Sabbatical Report

For Spring 1999

Shui-lien Huang

Computer Information Systems

Table of Contents

CIS38 Advanced Visual Basic Programming

International Database Management Project

Project Structure

SABBATICAL LEAVE PROPOSAL

Shui-lien Huang

With the rapid changes in the computer industry, our CIS Department has to modify the curriculum as well as expand the program to meet the needs of our students. Even though with the help of part-time faculty, our department is able to offer new courses to meet the demands, I am nevertheless convinced that our program would be stronger and healthier if more full-time faculty members also kept up with the trends in technology. This is the major reason that inspires my desire to take a sabbatical leave. By contributing my newly learned knowledge, my students will benefit from what I learn during my leave of absence.

With the ongoing changes of the software update and new textbooks for each class in every fall semester, it has been very difficult for me to find time to learn anything new other than to prepare material for courses I currently teach. To prepare myself in expanding my teaching skills and diversities, I would like to take advantage of my sabbatical leave by taking two computer classes.

Other than taking two computer classes, I also will have a joint project with Ms. Nancy King from the Counseling Department. Ms. King is in charge of creating a program for international students at Mt. SAC, while I will be responsible for designing a database management project and web pages for this program. Having easy accessibility of multi-language web pages for international students can provide information about Mt. SAC's history, admission procedures, and application requirements in their own languages. This is one of the least expensive means to recruit foreign students.

Part I -- Study Courses (Spring 1999):

A. **Course Title:** **CIS 38 Advanced Basic Programming (4 units)**

Mt. San Antonio College

Course Description: Advanced concepts using Visual Basic as the programming language. Designing, coding, testing, and implementing event-driven programs; creating and updating sequential and random files; validating input data; trapping errors; designing, displaying, searching, and updating database tables; creating recordset using SQL, producing business graphic; using OLE objects and DLLs; distributing applications.

Remarks: Visual Basic class has become a very popular course in our department. The need to add new sessions every semester indicates the increasing demand for the Advanced Visual Basic class. In addition, one of the full-time faculty members who is currently teaching the advanced class plans to retire soon. My growth in this advanced programming knowledge should provide the scheduling flexibility and staff support to the CIS Department.

B. Course Title: **CIS 33 Advanced Database Management - Microcomputers**

Mt. San Antonio College **(4 units)**

Course Description: Advanced programming techniques using Microsoft Access; building complete applications; interface design; handling errors; transaction processing; multi-user applications; client server; OLE; libraries; security.

Remarks: The lack of extensive knowledge of Access, a data management program, is one of the weakest points of our lab instructors and lab assistants. With the enhanced experience from this class, I will be able to assist our students better in the computer lab as well as in the classroom. My ability to share more knowledge in database management will raise the level of my instruction. This experience should also smooth the process of developing the database management program for the projected International Student Program, my joint venture with Ms. King.

Part II -- Joint Project (Spring 1999):

A. Database Management Project

Ms. Nancy King's database management program involves managing student's personal and academic records, analysis of her research study, and services available to international students, etc. The process of developing this database management program includes creating, testing, debugging, and implementing. The basic use of this program should be user-friendly in entering, updating, retrieving and analyzing student records. This program can help us to realize the impact we encounter from the foreign students in our college.

B. International Students Web Pages

The accessibility of multi-language web pages for international students allows them to obtain information about Mt. SAC prior to applying. Reading inquirer's own language on the Internet about our college should encourage the interest of foreign students. Mt. SAC will become more visible for potential international enrollments. Information about Mt. SAC's history, admission procedures, and application requirements are all a part of web pages. The emphasis on the international students shows that we care. This is one of the least expensive ways to recruit foreign students. Depending upon the extent of this program, I plan to spend at least six hours a week on my joint endeavor with Ms. King.

Part III -- Sabbatical Report:

A. Mt. San Antonio College transcript to verify the completion of CIS33 and CIS38 courses.

B. Project reports:

1. Printouts of the international student web pages.
2. Printouts of the international student database project.

Statement of Purpose

My one semester sabbatical leave project is divided into two sections. The first part is to develop web pages and a database management system for the international student services. The second part of this project is to increase my teaching expertise in Windows programming fields by attending classes in advanced Visual Basic and Access database management.

A. Study Courses - Windows Programming

To accommodate the increased beginning Visual Basic and Access classes, our CIS department needs to add more advanced classes for the two programs into our curriculum. Our advanced Visual Basic class Professor, Julia Bradley, has just retired recently, and the advanced Access Professor, Larry Brown, will be retiring next year. We are now in more need to have qualified staff to teach these advanced classes.

Another reason that I decided to take CIS33 Database Management – Access is to help me to design and develop the project for the international student services. The database management system will assist the international counselors in better record keeping.

