ,
    'btnFirst
    ,
    Me.btnFirst.BackColor = System.Drawing.Color.Lime
    Me.btnFirst.Location = New System.Drawing.Point(8, 240)
    Me.btnFirst.Name = "btnFirst"
    Me.btnFirst.Size = New System.Drawing.Size(72, 32)
    Me.btnFirst.TabIndex = 51
    Me.btnFirst.Text = "First"
    ,
    'btnNew
    ,
    Me.btnNew.BackColor = System.Drawing.Color.Yellow
    Me.btnNew.Location = New System.Drawing.Point(432, 240)
    Me.btnNew.Name = "btnNew"
    Me.btnNew.Size = New System.Drawing.Size(72, 32)
    Me.btnNew.TabIndex = 52
    Me.btnNew.Text = "New"
    Me.btnNew.Visible = False
    ,
    'btnUpdate
    ,
    Me.btnUpdate.BackColor = System.Drawing.Color.Yellow
    Me.btnUpdate.Location = New System.Drawing.Point(352, 240)
    Me.btnUpdate.Name = "btnUpdate"
    Me.btnUpdate.Size = New System.Drawing.Size(72, 32)

```

```

Me.btnUpdate.TabIndex = 53
Me.btnUpdate.Text = "Update Records"
'
'btnCopy
'
Me.btnCopy.BackColor = System.Drawing.Color.Purple
Me.btnCopy.ForeColor =
System.Drawing.SystemColors.HighlightText
Me.btnCopy.Location = New System.Drawing.Point(544, 240)
Me.btnCopy.Name = "btnCopy"
Me.btnCopy.Size = New System.Drawing.Size(96, 32)
Me.btnCopy.TabIndex = 54
Me.btnCopy.Text = "Copy Customer Information"
Me.btnCopy.Visible = False
'
'frmCustInfo
'
Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
Me.ClientSize = New System.Drawing.Size(752, 334)
Me.Controls.Add(Me.btnCopy)
Me.Controls.Add(Me.btnUpdate)
Me.Controls.Add(Me.btnNew)
Me.Controls.Add(Me.btnFirst)
Me.Controls.Add(Me.btnLast)
Me.Controls.Add(Me.btnNext)
Me.Controls.Add(Me.btnPrevious)
Me.Controls.Add(Me.Label1)
Me.Controls.Add(Me.btnClose)
Me.Controls.Add(Me.txtCustCompany)
Me.Controls.Add(Me.lblCustCompany)
Me.Controls.Add(Me.txtCustZip)
Me.Controls.Add(Me.lblCustZip)
Me.Controls.Add(Me.txtCustState)
Me.Controls.Add(Me.lblCustState)
Me.Controls.Add(Me.txtCustCity)
Me.Controls.Add(Me.lblCustCity)
Me.Controls.Add(Me.txtCustAddress)
Me.Controls.Add(Me.lblCustAddress)
Me.Controls.Add(Me.txtCustLastName)
Me.Controls.Add(Me.lblCustLastName)
Me.Controls.Add(Me.txtCustFirstName)
Me.Controls.Add(Me.lblCustFirstName)
Me.Controls.Add(Me.txtCustPhone2)
Me.Controls.Add(Me.txtCustPhone1)
Me.Controls.Add(Me.txtCustNum)
Me.Controls.Add(Me.lblPhone2)
Me.Controls.Add(Me.lblPhone1)
Me.Controls.Add(Me.lblCustNumber)
Me.Controls.Add(Me.GroupBox1)
Me.Name = "frmCustInfo"
Me.Text = "Customer Information"
CType(Me.DsClients1,
System.ComponentModel.ISupportInitialize).EndInit()
Me.ResumeLayout(False)

```

End Sub

#End Region

```
Private Sub frmCustInfo_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    'Command that is called when the form is loaded. It loads the data adapter and fills it.
    DsClients1.Clear()
    OleDbDataAdapter1.Fill(DsClients1)
End Sub
```

```
Private Sub btnClose_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnClose.Click
    'This command is used to close the window when called by the user.
    Close()
End Sub
```

```
Private Sub btnPrevious_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnPrevious.Click
    'Button used in navigating through the data
    Me.BindingContext(DsClients1, "customers").Position -= 1
End Sub
```

```
Private Sub btnNext_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnNext.Click
    'Button used in navigating through the data
    Me.BindingContext(DsClients1, "customers").Position += 1
End Sub
```

```
Private Sub btnFirst_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnFirst.Click
    'Button used in navigating through the data
    Me.BindingContext(DsClients1, "customers").Position = 0
End Sub
```

```
Private Sub btnLast_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnLast.Click
    'Button used in navigating through the data
    Me.BindingContext(DsClients1, "customers").Position = Me.BindingContext(DsClients1, "customers").Count - 1
End Sub
```

```
*****
*****
```

```

,
,
'*****NOT USED PROCEDURE MATERIAL***SLOTTED FOR FUTURE
UPDATES*****
```

```
Private Sub btnNew_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnNew.Click
```

```
    'Me.BindingContext(DsClients1, "customers").Position = Me.BindingContext(DsClients1, "customers").Count
```

```

'DsClients1.Customers.NewRow(
'OleDbDataAdapter1.InsertCommand()
'DsClients1.Customers.AddCustomersRow("John", "Dow", "", "", "",
"", "", "", "")
'Dim current As Short
'current = Me.BindingContext(DsClients1, "Customers").Count
'Me.BindingContext(DsClients1, "Customers").Position = 0
'Me.BindingContext(DsClients1, "Customers").Position = current
'OleDbDataAdapter1.Update(DsClients1)
'OleDbDataAdapter1.Update(DsClients1)

'INSERT INTO Customers(ContactFirstName, ContactLastName,
CompanyOrDepartment, CellularPhone, BusPhone, BillingAddress, City,
StateOrProvince, PostalCode) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?)

'MsgBox("New Customer Added", MsgBoxStyle.Information, "New Clients
Confirmation")
'End Sub
'
'

!*****
*****

Private Sub btnUpdate_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnUpdate.Click
'This procedure is called when the user clicks on the Update
Button.
'The procedure updates the database and displays a confirmation
for the users.
Dim Current As Short
Current = Me.BindingContext(DsClients1, "Customers").Position
Me.BindingContext(DsClients1, "Customers").Position = 0
Me.BindingContext(DsClients1, "Customers").Position = Current
OleDbDataAdapter1.Update(DsClients1)
MsgBox("Update Complete", MsgBoxStyle.Information, "Update
Confirmation")
End Sub

!*****
*****

'
'
'*****NOT USED PROCEDURE MATERIAL***SLOTTED FOR FUTURE
UPDATES*****
'
'Private Sub btnCopy_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnCopy.Click
' Dim objfrmCaterInfoSys As frmCaterInfoSys
' Dim txtCustNum As txtCustNum
' frmCaterInfoSys.

'End Sub
'

```

! *****

End Class

Appendix H - Event Information Form Code

```
Public Class frmEventInfo
    Inherits System.Windows.Forms.Form

#Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overrides Sub Dispose(ByVal disposing As
Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form
Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents txtHeadCount As System.Windows.Forms.TextBox
    Friend WithEvents txtLocationZip As System.Windows.Forms.TextBox
    Friend WithEvents lblLocationZip As System.Windows.Forms.Label
    Friend WithEvents txtLocationState As System.Windows.Forms.TextBox
    Friend WithEvents lblLocationState As System.Windows.Forms.Label
    Friend WithEvents txtLocationCity As System.Windows.Forms.TextBox
    Friend WithEvents lblLocationCity As System.Windows.Forms.Label
    Friend WithEvents txtLocationAddress As
System.Windows.Forms.TextBox
    Friend WithEvents lblLocationAddress As System.Windows.Forms.Label
    Friend WithEvents txtLocatooinName As System.Windows.Forms.TextBox
    Friend WithEvents lblLocationName As System.Windows.Forms.Label
    Friend WithEvents txtEventNum As System.Windows.Forms.TextBox
    Friend WithEvents groupBox2 As System.Windows.Forms.GroupBox
    Friend WithEvents lblNotes As System.Windows.Forms.Label
    Friend WithEvents txtMenu As System.Windows.Forms.TextBox
    Friend WithEvents lblMenu As System.Windows.Forms.Label
    Friend WithEvents txtSpecialInstructions As
System.Windows.Forms.TextBox
    Friend WithEvents lblSpecialInstructions As
System.Windows.Forms.Label
```

```

Friend WithEvents txtNotes As System.Windows.Forms.TextBox
Friend WithEvents Label1 As System.Windows.Forms.Label
Friend WithEvents txtCheckNum As System.Windows.Forms.TextBox
Friend WithEvents lblCheckNum As System.Windows.Forms.Label
Friend WithEvents txtStoreChargeNum As System.Windows.Forms.TextBox
Friend WithEvents lblStoreChargeNum As System.Windows.Forms.Label
Friend WithEvents txtCCDate As System.Windows.Forms.TextBox
Friend WithEvents lblCCDate As System.Windows.Forms.Label
Friend WithEvents txtCCNum As System.Windows.Forms.TextBox
Friend WithEvents lblCCNum As System.Windows.Forms.Label
Friend WithEvents Label2 As System.Windows.Forms.Label
Friend WithEvents txtTotalCost As System.Windows.Forms.TextBox
Friend WithEvents lblTotalCost As System.Windows.Forms.Label
Friend WithEvents txtAddCosts As System.Windows.Forms.TextBox
Friend WithEvents lblAddCosts As System.Windows.Forms.Label
Friend WithEvents txtCostPerPerson As System.Windows.Forms.TextBox
Friend WithEvents lblCostPerPerson As System.Windows.Forms.Label
Friend WithEvents GroupBox3 As System.Windows.Forms.GroupBox
Friend WithEvents lblHeadCountLabel As System.Windows.Forms.Label
Friend WithEvents lblEventNumLabel As System.Windows.Forms.Label
Friend WithEvents lblTimeLabel As System.Windows.Forms.Label
Friend WithEvents lblDateLabel As System.Windows.Forms.Label
Friend WithEvents txtDate As System.Windows.Forms.TextBox
Friend WithEvents txtTime As System.Windows.Forms.TextBox
Friend WithEvents cmdCancel As System.Windows.Forms.Button
Friend WithEvents txtPaymentType As System.Windows.Forms.TextBox
Friend WithEvents OleDbDataAdapter1 As
System.Data.OleDb.OleDbDataAdapter
    Friend WithEvents OleDbSelectCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbInsertCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbUpdateCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbDeleteCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbConnection1 As
System.Data.OleDb.OleDbConnection
    Friend WithEvents DsEventInfo1 As WindowsApplication1.DsEventInfo
    Friend WithEvents btnNext As System.Windows.Forms.Button
    Friend WithEvents btnPrevious As System.Windows.Forms.Button
    Friend WithEvents btnFirst As System.Windows.Forms.Button
    Friend WithEvents btnLast As System.Windows.Forms.Button
    Friend WithEvents btnNew As System.Windows.Forms.Button
    Friend WithEvents btnUpdate As System.Windows.Forms.Button
    Friend WithEvents lblCustNumLabel As System.Windows.Forms.Label
    Friend WithEvents txtCustNumber As System.Windows.Forms.TextBox
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()
    Me.txtHeadCount = New System.Windows.Forms.TextBox
    Me.DsEventInfo1 = New WindowsApplication1.DsEventInfo
    Me.txtLocationZip = New System.Windows.Forms.TextBox
    Me.lblLocationZip = New System.Windows.Forms.Label
    Me.txtLocationState = New System.Windows.Forms.TextBox
    Me.lblLocationState = New System.Windows.Forms.Label
    Me.txtLocationCity = New System.Windows.Forms.TextBox
    Me.lblLocationCity = New System.Windows.Forms.Label

```