B. Joint Projects for International Student Services

1. Database Management Project

   With the need to maintain international student' academic and personal information, Ms. Nancy King, the International Student Counselor, has asked me to develop a database management program for her.

2. International Students' Web Pages

   The students around the world are taking advantage of the Internet technology in information gathering about colleges and universities before making decisions to which to apply. Mt. SAC is competing with other colleges in recruiting more international student. An attractive, informational, and easy to follow international web ages would be helpful to serve this purpose. Adding foreign languages will definitely attract more viewers to the web site.

Advanced Access Database Management (CIS33)

Introduction

Access is a very powerful database management-developing tool. Taking this advanced Access class has helped me in designing and developing the international database management program for the international student services. With the knowledge I learned from this class, I would also be ready to teach this increasingly popular advanced Access class in the future.

Instrumentation

1. Hardware Used

   Office – IBM compatible 166 MHz computer, HP Deskjet 400

   Home – Compaq Deskpro 6000, HP Laserjet 4L

2. Software Used

   Microsoft Access 97

Course Contents

1. Developing tables, queries, forms, and reports
2. Using objects and properties
3. Designing event-driven program
4. Enhancing an application through macros, menus, and toolbars
5. Creating VBA modules, functions, and subroutines
6. Using objects and methods in VBA

7. Writing, debugging, and maintaining VBA statements

Course Assignments (Appendix A, p.49)

1. Chapter 1 – Tour of Access, Questions 1-16

   Database concepts; major characteristics of a database application; purposes of different Access windows; database utilities

2. Chapter 1 – Tour of Access, Questions 17-23

   Using Access Help; Switchboard Manager; purpose of some Access features

3. Chapter 3 – Using Objects and Properties, Questions 1-20

   Object-based vs. object-oriented programs; terminology of Access objects; DAO vs. Microsoft Access objects; collections; DBEngine; recordset vs. TableDef object; different recordset types

4. Chapter 4 – Event-Driven Programming and Access Events, Questions 1-18

   Differences among events, event properties, event procedures; events for opening and closing forms; events for updating records; sequence of events

5. Chapter 6 – VBA, Modules, Functions, and Subroutines, Questions 1-15

   Independent class modules vs. form or report class modules; standard modules; VBA (Visual Basic Application) statements; functions vs. sub procedures; passing arguments by reference or by value; event procedures; data types; variables and constants; arrays

Course Projects (See Appendix B, p.63)

1. Designing and creating linked tables for the database; creating forms and reports

   The first project creates two linked databases, one is called Service.mdb that contains only data, and the other is called EESS.mdb that contains program.

   Service.mdb: Creates System, SystemType, SystemSalesLocations, Staff, Sale, and ServiceQualifications tables in the file. Then, create relationships among tables.

   EESS.mdb: A form, a query, and a report are developed to display sales and system information.

2. Creating queries using the linked tables; creating macros and VBA procedures

   Developing a system that lists all the employees who are capable of servicing a particular machine. When the qualified employees are displayed on the screen, the form shows the name of the system they can service. To accomplish the task, a macro and a Visual Basic Application (VBA) procedure are created.

3. Developing a system to support system maintenance

   When customers need service, they provide the SystemID of the system that needs service. After the SystemID is entered, a form displays the sales details of the system. The form also displays "Under Warranty" label when the system is still under warranty. If it is under warranty, another form is displayed to show the names of service technicians who are qualified to service the system.

A VBA procedure is created to calculate and find out if the system is under warranty.

4. Using menus and toolbars to open and close forms

   Instead of using command buttons on the forms to open and close forms, the custom-made menus and toolbars are created for the project.

   The project uses macros to produce the same results as Project 3.

Result

I have applied the class material in designing the database management program for the international student services. On top of that, I am now very comfortable in teaching the beginning Access class in the coming semester and years to come. Helping more students in the computer lab is another benefit from taking the class.

Proof of completion of the class, the transcript - is attached. (Appendix D, p.138)

MT SAN ANTONIO COLLEGE
COURSE PREVIEW

Course Title: CIS 33 - Advanced ACCESS Database Mgt.

Instructor: Larry Brown

Required Text: Microsoft ACCESS (Advanced)
Author - Baldwin/Paradice
Publisher - Course Technology

Required Materials:
1. Three hole paper folders for submitting assignments
2. Three high density diskettes

Basis For Determining Semester Grades:

Mid Term Exam:	20%	A = 90 - 100 POINTS
		B = 80 - 89 "
Final Exam	20%	C = 70 - 79 "
		D = 60 - 69 "
Hands-On Projects	50%	F = 0 - 59 "
Quizzes	10%	

I will drop the quiz with the lowest grade

There will be no make up for missed mid-term exams. If the mid-term exam is missed, it will be given the same grade received on the final exam. If the final exam is missed, it will receive a grade of zero.