```

Me.txtLocationAddress = New System.Windows.Forms.TextBox
Me.lblLocationAddress = New System.Windows.Forms.Label
Me.txtLocatoinName = New System.Windows.Forms.TextBox
Me.lblLocationName = New System.Windows.Forms.Label
Me.txtEventNum = New System.Windows.Forms.TextBox
Me.GroupBox2 = New System.Windows.Forms.GroupBox
Me.txtCustNumber = New System.Windows.Forms.TextBox
Me.lblCustNumLabel = New System.Windows.Forms.Label
Me.txtTime = New System.Windows.Forms.TextBox
Me.txtDate = New System.Windows.Forms.TextBox
Me.lblHeadCountLabel = New System.Windows.Forms.Label
Me.lblTimeLabel = New System.Windows.Forms.Label
Me.lblDateLabel = New System.Windows.Forms.Label
Me.lblEventNumLabel = New System.Windows.Forms.Label
Me.lblNotes = New System.Windows.Forms.Label
Me.txtMenu = New System.Windows.Forms.TextBox
Me.lblMenu = New System.Windows.Forms.Label
Me.txtSpecialInstructions = New System.Windows.Forms.TextBox
Me.lblSpecialInstructions = New System.Windows.Forms.Label
Me.txtNotes = New System.Windows.Forms.TextBox
Me.Label1 = New System.Windows.Forms.Label
Me.txtCheckNum = New System.Windows.Forms.TextBox
Me.lblCheckNum = New System.Windows.Forms.Label
Me.txtStoreChargeNum = New System.Windows.Forms.TextBox
Me.lblStoreChargeNum = New System.Windows.Forms.Label
Me.txtCCDate = New System.Windows.Forms.TextBox
Me.lblCCDate = New System.Windows.Forms.Label
Me.txtCCNum = New System.Windows.Forms.TextBox
Me.lblCCNum = New System.Windows.Forms.Label
Me.Label2 = New System.Windows.Forms.Label
Me.txtTotalCost = New System.Windows.Forms.TextBox
Me.lblTotalCost = New System.Windows.Forms.Label
Me.txtAddCosts = New System.Windows.Forms.TextBox
Me.lblAddCosts = New System.Windows.Forms.Label
Me.txtCostPerPerson = New System.Windows.Forms.TextBox
Me.lblCostPerPerson = New System.Windows.Forms.Label
Me.GroupBox3 = New System.Windows.Forms.GroupBox
Me.txtPaymentType = New System.Windows.Forms.TextBox
Me.cmdCancel = New System.Windows.Forms.Button
Me.OleDbDataAdapter1 = New System.Data.OleDb.OleDbDataAdapter
Me.OleDbDeleteCommand1 = New System.Data.OleDb.OleDbCommand
Me.OleDbConnection1 = New System.Data.OleDb.OleDbConnection
Me.OleDbInsertCommand1 = New System.Data.OleDb.OleDbCommand
Me.OleDbSelectCommand1 = New System.Data.OleDb.OleDbCommand
Me.OleDbUpdateCommand1 = New System.Data.OleDb.OleDbCommand
Me.btnNext = New System.Windows.Forms.Button
Me.btnPrevious = New System.Windows.Forms.Button
Me.btnFirst = New System.Windows.Forms.Button
Me.btnLast = New System.Windows.Forms.Button
Me.btnNew = New System.Windows.Forms.Button
Me.btnUpdate = New System.Windows.Forms.Button
CType(Me.DsEventInfo1,
System.ComponentModel.ISupportInitialize).BeginInit()
Me.GroupBox2.SuspendLayout()
Me.GroupBox3.SuspendLayout()
Me.SuspendLayout()

```



```

        'txtHeadCount
        ,
        Me.txtHeadCount.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInfo1,
"Events.HeadCount"))
        Me.txtHeadCount.Location = New System.Drawing.Point(272, 120)
        Me.txtHeadCount.Name = "txtHeadCount"
        Me.txtHeadCount.Size = New System.Drawing.Size(72, 20)
        Me.txtHeadCount.TabIndex = 74
        Me.txtHeadCount.Text = ""
        ,
        'DsEventInfo1
        ,
        Me.DsEventInfo1.DataSetName = "DsEventInfo"
        Me.DsEventInfo1.Locale = New
System.Globalization.CultureInfo("en-US")
        ,
        'txtLocationZip
        ,
        Me.txtLocationZip.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInfo1,
"Events.PostalCode"))
        Me.txtLocationZip.Location = New System.Drawing.Point(592, 152)
        Me.txtLocationZip.Name = "txtLocationZip"
        Me.txtLocationZip.Size = New System.Drawing.Size(120, 20)
        Me.txtLocationZip.TabIndex = 73
        Me.txtLocationZip.Text = ""
        ,
        'lblLocationZip
        ,
        Me.lblLocationZip.Location = New System.Drawing.Point(544, 152)
        Me.lblLocationZip.Name = "lblLocationZip"
        Me.lblLocationZip.Size = New System.Drawing.Size(48, 24)
        Me.lblLocationZip.TabIndex = 72
        Me.lblLocationZip.Text = "Zip"
        Me.lblLocationZip.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
        ,
        'txtLocationState
        ,
        Me.txtLocationState.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInfo1,
"Events.StateOrProvince"))
        Me.txtLocationState.Location = New System.Drawing.Point(512,
152)
        Me.txtLocationState.Name = "txtLocationState"
        Me.txtLocationState.Size = New System.Drawing.Size(24, 20)
        Me.txtLocationState.TabIndex = 71
        Me.txtLocationState.Text = ""
        ,
        'lblLocationState
        ,
        Me.lblLocationState.Location = New System.Drawing.Point(448,
152)
        Me.lblLocationState.Name = "lblLocationState"
        Me.lblLocationState.Size = New System.Drawing.Size(56, 24)
        Me.lblLocationState.TabIndex = 70

```

```

        Me.lblLocationState.Text = "State"
        Me.lblLocationState.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
    '
    'txtLocationCity
    '
    Me.txtLocationCity.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInf1, "Events.City"))
    Me.txtLocationCity.Location = New System.Drawing.Point(304,
152)
    Me.txtLocationCity.Name = "txtLocationCity"
    Me.txtLocationCity.Size = New System.Drawing.Size(128, 20)
    Me.txtLocationCity.TabIndex = 69
    Me.txtLocationCity.Text = ""
    '
    'lblLocationCity
    '
    Me.lblLocationCity.Location = New System.Drawing.Point(224,
152)
    Me.lblLocationCity.Name = "lblLocationCity"
    Me.lblLocationCity.Size = New System.Drawing.Size(72, 24)
    Me.lblLocationCity.TabIndex = 68
    Me.lblLocationCity.Text = "City"
    Me.lblLocationCity.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
    '
    'txtLocationAddress
    '
    Me.txtLocationAddress.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInf1,
"Events.LocationAddress"))
    Me.txtLocationAddress.Location = New System.Drawing.Point(304,
120)
    Me.txtLocationAddress.Name = "txtLocationAddress"
    Me.txtLocationAddress.Size = New System.Drawing.Size(408, 20)
    Me.txtLocationAddress.TabIndex = 66
    Me.txtLocationAddress.Text = ""
    '
    'lblLocationAddress
    '
    Me.lblLocationAddress.Location = New System.Drawing.Point(224,
120)
    Me.lblLocationAddress.Name = "lblLocationAddress"
    Me.lblLocationAddress.Size = New System.Drawing.Size(72, 24)
    Me.lblLocationAddress.TabIndex = 65
    Me.lblLocationAddress.Text = "Location Address"
    Me.lblLocationAddress.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
    '
    'txtLocatoinName
    '
    Me.txtLocatoinName.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInf1,
"Events.Location"))
    Me.txtLocatoinName.Location = New System.Drawing.Point(304, 80)
    Me.txtLocatoinName.Name = "txtLocatoinName"
    Me.txtLocatoinName.Size = New System.Drawing.Size(408, 20)

```

```

Me.txtLocatoinName.TabIndex = 63
Me.txtLocatoinName.Text = ""
'
'lblLocationName
'
Me.lblLocationName.Location = New System.Drawing.Point(224, 80)
Me.lblLocationName.Name = "lblLocationName"
Me.lblLocationName.Size = New System.Drawing.Size(72, 32)
Me.lblLocationName.TabIndex = 62
Me.lblLocationName.Text = "Location Name"
Me.lblLocationName.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
'
'txtEventNum
'
Me.txtEventNum.BackColor =
System.Drawing.Color.FromArgb(CType(255, Byte), CType(255, Byte),
CType(128, Byte))
Me.txtEventNum.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInf01,
"Events.EventID"))
Me.txtEventNum.Location = New System.Drawing.Point(104, 80)
Me.txtEventNum.Name = "txtEventNum"
Me.txtEventNum.Size = New System.Drawing.Size(96, 20)
Me.txtEventNum.TabIndex = 61
Me.txtEventNum.Text = ""
'
'GroupBox2
'
Me.GroupBox2.Controls.Add(Me.txtCustNumber)
Me.GroupBox2.Controls.Add(Me.lblCustNumLabel)
Me.GroupBox2.Controls.Add(Me.txtTime)
Me.GroupBox2.Controls.Add(Me.txtDate)
Me.GroupBox2.Controls.Add(Me.lblHeadCountLabel)
Me.GroupBox2.Controls.Add(Me.lblTimeLabel)
Me.GroupBox2.Controls.Add(Me.lblDateLabel)
Me.GroupBox2.Controls.Add(Me.lblEventNumLabel)
Me.GroupBox2.Controls.Add(Me.lblNotes)
Me.GroupBox2.Controls.Add(Me.txtMenu)
Me.GroupBox2.Controls.Add(Me.lblMenu)
Me.GroupBox2.Controls.Add(Me.txtSpecialInstructions)
Me.GroupBox2.Controls.Add(Me.lblSpecialInstructions)
Me.GroupBox2.Controls.Add(Me.txtNotes)
Me.GroupBox2.Controls.Add(Me.txtHeadCount)
Me.GroupBox2.Location = New System.Drawing.Point(16, 64)
Me.GroupBox2.Name = "GroupBox2"
Me.GroupBox2.Size = New System.Drawing.Size(720, 304)
Me.GroupBox2.TabIndex = 77
Me.GroupBox2.TabStop = False
Me.GroupBox2.Text = "Event Information"
'
'txtCustNumber
'
Me.txtCustNumber.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInf01,
"Events.CustomerID"))
Me.txtCustNumber.Location = New System.Drawing.Point(96, 120)

```

```

Me.txtCustNumber.Name = "txtCustNumber"
Me.txtCustNumber.Size = New System.Drawing.Size(56, 20)
Me.txtCustNumber.TabIndex = 76
Me.txtCustNumber.Text = ""
'
'lblCustNumLabel
'
Me.lblCustNumLabel.Location = New System.Drawing.Point(16, 112)
Me.lblCustNumLabel.Name = "lblCustNumLabel"
Me.lblCustNumLabel.Size = New System.Drawing.Size(64, 32)
Me.lblCustNumLabel.TabIndex = 75
Me.lblCustNumLabel.Text = "Customer Number"
Me.lblCustNumLabel.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
'
'txtTime
'
Me.txtTime.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInfo1,
"Events.EventTime"))
Me.txtTime.Location = New System.Drawing.Point(88, 80)
Me.txtTime.Name = "txtTime"
Me.txtTime.Size = New System.Drawing.Size(96, 20)
Me.txtTime.TabIndex = 55
Me.txtTime.Text = ""
'
'txtDate
'
Me.txtDate.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInfo1,
"Events.EventDate"))
Me.txtDate.Location = New System.Drawing.Point(88, 48)
Me.txtDate.Name = "txtDate"
Me.txtDate.Size = New System.Drawing.Size(96, 20)
Me.txtDate.TabIndex = 54
Me.txtDate.Text = ""
'
'lblHeadCountLabel
'
Me.lblHeadCountLabel.Location = New System.Drawing.Point(200,
120)
Me.lblHeadCountLabel.Name = "lblHeadCountLabel"
Me.lblHeadCountLabel.Size = New System.Drawing.Size(64, 24)
Me.lblHeadCountLabel.TabIndex = 53
Me.lblHeadCountLabel.Text = "Head Count"
Me.lblHeadCountLabel.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
'
'lblTimeLabel
'
Me.lblTimeLabel.Location = New System.Drawing.Point(16, 80)
Me.lblTimeLabel.Name = "lblTimeLabel"
Me.lblTimeLabel.Size = New System.Drawing.Size(64, 24)
Me.lblTimeLabel.TabIndex = 52
Me.lblTimeLabel.Text = "Service Time"
Me.lblTimeLabel.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter

```

```

    ,
    'lblDateLabel
    ,
    Me.lblDateLabel.Location = New System.Drawing.Point(16, 48)
    Me.lblDateLabel.Name = "lblDateLabel"
    Me.lblDateLabel.Size = New System.Drawing.Size(64, 24)
    Me.lblDateLabel.TabIndex = 51
    Me.lblDateLabel.Text = "Date"
    Me.lblDateLabel.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
    ,
    'lblEventNumLabel
    ,
    Me.lblEventNumLabel.Location = New System.Drawing.Point(16, 16)
    Me.lblEventNumLabel.Name = "lblEventNumLabel"
    Me.lblEventNumLabel.Size = New System.Drawing.Size(64, 24)
    Me.lblEventNumLabel.TabIndex = 50
    Me.lblEventNumLabel.Text = "Event #"
    Me.lblEventNumLabel.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
    ,
    'lblNotes
    ,
    Me.lblNotes.Location = New System.Drawing.Point(16, 240)
    Me.lblNotes.Name = "lblNotes"
    Me.lblNotes.Size = New System.Drawing.Size(72, 24)
    Me.lblNotes.TabIndex = 46
    Me.lblNotes.Text = "Notes"
    Me.lblNotes.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
    ,
    'txtMenu
    ,
    Me.txtMenu.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInfo1, "Events.Menu"))
    Me.txtMenu.Location = New System.Drawing.Point(104, 160)
    Me.txtMenu.Name = "txtMenu"
    Me.txtMenu.ScrollBars =
System.Windows.Forms.ScrollBars.Vertical
    Me.txtMenu.Size = New System.Drawing.Size(600, 20)
    Me.txtMenu.TabIndex = 47
    Me.txtMenu.Text = ""
    ,
    'lblMenu
    ,
    Me.lblMenu.Location = New System.Drawing.Point(16, 152)
    Me.lblMenu.Name = "lblMenu"
    Me.lblMenu.Size = New System.Drawing.Size(72, 24)
    Me.lblMenu.TabIndex = 44
    Me.lblMenu.Text = "Menu"
    Me.lblMenu.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
    ,
    'txtSpecialInstructions
    ,

```

```

        Me.txtSpecialInstructions.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInfo1,
"Events.SpecialInst"))
        Me.txtSpecialInstructions.Location = New
System.Drawing.Point(104, 216)
        Me.txtSpecialInstructions.Name = "txtSpecialInstructions"
        Me.txtSpecialInstructions.ScrollBars =
System.Windows.Forms.ScrollBars.Vertical
        Me.txtSpecialInstructions.Size = New System.Drawing.Size(600,
20)
        Me.txtSpecialInstructions.TabIndex = 48
        Me.txtSpecialInstructions.Text = ""
        '
        'lblSpecialInstructions
        '
        Me.lblSpecialInstructions.Location = New
System.Drawing.Point(16, 208)
        Me.lblSpecialInstructions.Name = "lblSpecialInstructions"
        Me.lblSpecialInstructions.Size = New System.Drawing.Size(72,
24)
        Me.lblSpecialInstructions.TabIndex = 45
        Me.lblSpecialInstructions.Text = "Special Instructions"
        Me.lblSpecialInstructions.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
        '
        'txtNotes
        '
        Me.txtNotes.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInfo1, "Events.Notes"))
        Me.txtNotes.Location = New System.Drawing.Point(104, 248)
        Me.txtNotes.Name = "txtNotes"
        Me.txtNotes.ScrollBars =
System.Windows.Forms.ScrollBars.Vertical
        Me.txtNotes.Size = New System.Drawing.Size(600, 20)
        Me.txtNotes.TabIndex = 49
        Me.txtNotes.Text = ""
        '
        'Labell
        '
        Me.Labell.Font = New System.Drawing.Font("Microsoft Sans
Serif", 14.25!, System.Drawing.FontStyle.Regular,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
        Me.Labell.Location = New System.Drawing.Point(32, 16)
        Me.Labell.Name = "Labell"
        Me.Labell.Size = New System.Drawing.Size(392, 40)
        Me.Labell.TabIndex = 78
        Me.Labell.Text = "Catering Event Information"
        '
        'txtCheckNum
        '
        Me.txtCheckNum.Location = New System.Drawing.Point(648, 480)
        Me.txtCheckNum.Name = "txtCheckNum"
        Me.txtCheckNum.Size = New System.Drawing.Size(80, 20)
        Me.txtCheckNum.TabIndex = 94
        Me.txtCheckNum.Text = ""
        '
        'lblCheckNum

```

```

    Me.lblCheckNum.Location = New System.Drawing.Point(584, 480)
    Me.lblCheckNum.Name = "lblCheckNum"
    Me.lblCheckNum.Size = New System.Drawing.Size(56, 24)
    Me.lblCheckNum.TabIndex = 93
    Me.lblCheckNum.Text = "Check #"
    Me.lblCheckNum.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
    '
    'txtStoreChargeNum
    '
480) Me.txtStoreChargeNum.Location = New System.Drawing.Point(344,
    Me.txtStoreChargeNum.Name = "txtStoreChargeNum"
    Me.txtStoreChargeNum.Size = New System.Drawing.Size(232, 20)
    Me.txtStoreChargeNum.TabIndex = 92
    Me.txtStoreChargeNum.Text = ""
    '
    'lblStoreChargeNum
    '
480) Me.lblStoreChargeNum.Location = New System.Drawing.Point(216,
    Me.lblStoreChargeNum.Name = "lblStoreChargeNum"
    Me.lblStoreChargeNum.Size = New System.Drawing.Size(120, 24)
    Me.lblStoreChargeNum.TabIndex = 91
    Me.lblStoreChargeNum.Text = "Store Charge Number"
    '
    'txtCCDate
    '
    Me.txtCCDate.Font = New System.Drawing.Font("Microsoft Sans
Serif", 9.75!, System.Drawing.FontStyle.Bold,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
    Me.txtCCDate.Location = New System.Drawing.Point(648, 440)
    Me.txtCCDate.Name = "txtCCDate"
    Me.txtCCDate.Size = New System.Drawing.Size(80, 22)
    Me.txtCCDate.TabIndex = 90
    Me.txtCCDate.Text = "**/**"
    '
    'lblCCDate
    '
    Me.lblCCDate.Location = New System.Drawing.Point(584, 440)
    Me.lblCCDate.Name = "lblCCDate"
    Me.lblCCDate.Size = New System.Drawing.Size(56, 24)
    Me.lblCCDate.TabIndex = 89
    Me.lblCCDate.Text = "Exp. Date"
    '
    'txtCCNum
    '
    Me.txtCCNum.Font = New System.Drawing.Font("Microsoft Sans
Serif", 8.25!, System.Drawing.FontStyle.Bold,
System.Drawing.GraphicsUnit.Point, CType(0, Byte))
    Me.txtCCNum.Location = New System.Drawing.Point(344, 440)
    Me.txtCCNum.Name = "txtCCNum"
    Me.txtCCNum.Size = New System.Drawing.Size(232, 20)
    Me.txtCCNum.TabIndex = 88
    Me.txtCCNum.Text = "See Secure Card File"
    '

```

```

        'lblCCNum
        ,
        Me.lblCCNum.Location = New System.Drawing.Point(216, 440)
        Me.lblCCNum.Name = "lblCCNum"
        Me.lblCCNum.Size = New System.Drawing.Size(112, 24)
        Me.lblCCNum.TabIndex = 87
        Me.lblCCNum.Text = "CC Number"
        Me.lblCCNum.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
        ,
        'Label2
        ,
        Me.Label2.Location = New System.Drawing.Point(216, 400)
        Me.Label2.Name = "Label2"
        Me.Label2.Size = New System.Drawing.Size(112, 24)
        Me.Label2.TabIndex = 85
        Me.Label2.Text = "Payment Type"
        Me.Label2.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
        ,
        'txtTotalCost
        ,
        Me.txtTotalCost.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInf01,
"Events.EventCostTotal"))
        Me.txtTotalCost.Location = New System.Drawing.Point(128, 480)
        Me.txtTotalCost.Name = "txtTotalCost"
        Me.txtTotalCost.Size = New System.Drawing.Size(72, 20)
        Me.txtTotalCost.TabIndex = 84
        Me.txtTotalCost.Text = ""
        ,
        'lblTotalCost
        ,
        Me.lblTotalCost.Location = New System.Drawing.Point(32, 480)
        Me.lblTotalCost.Name = "lblTotalCost"
        Me.lblTotalCost.Size = New System.Drawing.Size(88, 32)
        Me.lblTotalCost.TabIndex = 83
        Me.lblTotalCost.Text = "Total Cost"
        Me.lblTotalCost.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
        ,
        'txtAddCosts
        ,
        Me.txtAddCosts.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInf01,
"Events.AdditonalCharges"))
        Me.txtAddCosts.Location = New System.Drawing.Point(128, 440)
        Me.txtAddCosts.Name = "txtAddCosts"
        Me.txtAddCosts.Size = New System.Drawing.Size(72, 20)
        Me.txtAddCosts.TabIndex = 82
        Me.txtAddCosts.Text = ""
        ,
        'lblAddCosts
        ,
        Me.lblAddCosts.Location = New System.Drawing.Point(32, 440)
        Me.lblAddCosts.Name = "lblAddCosts"
        Me.lblAddCosts.Size = New System.Drawing.Size(88, 32)

```



```

        Me.lblAddCosts.TabIndex = 81
        Me.lblAddCosts.Text = "Additional Charges"
        Me.lblAddCosts.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
    '
    'txtCostPerPerson
    '
        Me.txtCostPerPerson.DataBindings.Add(New
System.Windows.Forms.Binding("Text", Me.DsEventInfol,
"Events.CostPerPerson"))
        Me.txtCostPerPerson.Location = New System.Drawing.Point(128,
400)
        Me.txtCostPerPerson.Name = "txtCostPerPerson"
        Me.txtCostPerPerson.Size = New System.Drawing.Size(72, 20)
        Me.txtCostPerPerson.TabIndex = 80
        Me.txtCostPerPerson.Text = ""
    '
    'lblCostPerPerson
    '
        Me.lblCostPerPerson.Location = New System.Drawing.Point(32,
400)
        Me.lblCostPerPerson.Name = "lblCostPerPerson"
        Me.lblCostPerPerson.Size = New System.Drawing.Size(88, 24)
        Me.lblCostPerPerson.TabIndex = 79
        Me.lblCostPerPerson.Text = "Cost Per Person"
        Me.lblCostPerPerson.TextAlign =
System.Drawing.ContentAlignment.MiddleCenter
    '
    'GroupBox3
    '
        Me.GroupBox3.Controls.Add(Me.txtPaymentType)
        Me.GroupBox3.Location = New System.Drawing.Point(24, 384)
        Me.GroupBox3.Name = "GroupBox3"
        Me.GroupBox3.Size = New System.Drawing.Size(720, 144)
        Me.GroupBox3.TabIndex = 95
        Me.GroupBox3.TabStop = False
        Me.GroupBox3.Text = "Payment Information"
    '
    'txtPaymentType
    '
        Me.txtPaymentType.Location = New System.Drawing.Point(320, 16)
        Me.txtPaymentType.Name = "txtPaymentType"
        Me.txtPaymentType.Size = New System.Drawing.Size(232, 20)
        Me.txtPaymentType.TabIndex = 0
        Me.txtPaymentType.Text = ""
    '
    'cmdCancel
    '
        Me.cmdCancel.BackColor = System.Drawing.Color.Red
        Me.cmdCancel.Location = New System.Drawing.Point(688, 16)
        Me.cmdCancel.Name = "cmdCancel"
        Me.cmdCancel.Size = New System.Drawing.Size(64, 32)
        Me.cmdCancel.TabIndex = 97
        Me.cmdCancel.Text = "Close"
    '
    'OleDbDataAdapter1
    '

```