With the exception of the first class, I will not drop any student. If you want to drop this class, you must formally drop CIS 33 yourself.

The last day you may drop this (or any other class), with a "W" grade is _______ _______.
After this date you will receive a letter grade.

I will post class grades by your social security number in the ICC lab at the end of the semester. If you do not want your grades posted, let me know and we will arrange another method.

It will be necessary for you to use the bar code on your student ID to log in and out when you use the ICC lab.

CIS 33 is a hands-on course. If you fall behind, experience has shown that it is very difficult to catch up.

Weekly hands-on computer assignments will be accepted up to two weeks late with a 5 point penalty per week. After two weeks, late assignments will not be accepted.

CIS 33
ADVANCED ACCESS
COURSE OUTLINE

Overview of the course
Explanation of CTI databases
Loading CTI Advanced Access Databases
Exploring and learning CTI databases, Queries, Forms Reports, Macros, and Modules
Building an in-class database application

- Building the database and tables
- Queries
- Forms
- Reports

PROJECT ONE

Using Objects and Properties
Data Access Objects

PROJECT TWO

Event Driven ACCESS Programming

- Using events to display and update your database

PROJECT THREE

Macros

PROJECT FOUR

Introduction to Visual Basic for Application Programming (VBA)
Building Sub Procedures
Types of VBA Declarations and Constants
VBA Functions

HANDS-ON MID-TERM EXAM

PROJECT FIVE

VBA Objects and Methods
Object Variables
Methods and Access Objects
Data Access Object (DAO) Methods
Non-DAO Objects
OLE Object Variables and Methods

PROJECT SIX

VBA Programming Constructs
Data Integrity in a Multi-User Environment

PROJECT SEVEN

Final open subject area. Student needs will determine subject
Industry Strength Database Management
HANDS-ON FINAL EXAM

Advanced Programming in Visual Basic (CIS38)

Introduction

Visual Basic classes have been very popular in our department. As the beginning classes grow over the years, more advanced Visual Basic classes are now in demand. With our short staffing in this area, CIS department needs to be ready to fill in the gap. By taking this advanced Visual Basic class which was offered in spring 1999 and was taught by Professor Anita Millspaugh, I would be equipped to teach this advanced class as needed.

Instrumentation

1. Hardware Used

   Office – IBM compatible 166 MHz computer, HP Deskjet 400

   Home – Compaq Deskpro 6000, HP Laserjet 4L

2. Software Used

   Microsoft Visual Basic version 6

Course Contents

1. Creating MDI applications
2. Updating sequential and random files
3. Managing programmer defined objects in a collection
4. Incorporating routines for error checking

5. Navigating and updating database files using data control methods, SQL, and DAO

6. Creating custom controls

7. ActiveX

8. Creating help files

9. Distributing your application – packaging and deployment

Course Projects (Appendix C, p.83)

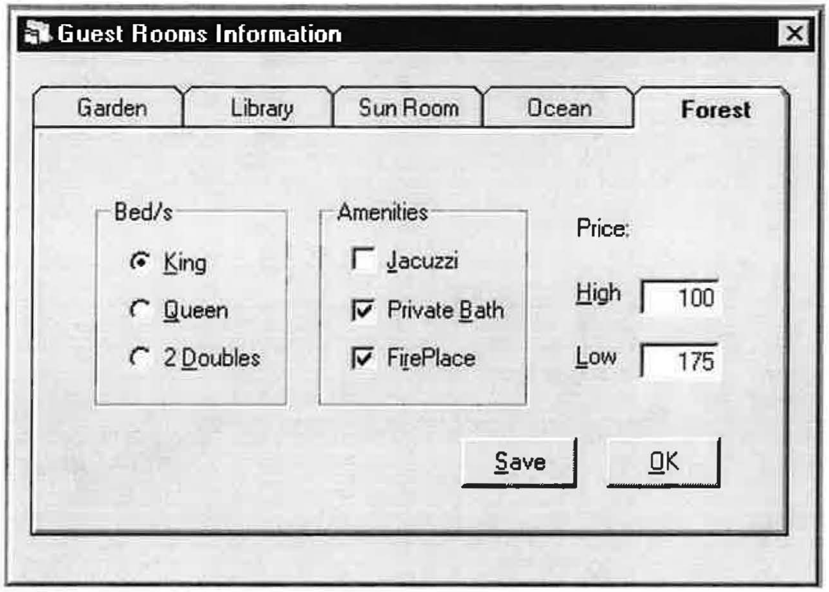
1. Creating an MDI project

   The project creates three menus for Huang Bed & Breakfast Reservation System.