```

Me.OleDbDataAdapter1.DeleteCommand = Me.OleDbDeleteCommand1
Me.OleDbDataAdapter1.InsertCommand = Me.OleDbInsertCommand1
Me.OleDbDataAdapter1.SelectCommand = Me.OleDbSelectCommand1
Me.OleDbDataAdapter1.TableMappings.AddRange(New
System.Data.Common.DataTableMapping() {New
System.Data.Common.DataTableMapping("Table", "Events", New
System.Data.Common.DataColumnMapping() {New
System.Data.Common.DataColumnMapping("AdditonalCharges",
"AdditonalCharges"), New System.Data.Common.DataColumnMapping("City",
"City"), New System.Data.Common.DataColumnMapping("CostPerPerson",
"CostPerPerson"), New
System.Data.Common.DataColumnMapping("CustomerID", "CustomerID"), New
System.Data.Common.DataColumnMapping("EventCostTotal",
"EventCostTotal"), New
System.Data.Common.DataColumnMapping("EventDate", "EventDate"), New
System.Data.Common.DataColumnMapping("EventID", "EventID"), New
System.Data.Common.DataColumnMapping("EventTime", "EventTime"), New
System.Data.Common.DataColumnMapping("HeadCount", "HeadCount"), New
System.Data.Common.DataColumnMapping("Location", "Location"), New
System.Data.Common.DataColumnMapping("LocationAddress",
"LocationAddress"), New System.Data.Common.DataColumnMapping("Menu",
"Menu"), New System.Data.Common.DataColumnMapping("Notes", "Notes"),
New System.Data.Common.DataColumnMapping("PostalCode", "PostalCode"),
New System.Data.Common.DataColumnMapping("SpecialInst", "SpecialInst"),
New System.Data.Common.DataColumnMapping("StateOrProvince",
"StateOrProvince")}}})
Me.OleDbDataAdapter1.UpdateCommand = Me.OleDbUpdateCommand1
,
'OleDbDeleteCommand1
,
Me.OleDbDeleteCommand1.CommandText = "DELETE FROM Events WHERE
(EventID = ?) AND (AdditonalCharges = ? OR ? IS NULL AND" & _
" AdditonalCharges IS NULL) AND (City = ? OR ? IS NULL AND City
IS NULL) AND (Cos" & _
"tPerPerson = ? OR ? IS NULL AND CostPerPerson IS NULL) AND
(CustomerID = ? OR ? " & _
"IS NULL AND CustomerID IS NULL) AND (EventCostTotal = ? OR ?
IS NULL AND EventCo" & _
"stTotal IS NULL) AND (EventDate = ? OR ? IS NULL AND EventDate
IS NULL) AND (Eve" & _
"ntTime = ? OR ? IS NULL AND EventTime IS NULL) AND (HeadCount
= ?) AND (Location" & _
" = ? OR ? IS NULL AND Location IS NULL) AND (LocationAddress =
? OR ? IS NULL AN" & _
"D LocationAddress IS NULL) AND (PostalCode = ? OR ? IS NULL
AND PostalCode IS NU" & _
"LL) AND (StateOrProvince = ? OR ? IS NULL AND StateOrProvince
IS NULL)"
Me.OleDbDeleteCommand1.Connection = Me.OleDbConnection1
Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventID", System.Data.DataRowVersion.Original, Nothing))
Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_AdditonalCharges",
System.Data.OleDb.OleDbType.Currency, 0,

```

```

System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "AdditonalCharges", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_AdditonalCharges1",
System.Data.OleDb.OleDbType.Currency, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "AdditonalCharges", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_City",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "City", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_City1",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "City", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CostPerPerson",
System.Data.OleDb.OleDbType.Currency, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CostPerPerson", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CostPerPerson1",
System.Data.OleDb.OleDbType.Currency, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CostPerPerson", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID1",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventCostTotal",
System.Data.OleDb.OleDbType.Currency, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventCostTotal", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventCostTotal1",
System.Data.OleDb.OleDbType.Currency, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventCostTotal", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventDate",
System.Data.OleDb.OleDbType.DBDate, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventDate", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventDate1",
System.Data.OleDb.OleDbType.DBDate, 0,

```

```

System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventDate", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventTime",
System.Data.OleDb.OleDbType.DBDate, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventTime", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventTime1",
System.Data.OleDb.OleDbType.DBDate, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventTime", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_HeadCount",
System.Data.OleDb.OleDbType.Boolean, 2,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "HeadCount", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location1",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_LocationAddress",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "LocationAddress", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_LocationAddress1",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "LocationAddress", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_PostalCode",
System.Data.OleDb.OleDbType.VarWChar, 20,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "PostalCode", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_PostalCode1",
System.Data.OleDb.OleDbType.VarWChar, 20,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "PostalCode", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_StateOrProvince",
System.Data.OleDb.OleDbType.VarWChar, 20,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "StateOrProvince", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_StateOrProvince1",

```

```

System.Data.OleDb.OleDbType.VarWChar, 20,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "StateOrProvince", System.Data.DataRowVersion.Original,
Nothing))
,
'OleDbConnection1
,
Me.OleDbConnection1.ConnectionString = "Jet OLEDB:Global
Partial Bulk Ops=2;Jet OLEDB:Registry Path=;Jet OLEDB:Database L" & _
"ocking Mode=1;Jet OLEDB:Database Password=;Data
Source=""C:\catering\Events.mdb"";" & _
"Password=;Jet OLEDB:Engine Type=5;Jet OLEDB:Global Bulk
Transactions=1;Provider=" & _
""Microsoft.Jet.OLEDB.4.0"";Jet OLEDB:System database=;Jet
OLEDB:SFP=False;Extende" & _
"d Properties=;Mode=Share Deny None;Jet OLEDB:New Database
Password=;Jet OLEDB:Cr" & _
"eate System Database=False;Jet OLEDB:Don't Copy Locale on
Compact=False;Jet OLEDB" & _
"B:Compact Without Replica Repair=False;User ID=Admin;Jet
OLEDB:Encrypt Database=" & _
"False"
,
'OleDbInsertCommand1
,
Me.OleDbInsertCommand1.CommandText = "INSERT INTO
Events(AdditonalCharges, City, CostPerPerson, CustomerID, EventCostTo"
& _
"tal, EventDate, EventTime, HeadCount, Location,
LocationAddress, Menu, Notes, Po" & _
"stalCode, SpecialInst, StateOrProvince) VALUES (?, ?, ?, ?, ?,
?, ?, ?, ?, ?, ?, ?, ?)" & _
" ?, ?, ?, ?)"
Me.OleDbInsertCommand1.Connection = Me.OleDbConnection1
Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("AdditonalCharges",
System.Data.OleDb.OleDbType.Currency, 0, "AdditonalCharges"))
Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("City",
System.Data.OleDb.OleDbType.VarWChar, 50, "City"))
Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("CostPerPerson",
System.Data.OleDb.OleDbType.Currency, 0, "CostPerPerson"))
Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("CustomerID",
System.Data.OleDb.OleDbType.Integer, 0, "CustomerID"))
Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("EventCostTotal",
System.Data.OleDb.OleDbType.Currency, 0, "EventCostTotal"))
Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("EventDate",
System.Data.OleDb.OleDbType.DBDate, 0, "EventDate"))
Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("EventTime",
System.Data.OleDb.OleDbType.DBDate, 0, "EventTime"))

```

```

        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("HeadCount",
System.Data.OleDb.OleDbType.Boolean, 2, "HeadCount"))
        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Location",
System.Data.OleDb.OleDbType.VarWChar, 255, "Location"))
        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("LocationAddress",
System.Data.OleDb.OleDbType.VarWChar, 255, "LocationAddress"))
        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Menu",
System.Data.OleDb.OleDbType.VarWChar, 0, "Menu"))
        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Notes",
System.Data.OleDb.OleDbType.VarWChar, 0, "Notes"))
        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("PostalCode",
System.Data.OleDb.OleDbType.VarWChar, 20, "PostalCode"))
        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("SpecialInst",
System.Data.OleDb.OleDbType.VarWChar, 0, "SpecialInst"))
        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("StateOrProvince",
System.Data.OleDb.OleDbType.VarWChar, 20, "StateOrProvince"))
        ,
        'OleDbSelectCommand1
        ,
        Me.OleDbSelectCommand1.CommandText = "SELECT AdditonalCharges,
City, CostPerPerson, CustomerID, EventCostTotal, EventDa" & _
        "te, EventID, EventTime, HeadCount, Location, LocationAddress,
Menu, Notes, Posta" & _
        "lCode, SpecialInst, StateOrProvince FROM Events"
        Me.OleDbSelectCommand1.Connection = Me.OleDbConnection1
        ,
        'OleDbUpdateCommand1
        ,
        Me.OleDbUpdateCommand1.CommandText = "UPDATE Events SET
AdditonalCharges = ?, City = ?, CostPerPerson = ?, CustomerID =" & _
        " ?, EventCostTotal = ?, EventDate = ?, EventTime = ?,
HeadCount = ?, Location = " & _
        "?, LocationAddress = ?, Menu = ?, Notes = ?, PostalCode = ?,
SpecialInst = ?, St" & _
        "ateOrProvince = ? WHERE (EventID = ?) AND (AdditonalCharges =
? OR ? IS NULL AND" & _
        " AdditonalCharges IS NULL) AND (City = ? OR ? IS NULL AND City
IS NULL) AND (Cos" & _
        "tPerPerson = ? OR ? IS NULL AND CostPerPerson IS NULL) AND
(CustomerID = ? OR ? " & _
        "IS NULL AND CustomerID IS NULL) AND (EventCostTotal = ? OR ?
IS NULL AND EventCo" & _
        "stTotal IS NULL) AND (EventDate = ? OR ? IS NULL AND EventDate
IS NULL) AND (Eve" & _
        "ntTime = ? OR ? IS NULL AND EventTime IS NULL) AND (HeadCount
= ?) AND (Location" & _
        " = ? OR ? IS NULL AND Location IS NULL) AND (LocationAddress =
? OR ? IS NULL AN" & _

```



```

        "D LocationAddress IS NULL) AND (PostalCode = ? OR ? IS NULL
AND PostalCode IS NU" & _
        "LL) AND (StateOrProvince = ? OR ? IS NULL AND StateOrProvince
IS NULL)"
        Me.OleDbUpdateCommand1.Connection = Me.OleDbConnection1
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("AdditonalCharges",
System.Data.OleDb.OleDbType.Currency, 0, "AdditonalCharges"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("City",
System.Data.OleDb.OleDbType.VarWChar, 50, "City"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("CostPerPerson",
System.Data.OleDb.OleDbType.Currency, 0, "CostPerPerson"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("CustomerID",
System.Data.OleDb.OleDbType.Integer, 0, "CustomerID"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("EventCostTotal",
System.Data.OleDb.OleDbType.Currency, 0, "EventCostTotal"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("EventDate",
System.Data.OleDb.OleDbType.DBDate, 0, "EventDate"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("EventTime",
System.Data.OleDb.OleDbType.DBDate, 0, "EventTime"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("HeadCount",
System.Data.OleDb.OleDbType.Boolean, 2, "HeadCount"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Location",
System.Data.OleDb.OleDbType.VarWChar, 255, "Location"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("LocationAddress",
System.Data.OleDb.OleDbType.VarWChar, 255, "LocationAddress"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Menu",
System.Data.OleDb.OleDbType.VarWChar, 0, "Menu"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Notes",
System.Data.OleDb.OleDbType.VarWChar, 0, "Notes"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("PostalCode",
System.Data.OleDb.OleDbType.VarWChar, 20, "PostalCode"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("SpecialInst",
System.Data.OleDb.OleDbType.VarWChar, 0, "SpecialInst"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("StateOrProvince",
System.Data.OleDb.OleDbType.VarWChar, 20, "StateOrProvince"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventID", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_AdditonalCharges",

```

```

System.Data.OleDb.OleDbType.Currency, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "AdditionalCharges", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_AdditonalCharges1",
System.Data.OleDb.OleDbType.Currency, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "AdditionalCharges", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_City",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "City", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_City1",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "City", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CostPerPerson",
System.Data.OleDb.OleDbType.Currency, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CostPerPerson", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CostPerPerson1",
System.Data.OleDb.OleDbType.Currency, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CostPerPerson", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID1",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventCostTotal",
System.Data.OleDb.OleDbType.Currency, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventCostTotal", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventCostTotal1",
System.Data.OleDb.OleDbType.Currency, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventCostTotal", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventDate",
System.Data.OleDb.OleDbType.DBDate, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventDate", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventDate1",

```



```

System.Data.OleDb.OleDbType.DBDate, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventDate", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventTime",
System.Data.OleDb.OleDbType.DBDate, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventTime", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventTime1",
System.Data.OleDb.OleDbType.DBDate, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventTime", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_HeadCount",
System.Data.OleDb.OleDbType.Boolean, 2,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "HeadCount", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location1",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_LocationAddress",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "LocationAddress", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_LocationAddress1",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "LocationAddress", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_PostalCode",
System.Data.OleDb.OleDbType.VarWChar, 20,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "PostalCode", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_PostalCode1",
System.Data.OleDb.OleDbType.VarWChar, 20,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "PostalCode", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_StateOrProvince",
System.Data.OleDb.OleDbType.VarWChar, 20,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "StateOrProvince", System.Data.DataRowVersion.Original,
Nothing))

```

```

        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_StateOrProvince",
System.Data.OleDb.OleDbType.VarWChar, 20,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "StateOrProvince", System.Data.DataRowVersion.Original,
Nothing))
    ,
    'btnNext
    ,
    Me.btnNext.BackColor = System.Drawing.Color.FromArgb(CType(128,
Byte), CType(255, Byte), CType(255, Byte))
    Me.btnNext.Location = New System.Drawing.Point(424, 16)
    Me.btnNext.Name = "btnNext"
    Me.btnNext.Size = New System.Drawing.Size(64, 32)
    Me.btnNext.TabIndex = 99
    Me.btnNext.Text = "Next"
    ,
    'btnPrevious
    ,
    Me.btnPrevious.BackColor =
System.Drawing.Color.FromArgb(CType(128, Byte), CType(255, Byte),
CType(255, Byte))
    Me.btnPrevious.Location = New System.Drawing.Point(352, 16)
    Me.btnPrevious.Name = "btnPrevious"
    Me.btnPrevious.Size = New System.Drawing.Size(64, 32)
    Me.btnPrevious.TabIndex = 100
    Me.btnPrevious.Text = "Previous"
    ,
    'btnFirst
    ,
    Me.btnFirst.BackColor =
System.Drawing.Color.FromArgb(CType(128, Byte), CType(255, Byte),
CType(128, Byte))
    Me.btnFirst.Location = New System.Drawing.Point(280, 16)
    Me.btnFirst.Name = "btnFirst"
    Me.btnFirst.Size = New System.Drawing.Size(64, 32)
    Me.btnFirst.TabIndex = 101
    Me.btnFirst.Text = "First"
    ,
    'btnLast
    ,
    Me.btnLast.BackColor = System.Drawing.Color.FromArgb(CType(128,
Byte), CType(255, Byte), CType(128, Byte))
    Me.btnLast.Location = New System.Drawing.Point(496, 16)
    Me.btnLast.Name = "btnLast"
    Me.btnLast.Size = New System.Drawing.Size(56, 32)
    Me.btnLast.TabIndex = 102
    Me.btnLast.Text = "Last"
    ,
    'btnNew
    ,
    Me.btnNew.BackColor = System.Drawing.Color.Yellow
    Me.btnNew.Location = New System.Drawing.Point(624, 16)
    Me.btnNew.Name = "btnNew"
    Me.btnNew.Size = New System.Drawing.Size(56, 32)
    Me.btnNew.TabIndex = 103
    Me.btnNew.Text = "New"

```

```

Me.btnNew.Visible = False
'
'btnUpdate
'
Me.btnUpdate.BackColor = System.Drawing.Color.Yellow
Me.btnUpdate.Location = New System.Drawing.Point(560, 16)
Me.btnUpdate.Name = "btnUpdate"
Me.btnUpdate.Size = New System.Drawing.Size(56, 32)
Me.btnUpdate.TabIndex = 104
Me.btnUpdate.Text = "Update Records"
'
' frmEventInfo
'
Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
Me.ClientSize = New System.Drawing.Size(768, 746)
Me.Controls.Add(Me.btnUpdate)
Me.Controls.Add(Me.btnNew)
Me.Controls.Add(Me.btnLast)
Me.Controls.Add(Me.btnFirst)
Me.Controls.Add(Me.btnPrevious)
Me.Controls.Add(Me.btnNext)
Me.Controls.Add(Me.cmdCancel)
Me.Controls.Add(Me.txtCheckNum)
Me.Controls.Add(Me.lblCheckNum)
Me.Controls.Add(Me.txtStoreChargeNum)
Me.Controls.Add(Me.lblStoreChargeNum)
Me.Controls.Add(Me.txtCCDate)
Me.Controls.Add(Me.lblCCDate)
Me.Controls.Add(Me.txtCCNum)
Me.Controls.Add(Me.lblCCNum)
Me.Controls.Add(Me.Label2)
Me.Controls.Add(Me.txtTotalCost)
Me.Controls.Add(Me.lblTotalCost)
Me.Controls.Add(Me.txtAddCosts)
Me.Controls.Add(Me.lblAddCosts)
Me.Controls.Add(Me.txtCostPerPerson)
Me.Controls.Add(Me.lblCostPerPerson)
Me.Controls.Add(Me.GroupBox3)
Me.Controls.Add(Me.Label1)
Me.Controls.Add(Me.txtLocationZip)
Me.Controls.Add(Me.lblLocationZip)
Me.Controls.Add(Me.txtLocationState)
Me.Controls.Add(Me.lblLocationState)
Me.Controls.Add(Me.txtLocationCity)
Me.Controls.Add(Me.lblLocationCity)
Me.Controls.Add(Me.txtLocationAddress)
Me.Controls.Add(Me.lblLocationAddress)
Me.Controls.Add(Me.txtLocatoinName)
Me.Controls.Add(Me.lblLocationName)
Me.Controls.Add(Me.txtEventNum)
Me.Controls.Add(Me.GroupBox2)
Me.Name = "frmEventInfo"
Me.Text = "Event Information"
CType(Me.DsEventInfo1,
System.ComponentModel.ISupportInitialize).EndInit()
Me.GroupBox2.ResumeLayout(False)
Me.GroupBox3.ResumeLayout(False)

```

```
Me.ResumeLayout(False)
```

```
End Sub
```

```
#End Region
```

```
*****  
,  
,  
, Form Name: frmEventInfo  
,  
, Purpose: This form is used to review and update the information  
, about an event. It is made up of several different  
, text boxes and labels. It also contains controls  
, for navigating through the records.  
,  
, Updated: November 10, 2004  
, Updated By: Matthew D. Johnson  
,  
,  
*****
```

```
Private Sub cmdCancel_Click(ByVal sender As System.Object, ByVal e  
As System.EventArgs) Handles cmdCancel.Click  
    'This command is used to close the window when called by the  
user.  
    Close()  
End Sub
```

```
*****  
*****  
,  
,  
, *****NOT USED PROCEDURE MATERIAL***SLOTTED FOR FUTURE  
UPDATES*****  
,  
,  
, Private Sub btnLoad_Click(ByVal sender As System.Object, ByVal e  
As System.EventArgs)  
,  
,     DsEventInfo1.Clear()  
,     OleDbDataAdapter1.Fill(DsEventInfo1)  
,  
, End Sub  
,  
,  
,  
*****
```

```
*****  
*****  
Private Sub btnNext_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles btnNext.Click  
    'Button used in navigating through the data  
    Me.BindingContext(DsEventInfo1, "events").Position += 1  
End Sub
```

```
Private Sub frmEventInfo_Load(ByVal sender As System.Object, ByVal  
e As System.EventArgs) Handles MyBase.Load
```

```

        'Command that is called when the form is loaded. It loads the
data adapter and fills it.
        DsEventInfo1.Clear()
        OleDbDataAdapter1.Fill(DsEventInfo1)
    End Sub

    Private Sub btnPrevious_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles btnPrevious.Click
        'Button used in navigating through the data
        Me.BindingContext(DsEventInfo1, "events").Position -= 1
    End Sub

    Private Sub btnFirst_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnFirst.Click
        'Button used in navigating through the data
        Me.BindingContext(DsEventInfo1, "events").Position = 0
    End Sub

    Private Sub btnLast_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLast.Click
        'Button used in navigating through the data
        Me.BindingContext(DsEventInfo1, "events").Position =
Me.BindingContext(DsEventInfo1, "events").Count - 1
    End Sub

    Private Sub btnUpdate_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnUpdate.Click
        'Command that is used to update the database when the button is
clicked.

        Dim Current As Short
        Current = Me.BindingContext(DsEventInfo1, "Events").Position
        Me.BindingContext(DsEventInfo1, "Events").Position = 0
        Me.BindingContext(DsEventInfo1, "Events").Position = Current
        OleDbDataAdapter1.Update(DsEventInfo1)
        MsgBox("Update Complete", MsgBoxStyle.Information, "Update
Confirmation")
    End Sub

End Class

```

Appendix I - Search Customer Form Code

```
Public Class frmSearchCust
    Inherits System.Windows.Forms.Form

#Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overrides Sub Dispose(ByVal disposing As
Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form
Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents OleDbDataAdapter1 As
System.Data.OleDb.OleDbDataAdapter
    Friend WithEvents OleDbSelectCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbInsertCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbUpdateCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbDeleteCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbConnection1 As
System.Data.OleDb.OleDbConnection
    Friend WithEvents DsCust1 As WindowsApplication1.DsCust
    Friend WithEvents DataGrid1 As System.Windows.Forms.DataGrid
    Friend WithEvents btnClose As System.Windows.Forms.Button
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()
        Me.OleDbDataAdapter1 = New System.Data.OleDb.OleDbDataAdapter
        Me.OleDbDeleteCommand1 = New System.Data.OleDb.OleDbCommand
        Me.OleDbConnection1 = New System.Data.OleDb.OleDbConnection
        Me.OleDbInsertCommand1 = New System.Data.OleDb.OleDbCommand
```

```

Me.OleDbSelectCommand1 = New System.Data.OleDb.OleDbCommand
Me.OleDbUpdateCommand1 = New System.Data.OleDb.OleDbCommand
Me.DsCust1 = New WindowsApplication1.DsCust
Me.DataGrid1 = New System.Windows.Forms.DataGrid
Me.btnClose = New System.Windows.Forms.Button
CType(Me.DsCust1,
System.ComponentModel.ISupportInitialize).BeginInit()
CType(Me.DataGrid1,
System.ComponentModel.ISupportInitialize).BeginInit()
Me.SuspendLayout()
'
'OleDbDataAdapter1
'
Me.OleDbDataAdapter1.DeleteCommand = Me.OleDbDeleteCommand1
Me.OleDbDataAdapter1.InsertCommand = Me.OleDbInsertCommand1
Me.OleDbDataAdapter1.SelectCommand = Me.OleDbSelectCommand1
Me.OleDbDataAdapter1.TableMappings.AddRange(New
System.Data.Common.DataTableMapping() {New
System.Data.Common.DataTableMapping("Table", "Customers", New
System.Data.Common.DataColumnMapping() {New
System.Data.Common.DataColumnMapping("CustomerID", "CustomerID"), New
System.Data.Common.DataColumnMapping("ContactFirstName",
"ContactFirstName"), New
System.Data.Common.DataColumnMapping("ContactLastName",
"ContactLastName"), New
System.Data.Common.DataColumnMapping("CompanyOrDepartment",
"CompanyOrDepartment")}}})
Me.OleDbDataAdapter1.UpdateCommand = Me.OleDbUpdateCommand1
'OleDbDeleteCommand1
'
Me.OleDbDeleteCommand1.CommandText = "DELETE FROM Customers
WHERE (CustomerID = ?) AND (CompanyOrDepartment = ? OR ? IS" & _
" NULL AND CompanyOrDepartment IS NULL) AND (ContactFirstName =
? OR ? IS NULL AN" & _
"D ContactFirstName IS NULL) AND (ContactLastName = ? OR ? IS
NULL AND ContactLas" & _
"tName IS NULL)"
Me.OleDbDeleteCommand1.Connection = Me.OleDbConnection1
Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CompanyOrDepartment",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CompanyOrDepartment", System.Data.DataRowVersion.Original,
Nothing))
Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CompanyOrDepartment1",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CompanyOrDepartment", System.Data.DataRowVersion.Original,
Nothing))

```

```

        Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactFirstName",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactFirstName", System.Data.DataRowVersion.Original,
Nothing))
        Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactFirstName1",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactFirstName", System.Data.DataRowVersion.Original,
Nothing))
        Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactLastName",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactLastName", System.Data.DataRowVersion.Original,
Nothing))
        Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactLastName1",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactLastName", System.Data.DataRowVersion.Original,
Nothing))
        ,
        'OleDbConnection1
        ,
        Me.OleDbConnection1.ConnectionString = "Jet OLEDB:Global
Partial Bulk Ops=2;Jet OLEDB:Registry Path=;Jet OLEDB:Database L" & _
"ocking Mode=1;Jet OLEDB:Database Password=;Data
Source="C:\catering\Client.mdb""; " & _
"Password=;Jet OLEDB:Engine Type=5;Jet OLEDB:Global Bulk
Transactions=1;Provider=" & _
""Microsoft.Jet.OLEDB.4.0"";Jet OLEDB:System database=;Jet
OLEDB:SFP=False;Extende" & _
"d Properties=;Mode=Share Deny None;Jet OLEDB:New Database
Password=;Jet OLEDB:Cr" & _
"eate System Database=False;Jet OLEDB:Don't Copy Locale on
Compact=False;Jet OLEDB" & _
"B:Compact Without Replica Repair=False;User ID=Admin;Jet
OLEDB:Encrypt Database=" & _
"False"
        ,
        'OleDbInsertCommand1
        ,
        Me.OleDbInsertCommand1.CommandText = "INSERT INTO
Customers(ContactFirstName, ContactLastName, CompanyOrDepartment) VAL"
& _
"UES (?, ?, ?)"
        Me.OleDbInsertCommand1.Connection = Me.OleDbConnection1
        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("ContactFirstName",
System.Data.OleDb.OleDbType.VarWChar, 30, "ContactFirstName"))
        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("ContactLastName",
System.Data.OleDb.OleDbType.VarWChar, 50, "ContactLastName"))

```



```

        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("CompanyOrDepartment",
System.Data.OleDb.OleDbType.VarWChar, 50, "CompanyOrDepartment"))
    ,
    'OleDbSelectCommand1
    ,
    Me.OleDbSelectCommand1.CommandText = "SELECT CustomerID,
ContactFirstName, ContactLastName, CompanyOrDepartment FROM Cu" & _
    "stomers"
    Me.OleDbSelectCommand1.Connection = Me.OleDbConnection1
    ,
    'OleDbUpdateCommand1
    ,
    Me.OleDbUpdateCommand1.CommandText = "UPDATE Customers SET
ContactFirstName = ?, ContactLastName = ?, CompanyOrDepartme" & _
    "nt = ? WHERE (CustomerID = ?) AND (CompanyOrDepartment = ? OR
? IS NULL AND Comp" & _
    "anyOrDepartment IS NULL) AND (ContactFirstName = ? OR ? IS
NULL AND ContactFirst" & _
    "Name IS NULL) AND (ContactLastName = ? OR ? IS NULL AND
ContactLastName IS NULL)" & _
    ""
    Me.OleDbUpdateCommand1.Connection = Me.OleDbConnection1
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("ContactFirstName",
System.Data.OleDb.OleDbType.VarWChar, 30, "ContactFirstName"))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("ContactLastName",
System.Data.OleDb.OleDbType.VarWChar, 50, "ContactLastName"))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("CompanyOrDepartment",
System.Data.OleDb.OleDbType.VarWChar, 50, "CompanyOrDepartment"))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CompanyOrDepartment",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CompanyOrDepartment", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CompanyOrDepartment1",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CompanyOrDepartment", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactFirstName",
System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactFirstName", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactFirstName1",

```

```

System.Data.OleDb.OleDbType.VarWChar, 30,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactFirstName", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactLastName",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactLastName", System.Data.DataRowVersion.Original,
Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_ContactLastName1",
System.Data.OleDb.OleDbType.VarWChar, 50,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "ContactLastName", System.Data.DataRowVersion.Original,
Nothing))
    ,
    'DsCust1
    ,
    Me.DsCust1.DataSetName = "DsCust"
    Me.DsCust1.Locale = New System.Globalization.CultureInfo("en-
US")
    ,
    'DataGrid1
    ,
    Me.DataGrid1.DataMember = "Customers"
    Me.DataGrid1.DataSource = Me.DsCust1
    Me.DataGrid1.HeaderForeColor =
System.Drawing.SystemColors.ControlText
    Me.DataGrid1.Location = New System.Drawing.Point(32, 32)
    Me.DataGrid1.Name = "DataGrid1"
    Me.DataGrid1.PreferredColumnWidth = 125
    Me.DataGrid1.ReadOnly = True
    Me.DataGrid1.Size = New System.Drawing.Size(536, 216)
    Me.DataGrid1.TabIndex = 0
    ,
    'btnClose
    ,
    Me.btnClose.Location = New System.Drawing.Point(192, 264)
    Me.btnClose.Name = "btnClose"
    Me.btnClose.Size = New System.Drawing.Size(120, 32)
    Me.btnClose.TabIndex = 1
    Me.btnClose.Text = "Close"
    ,
    'frmSearchCust
    ,
    Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
    Me.ClientSize = New System.Drawing.Size(624, 334)
    Me.Controls.Add(Me.btnClose)
    Me.Controls.Add(Me.DataGrid1)
    Me.Name = "frmSearchCust"
    Me.Text = "Search Customers"
    CType(Me.DsCust1,
System.ComponentModel.ISupportInitialize).EndInit()
    CType(Me.DataGrid1,
System.ComponentModel.ISupportInitialize).EndInit()
    Me.ResumeLayout(False)

```

```

End Sub

#End Region

'*****
'
'   Form Name:   frmSearchCust
'
'   Purpose:     This form is used to search for customers by using
'                a DataGrid of information.  The user can search
'                based on name, ID number, or company.
'
'   Updated:     November 10, 2004
'   Updated By:  Matthew D. Johnson
'
'*****

Private Sub btnClose_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnClose.Click
    'Procedure that closes the window when the user clicks on a
button
    Close()
End Sub

Private Sub frmSearchCust_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load
    'Procedure that is called when the form is loaded.  It fills
the data grid.
    DsCust1.Clear()
    OleDbDataAdapter1.Fill(DsCust1)
End Sub

End Class

```

Appendix J - Search Events Form Code

```
Public Class frmSearchEvent
    Inherits System.Windows.Forms.Form

#Region " Windows Form Designer generated code "

    Public Sub New()
        MyBase.New()

        'This call is required by the Windows Form Designer.
        InitializeComponent()

        'Add any initialization after the InitializeComponent() call

    End Sub

    'Form overrides dispose to clean up the component list.
    Protected Overrides Sub Dispose(ByVal disposing As
Boolean)
        If disposing Then
            If Not (components Is Nothing) Then
                components.Dispose()
            End If
        End If
        MyBase.Dispose(disposing)
    End Sub

    'Required by the Windows Form Designer
    Private components As System.ComponentModel.IContainer

    'NOTE: The following procedure is required by the Windows Form
Designer
    'It can be modified using the Windows Form Designer.
    'Do not modify it using the code editor.
    Friend WithEvents OleDbDataAdapter1 As
System.Data.OleDb.OleDbDataAdapter
    Friend WithEvents OleDbSelectCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbInsertCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbUpdateCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbDeleteCommand1 As
System.Data.OleDb.OleDbCommand
    Friend WithEvents OleDbConnection1 As
System.Data.OleDb.OleDbConnection
    Friend WithEvents DataGridView1 As System.Windows.Forms.DataGridView
    Friend WithEvents DsEventSearch1 As
WindowsApplication1.DsEventSearch
    Friend WithEvents btnClose As System.Windows.Forms.Button
    <System.Diagnostics.DebuggerStepThrough()> Private Sub
InitializeComponent()
        Me.OleDbDataAdapter1 = New System.Data.OleDb.OleDbDataAdapter
        Me.OleDbDeleteCommand1 = New System.Data.OleDb.OleDbCommand
        Me.OleDbConnection1 = New System.Data.OleDb.OleDbConnection
```

```

Me.OleDbInsertCommand1 = New System.Data.OleDb.OleDbCommand
Me.OleDbSelectCommand1 = New System.Data.OleDb.OleDbCommand
Me.OleDbUpdateCommand1 = New System.Data.OleDb.OleDbCommand
Me.DataGrid1 = New System.Windows.Forms.DataGrid
Me.DsEventSearch1 = New WindowsApplication1.DsEventSearch
Me.btnClose = New System.Windows.Forms.Button
CType(Me.DataGrid1,
System.ComponentModel.ISupportInitialize).BeginInit()
CType(Me.DsEventSearch1,
System.ComponentModel.ISupportInitialize).BeginInit()
Me.SuspendLayout()
,
'OleDbDataAdapter1
,
Me.OleDbDataAdapter1.DeleteCommand = Me.OleDbDeleteCommand1
Me.OleDbDataAdapter1.InsertCommand = Me.OleDbInsertCommand1
Me.OleDbDataAdapter1.SelectCommand = Me.OleDbSelectCommand1
Me.OleDbDataAdapter1.TableMappings.AddRange(New
System.Data.Common.DataTableMapping() {New
System.Data.Common.DataTableMapping("Table", "Events", New
System.Data.Common.DataColumnMapping() {New
System.Data.Common.DataColumnMapping("EventID", "EventID"), New
System.Data.Common.DataColumnMapping("CustomerID", "CustomerID"), New
System.Data.Common.DataColumnMapping("Location", "Location"), New
System.Data.Common.DataColumnMapping("EventDate", "EventDate"), New
System.Data.Common.DataColumnMapping("EventCostTotal",
"EventCostTotal")}}})
Me.OleDbDataAdapter1.UpdateCommand = Me.OleDbUpdateCommand1
,
'OleDbDeleteCommand1
,
Me.OleDbDeleteCommand1.CommandText = "DELETE FROM Events WHERE
(EventID = ?) AND (CustomerID = ? OR ? IS NULL AND Custo" & _
"merID IS NULL) AND (EventCostTotal = ? OR ? IS NULL AND
EventCostTotal IS NULL) " & _
"AND (EventDate = ? OR ? IS NULL AND EventDate IS NULL) AND
(Location = ? OR ? IS" & _
" NULL AND Location IS NULL)"
Me.OleDbDeleteCommand1.Connection = Me.OleDbConnection1
Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventID", System.Data.DataRowVersion.Original, Nothing))
Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID1",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventCostTotal",
System.Data.OleDb.OleDbType.Currency, 0,

```

```

System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventCostTotal", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventCostTotal1",
System.Data.OleDb.OleDbType.Currency, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventCostTotal", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventDate",
System.Data.OleDb.OleDbType.DBDate, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventDate", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventDate1",
System.Data.OleDb.OleDbType.DBDate, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventDate", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbDeleteCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location1",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
    ,
    'OleDbConnection1
    ,
    Me.OleDbConnection1.ConnectionString = "Jet OLEDB:Global
Partial Bulk Ops=2;Jet OLEDB:Registry Path=;Jet OLEDB:Database L" & _
"ocking Mode=0;Jet OLEDB:Database Password=;Data
Source=""C:\catering\Events.mdb""; " & _
"Password=;Jet OLEDB:Engine Type=5;Jet OLEDB:Global Bulk
Transactions=1;Provider=" & _
""Microsoft.Jet.OLEDB.4.0"";Jet OLEDB:System database=;Jet
OLEDB:SFP=False;Extend" & _
"d Properties=;Mode=Share Deny None;Jet OLEDB:New Database
Password=;Jet OLEDB:Cr" & _
"eate System Database=False;Jet OLEDB:Don't Copy Locale on
Compact=False;Jet OLEDB" & _
"B:Compact Without Replica Repair=False;User ID=Admin;Jet
OLEDB:Encrypt Database=" & _
"False"
    ,
    'OleDbInsertCommand1
    ,
    Me.OleDbInsertCommand1.CommandText = "INSERT INTO
Events(CustomerID, Location, EventDate, EventCostTotal) VALUES (?, ?, "
& _
" ?, ?)"
    Me.OleDbInsertCommand1.Connection = Me.OleDbConnection1
    Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("CustomerID",
System.Data.OleDb.OleDbType.Integer, 0, "CustomerID"))

```

```

        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Location",
System.Data.OleDb.OleDbType.VarWChar, 255, "Location"))
        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("EventDate",
System.Data.OleDb.OleDbType.DBDate, 0, "EventDate"))
        Me.OleDbInsertCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("EventCostTotal",
System.Data.OleDb.OleDbType.Currency, 0, "EventCostTotal"))
    ,
        'OleDbSelectCommand1
    ,
        Me.OleDbSelectCommand1.CommandText = "SELECT EventID,
CustomerID, Location, EventDate, EventCostTotal FROM Events"
        Me.OleDbSelectCommand1.Connection = Me.OleDbConnection1
    ,
        'OleDbUpdateCommand1
    ,
        Me.OleDbUpdateCommand1.CommandText = "UPDATE Events SET
CustomerID = ?, Location = ?, EventDate = ?, EventCostTotal = ?" & _
        " WHERE (EventID = ?) AND (CustomerID = ? OR ? IS NULL AND
CustomerID IS NULL) AN" & _
        "D (EventCostTotal = ? OR ? IS NULL AND EventCostTotal IS NULL)
AND (EventDate = " & _
        "? OR ? IS NULL AND EventDate IS NULL) AND (Location = ? OR ?
IS NULL AND Locatio" & _
        "n IS NULL)"
        Me.OleDbUpdateCommand1.Connection = Me.OleDbConnection1
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("CustomerID",
System.Data.OleDb.OleDbType.Integer, 0, "CustomerID"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Location",
System.Data.OleDb.OleDbType.VarWChar, 255, "Location"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("EventDate",
System.Data.OleDb.OleDbType.DBDate, 0, "EventDate"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("EventCostTotal",
System.Data.OleDb.OleDbType.Currency, 0, "EventCostTotal"))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventID", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_CustomerID1",
System.Data.OleDb.OleDbType.Integer, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "CustomerID", System.Data.DataRowVersion.Original, Nothing))
        Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventCostTotal",

```

```

System.Data.OleDb.OleDbType.Currency, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventCostTotal", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventCostTotal1",
System.Data.OleDb.OleDbType.Currency, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventCostTotal", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventDate",
System.Data.OleDb.OleDbType.DBDate, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventDate", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_EventDate1",
System.Data.OleDb.OleDbType.DBDate, 0,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "EventDate", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
    Me.OleDbUpdateCommand1.Parameters.Add(New
System.Data.OleDb.OleDbParameter("Original_Location1",
System.Data.OleDb.OleDbType.VarWChar, 255,
System.Data.ParameterDirection.Input, False, CType(0, Byte), CType(0,
Byte), "Location", System.Data.DataRowVersion.Original, Nothing))
    ,
    'DataGrid1
    ,
    Me.DataGrid1.DataMember = "Events"
    Me.DataGrid1.DataSource = Me.DsEventSearch1
    Me.DataGrid1.HeaderForeColor =
System.Drawing.SystemColors.ControlText
    Me.DataGrid1.Location = New System.Drawing.Point(32, 40)
    Me.DataGrid1.Name = "DataGrid1"
    Me.DataGrid1.PreferredColumnWidth = 90
    Me.DataGrid1.ReadOnly = True
    Me.DataGrid1.Size = New System.Drawing.Size(584, 192)
    Me.DataGrid1.TabIndex = 0
    ,
    'DsEventSearch1
    ,
    Me.DsEventSearch1.DataSetName = "DsEventSearch"
    Me.DsEventSearch1.Locale = New
System.Globalization.CultureInfo("en-US")
    ,
    'btnClose
    ,
    Me.btnClose.Location = New System.Drawing.Point(248, 256)
    Me.btnClose.Name = "btnClose"
    Me.btnClose.Size = New System.Drawing.Size(144, 40)
    Me.btnClose.TabIndex = 1
    Me.btnClose.Text = "Exit Search"
    ,
    'frmSearchEvent

```



```

    Me.AutoScaleBaseSize = New System.Drawing.Size(5, 13)
    Me.ClientSize = New System.Drawing.Size(640, 326)
    Me.Controls.Add(Me.btnClose)
    Me.Controls.Add(Me.DataGrid1)
    Me.Name = "frmSearchEvent"
    Me.Text = "Search for Events"
    CType(Me.DataGrid1,
System.ComponentModel.ISupportInitialize).EndInit()
    CType(Me.DsEventSearch1,
System.ComponentModel.ISupportInitialize).EndInit()
    Me.ResumeLayout(False)

End Sub

#End Region

'*****
'
'   Form Name:   frmSearchEvents
'
'   Purpose:     This form is used to search for events by using
'                a DataGrid of information.  The user can search
'                based on customer ID, location, ID number, or date.
'
'   Updated:     November 10, 2004
'   Updated By:  Matthew D. Johnson
'
'*****

Private Sub frmSearchEvent_Load(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles MyBase.Load
    'Command that is called when the form is loaded.  This command
fills the data grid with the data.
    DsEventSearch1.Clear()
    OleDbDataAdapter1.Fill(DsEventSearch1)
End Sub

Private Sub btnClose_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnClose.Click
    'Command that closes the window.  It is called by the user when
they click a command button.
    Close()
End Sub
End Class

```

Appendix K – Design Limitation Sheet

MSIS Project: Centralized Catering Information System Design Limitations Sheet

The following is a list of the **INCLUDED** functions for the program design:

- An easy to follow and understand Graphic User Interface.
- The ability to enter information into the program for storage
- The ability to retrieve stored information by using a search function. This function will use inputted criteria from the user.
- The ability to report results from searches to either the screen or to an output device, such as a printer.
- Complete documentation of program. This will be imbedded documentation within the final program itself.
- Notes about Included functions: The end user stresses that this program must be simple enough so that people with varying degree of education and technical expertise will be able to use it. The end user also asks the program perform only basic information storage and retrieval at this time. Additional functions are discussed as a future expansion of the project, but are not to be considered at this point in time.

The following is a list of what is **NOT INCLUDED** in the program design specs:

- The ability to share stored information between the program and another program.
- The ability to share information stored at one location with another location.
- Any online help. This will be provided by end-user via help desk.
- No assistance in entering past catering information will be provided for end user.
- Notes about Not Included functions: The end user will be using this completed program in a test setting at one location only initially. This is in order to allow the corporate technical support team to evaluate the program for any problems with other components on the established network. This is the case with all new programs prior to release corporate wide. Further functionality in the program has been discussed as a future update release of the program, but this is not something that is in anyway part of this initial development.

Appendix L – End User Survey

MSIS Project: Centralized Catering Information System End User Survey

Please take a few minutes to answer the following questions to the best of your ability or knowledge. The more information that you are able to provide will be greatly appreciated. Any question that does not pertain to you or your operation may be left blank. Thank you for participating in this voluntary survey. Your responses will be of invaluable assistance in this project.

1. How much catering business does your department do in an average quarter? What is the percentage of catering sales to total department sales?
2. Do you currently keep records of your past catering events? If yes, please continue with question 3. If NO, please skip to question 4.
3. What is your current method of maintaining records of your past caterings (i.e. Paper copies of forms, notebook, etc.)? Are there any problems with this storage method? How long do you keep records for?
4. Why do you not keep a record of past events? Do you feel that maintaining a record of your catering events would benefit your department in some way?
5. How do you rate your ability to use the stores computer systems? Would you be comfortable using the system on a more frequent basis in order to store catering information?
6. If you could design the perfect system for storing information on catering events, what would it be like? Think of things such as the amount of information that could be stored, ways of searching stored information, time frame for storing information, and any other features that you can think of. Please note that your perfect system does not have to be computer based and please be as complete in your answer as possible.

Thank you all for taking the time to answer these questions. Your input will be a huge help to me in completing this part of my schooling as well as being invaluable in designing this new system.

Thank you again for your corporation.

Matt Johnson
Sioux Falls #1 Kitchen

Appendix M – End User Survey Results

The end user survey was sent via e-mail to 45 different stores within the company. They were selected from a list of all of the stores that have reported doing some sort of catering sales in the last year on their profit and loss sheets.

These stores were selected to represent a range of stores from both larger urban area location as well as smaller rural stores. The stores were also selected to cover as much of the geographical area of our company as possible covering six of the seven states that we operate in.

Of the 45 surveys that were sent out, I received responses from 23 managers representing approximately 18% of the company's catering stores. Most were very helpful with their responses. I have summarized the results below.

- 1. How much catering business does your department do in an average quarter? What is the percentage of catering sales to total department sales?*

This is one area that stores varied in their responses widely based mostly on location. Stores reported catering sales that ranged from a few thousand dollars per quarter to one that does on average of \$80,000 per quarter. The average for the company over the last two years is just over \$24,000 per quarter.

The percentage of catering sales to total department sales again varied greatly. Most of the stores reported that catering made up approximately one quarter of their total department sales, and the average over all the company is slightly lower at 21.5%. The responses that we got from the survey represented a range from 9% up to 42%.

2. *Do you currently keep records of your past catering events? If yes, please continue with question 3. If NO, please skip to question 4.*

Of the stores that completed the survey, 15 do keep some sort of record of past catering business. The remaining 8 indicated that do not keep past records.

3. *What is your current method of maintaining records of your past caterings (i.e. Paper copies of forms, notebook, etc.)? Are there any problems with this storage method? How long do you keep records for?*

The 15 stores that maintain records indicated that they do maintain records indicated that they use primarily two different methods of recordkeeping. Eight stores maintain paper copies of the order forms. Four stores keep record of caters by writing down the information on

either note cards or in a notebook. The remaining three stores did not indicate what method they use, only that they did maintain records of some sort.

Nine of the 15 stores that store records indicated that they have some sort of problems with records. These included incomplete records, lack of storage space, records that were destroyed due to accidents or neglect, and others. The rest didn't say that they have had any problems with storage.

Records for events were kept for various time spans. This ranged from three months to 10 years. The average for the people that responded is 22 months.

4. *Why do you not keep a record of past events? Do you feel that maintaining a record of your catering events would benefit your department in some way?*

Out of the eight stores that responded that they do not keep past records, four said that it was due largely to concerns over storage space and time commitments that would be needed to track the information. Two additional stores said that they use to track that

information but have since stopped due to managerial changes or some other factor.

The last two stores indicated that they have never attempted to track caterings since they opened. (Both of these stores have only been opened less than two years.)

Only two of the eight stores that do not maintain records said that they felt it would be beneficial to their business.

5. *How do you rate your ability to use the stores computer systems? Would you be comfortable using the system on a more frequent basis in order to store catering information?*

It was no surprise in this day that 18 of the 23 people responding said that they were quite able to use the stores computer systems and did so on a regular basis. Three more managers said that while they were able to use the computer systems somewhat, they were somewhat shaky on anything more than the basics. The last two managers said that they were not good with the computer and had to have others help them with checking their mail and even responding to this message.

As far as the comfort level of people using the computer system to track catering information, 16 of the 23 said that they would be comfortable using a simple system to track the important information. Six additional people said that they were not sure if they wanted to go through the hassle of using a computer based system to track the information. The last person thought it was a bad idea. This is also the person that needed help getting their mail checked.

6. *If you could design the perfect system for storing information on catering events, what would it be like? Think of things such as the amount of information that could be stored, ways of searching stored information, time frame for storing information, and any other features that you can think of. Please note that your perfect system does not have to be computer based and please be as complete in your answer as possible.*

In a surprise to me, this is one area that people didn't seem to have much to say. Only 56% of the returned surveys had some sort of suggestions. These included suggestions for both the type of system they would like to see along with suggestions for what should be included in the system.

Most of the people responded that a computer based system would be preferred to a paper based system. This is easier for people to work

with as well as saves on having to store even more paper work than we already do. The responders also mentioned that they would like a system that is similar to the existing systems that we use at the store level already. This would hopefully mean less training would be needed.

The one unanimous suggestion for the system was that it needed to be easy to use. People did not want something that was going to take them two weeks to learn or that was going to require an hour of work each day to maintain. They also indicated that they wanted to keep the material in the system to a minimum. They didn't want a whole lot of extras that would only slow down work.

Overall I would have to say that I was very happy with the responses that I received from the survey. While I didn't get back the number that I was hoping to, I feel that the responses that I did get back were a good cross section of the company and would represent the overall feelings well.

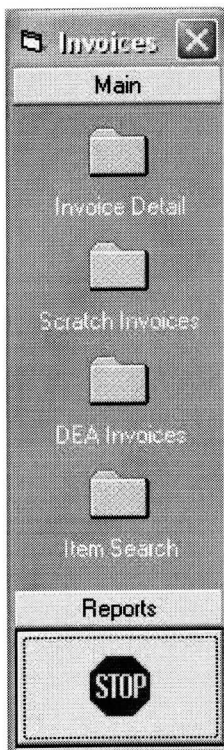
I was also quite pleased with the suggestions that I received from people. It showed me that people had an interest in what I was attempting to do and thought that it was worth some work and effort on both their part and mine. The feedback was most helpful in the design and construction of the final system.

Appendix N – Existing Company Programs Screen Shots

As I said earlier in this report one of the most requested design specs for this program is that it be easy for people to use. This is a must for a company that has people of such wide computer expertise levels. It was no different when I was planning out the cater tracking program.

In order to better plan for the actual design of the program, I took some screen shots of what existing programs at the store look like. You can see that they are very simple in design and are very easy to use.

Figure 9: Invoice Program Main Menu



This is the main startup menu of our online Invoice system. The system allows for the store users to access and print off copies of their invoices from most of our vendors. The system also allows for the user to search past invoices for a certain item.

Figure 10: Invoice Program Invoice Detail Screen

Invoice Number	Vendor	Delivery Date	PO Number	Warehouse Desc	Status	Department	Item Count
52307	00284500	12/20/04	0	LOMAR	Final	All Depts.	3
52307	00284500	12/20/04	0	LOMAR	Final	Health Market	3
584710	HYVEE	12/20/04	5778	CHARITON-GENERAL M...	Preliminary	All Depts.	227
584710	HYVEE	12/20/04	5778	CHARITON-GENERAL M...	Preliminary	HBC	189
584710	HYVEE	12/20/04	5778	CHARITON-GENERAL M...	Preliminary	Health Market	38
857625	HYVEE	12/20/04	5778	CHEROKEE-GROCERY	Preliminary	All Depts.	2
857625	HYVEE	12/20/04	5778	CHEROKEE-GROCERY	Preliminary	HBC	2
585873	HYVEE	12/19/04	5761	CHARITON-FROZEN	Preliminary	All Depts.	235
585873	HYVEE	12/19/04	5761	CHARITON-FROZEN	Preliminary	Dairy	24
585873	HYVEE	12/19/04	5761	CHARITON-FROZEN	Preliminary	Frozen	208
585873	HYVEE	12/19/04	5761	CHARITON-FROZEN	Preliminary	Health Market	3
585877	HYVEE	12/19/04	5761	CHARITON-CHS & RFG	Preliminary	All Depts.	388
585877	HYVEE	12/19/04	5761	CHARITON-CHS & RFG	Preliminary	Dairy	278
585877	HYVEE	12/19/04	5761	CHARITON-CHS & RFG	Preliminary	Health Market	5
585877	HYVEE	12/19/04	5761	CHARITON-CHS & RFG	Preliminary	Meat	105
3406265	00503100	12/18/04	5765	WELLS DAIRY	Final	All Depts.	2578
3406265	00503100	12/18/04	5765	WELLS DAIRY	Final	Dairy	2578
354461	00458201	12/18/04	5762	SRD-ANKENY	Final	All Depts.	220

** To view any warehouse invoices that are greater than 30 days old click on the binoculars button.

Figure 11: Invoice Program Search Program Screen

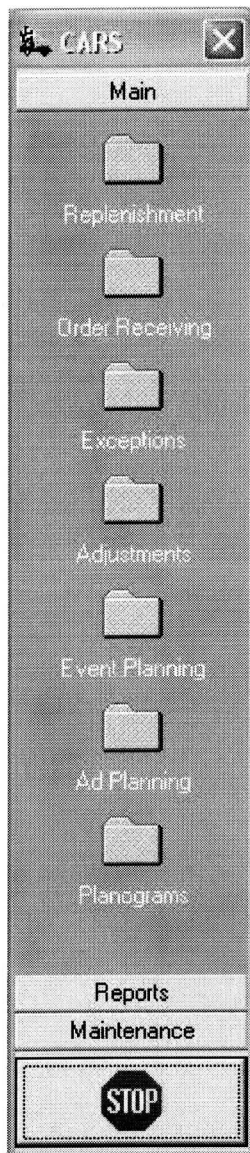
UPC Code:

Whse Code: Vendor:

Begin Date: End Date:

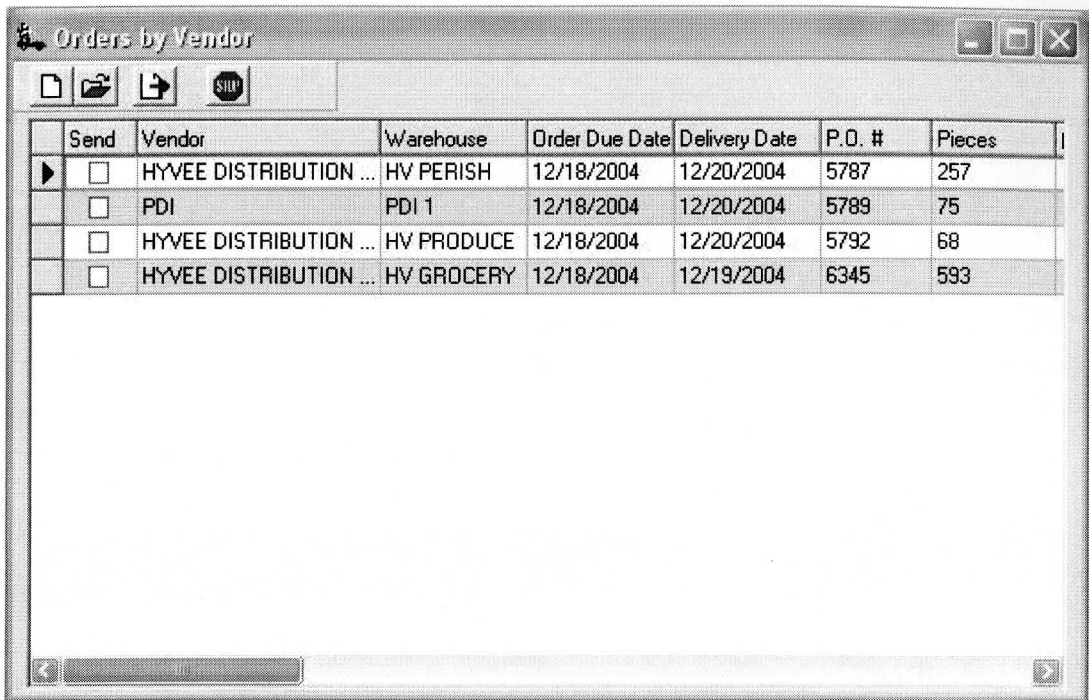
OK Cancel

Figure 12: CARS Program Main Startup Screen



This is the main startup screen from our Computer Automated Reorder System (CARS) program. This software helps with tracking our movement of items and reorders automatically when needed.

Figure 13: CARS Program New Order Screen



Send	Vendor	Warehouse	Order Due Date	Delivery Date	P.O. #	Pieces
<input type="checkbox"/>	HYVEE DISTRIBUTION ...	HV PERISH	12/18/2004	12/20/2004	5787	257
<input type="checkbox"/>	PDI	PDI 1	12/18/2004	12/20/2004	5789	75
<input type="checkbox"/>	HYVEE DISTRIBUTION ...	HV PRODUCE	12/18/2004	12/20/2004	5792	68
<input type="checkbox"/>	HYVEE DISTRIBUTION ...	HV GROCERY	12/18/2004	12/19/2004	6345	593

Figure 14: CARS Program Order Detail Screen

Order Form

Bolded items have had their SOQ changed from the programs original SOQ.

Vendor: PDI Warehouse: PDI 1 Dept: Total Pieces: 75

Summary Regular Ad

Drag a column header here to group by that column.

	SOQ	Whse Code	Description	Section	Size	Units Per Case	On Hand	On Order
▶	1	000000035689	BAP'S JUMBO JUMB BEEF FRANKS	50130	4/3	4	3	0
	1	000000040017	HY-VEE PICKLED PIMENTO LOAF	50120	16 OZ	12	9	0
	1	000000040018	HY-VEE GERMAN BOLOGNA	50120	16 OZ	12	18	0
	1	000000045018	OSCAR MAYER BEEF FRANKS	50130	16 OZ	24	45	0
	1	000000045102	OM VARIETY CLUB SANDWICH MEAT	50120	9 OZ	15	12	0
	1	000000045296	OM DELI SHAVED O/R TURKEY	50120	09OZ	12	6	0
	1	000000045432	OM DELI STYLE SMOKED TURKEY	50120	6 OZ	12	19	0
	1	000000085508	ROSINA ITALIAN MEATBALLS	50155	38 OZ	6	6	0
	1	000000090297	CARL BUDDING SLICED CORNED BF	50120	2.5 OZ	24	22	0
	1	000000090509	JOHNSONVILLE BEER BRATS	50130	19.76 OX	12	29	0
	1	000000090514	JOHNSONVILLE ITALIAN SAUSAGE	50130	19.76 OZ	12	8	0
	1	000000090522	JOHNSONVILLE BEDDAR W CHEDD...	50130	1#	10	15	0
	1	000000090568	JOHNSONVILLE HEAT N SERVE BRAT	50155	20 OZ	8	5	0
	1	000000090578	GOURMET DINING SHRIMP LINGUINI	50155	2LB	8	7	0
	1	000000090586	GOURMET DINING STIR FRY CHICKE	50155	2#	8	2	0
	1	000000090598	JIMMY DEAN HAM & CHEESE OMELET	50145	8.6OZ	8	2	0
	1	000000090641	JIMMY DEAN SAUS/EGG/CHEESE	50150	6CT	4	4	0

Figure 15: New Item Order Screen

Add Items to the Order

Department: [Dropdown]

Warehouse Code: [Text]

Quantity: [Text]

Enter warehouse code and press enter to add.

Figure 16: Online Catalog Order Form

The image shows a software dialog box titled "Search Catalog". It features a series of input fields for searching items in a catalog. The fields are: "Warehouse" (a dropdown menu), "Section Code" (a dropdown menu) and "To" (a dropdown menu), "Warehouse Code" (a text input field) and "To" (a text input field), "UPC" (a text input field), "Description" (a text input field), "Retail Department" (a dropdown menu), "Category" (a dropdown menu), "SubCategory" (a dropdown menu), "Private Label" (a dropdown menu), and "Department Supplies" (a dropdown menu). At the bottom of the dialog box, there are two buttons: "Search" and "Cancel".

The online catalog that we use is our access to all of the items that we carry in our warehouse. This is by far the easiest and fastest way to look up an item for someone.