File	Edit	Help
Exit	Guests	About
	Rooms	
	Reservations	

2. Using a sequential file to store and retrieve information

   This project creates a form displayed with several Tabs showing rooms information. When Rooms command in the Edit menu is clicked, the following user interface is displayed with the appropriate information from the Rooms sequential file.



Bed/s

● King
○ Queen
○ 2 Doubles

Amenities

☐ Jacuzzi
☑ Private Bath
☑ FirePlace

Price:

High 100
Low 175

3. Creating objects; using a random file to store and retrieve information

This project creates objects to store and retrieve guest information using a random file. The guest data will be displayed when the user clicks the Guests command from the Edit menu. Error trapping techniques are used in this project.

4. Using a data control to access a database file

Instead of working with a sequential or a random file, this project takes the advantages of working with a database file. A Visual Basic data control is included in the project to access guest's data. The users would be able to add, save, delete, and navigate the records from the database file. This

Visual Basic project displays guest's data that is saved from a Microsoft Access database file.

5. Searching information from a database file using SQL

   Reservation is the topic of the project. When the user clicks the Reservations from the Edit menu, the Reservation form will be displayed on the screen. Once a guest name is entered, the system displays the Guests database file for the user to enter more data if this is a new guest. Otherwise, the system will check for the availability of the rooms based upon the date requested.

   The program contains options such as Calendar, Car Rental, and Tour Information for the convenience for the user.

   SQL search technique is used to search for the available rooms.

6. Creating, displaying, and releasing objects in a DLL component

   This is an independent project from the Huang Bed & Breakfast Reservation System. The project allows the user to create, display, and releasing objects created in a DLL component.

7. Creating an ActiveX document

   The project creates an ActiveX document that will contain one UserDocument to calculate snowboard rental charges in Microsoft Internet Explorer browser. This little project shows how easy an Internet application can be created using ActiveX Control.

8. Accessing data from a remote file

   It demonstrates how Visual Basic can display data from a remote AS400 COBOL data file using ODBC connection. The project adds a help menu that displays the user-defined help file.

9. Presentation – using the FlexGrid control

   The presentation I did was on the FlexGrid control that was not covered in the class. Using this control in a Visual Basic application allows complete flexibility to sort, merge, and format tables.

Result

The in depth advanced Visual Basic programming class has given me a broader view of the Visual Basic product. A Visual Basic application not only can link to an Access database file, but also can link to a remote database file through ODBC. Internet application can also be created by using Visual Basic ActiveX control. After finishing the class, I am now ready to satisfy the demand for the advanced Visual Basic class. In addition, I am sure more students in our computer lab can benefit from my knowledge in Visual Basic.

Proof of completion of the class, the transcript - is attached. (Appendix D, p.138)

Advanced Programming in Visual Basic
Course Syllabus

Course Number:	CIS 38
Professor:	Anita Millspaugh
Office:	26B-202A; x4662
email:	amillspa@mtsac.edu

Text: Programming in Microsoft Visual Basic Version 6.0
Bradley and Millspaugh

Materials: 2 3 1/2" disks

Grading:

90-100	A	Programs	50%
80-89	B	Midterm	20%
70-79	C	Quizzes	10%
60-69	D	Final	20%

All assignments are due at the beginning of class. Late work will lose 10 points per week, early assignments will earn 10 points per week.

Regular attendance is strongly encouraged. It is your responsiblity to drop this class if that is your intention. No drops may be made after the 14th week.

COURSE OBJECTIVES:

1. Create an MDI application.
2. Store programmer defined objects in a collection.
3. Update sequential and random files.
2. Incorporate routines for error checking.
3. Navigate and update database files using data control methods, SQL, and DAO.
4. Create custom controls.

Advanced Programming in Visual Basic
Tentative Schedule

Week	Chapter	Topic
1	15	MDI; Creating a system - planning and design
2	9,10	Objects ; Sequential files updates
3	9	Containers
4	10	On Error Routines
5-6	10	Random file updates
7	11	Navigating database files
8	12	Updating database files using database methods
9		Midterm
10-11	12	Updating database files using SQL
12	12	Data validation with database files
13-14		DAO
15		ODBC to the AS400
16	14	Creating Custom Controls
17		Final Exam

International Students Database Project

Introduction

A student database management program is crucial for the international office to enter, update, retrieve, and print the student information. Through collaboration with Ms. Nancy King, Mt. SAC's International Student Counselor, we developed the contents for this project. Due to time constrains, this phase of the project includes only retrieving data and generating reports.

Instrumentation

1. Hardware Used

   Office – IBM compatible 166 MHz computer, HP Deskjet 400

   Home – Compaq Deskpro 6000, HP Laserjet 4L

2. Software Used

   Microsoft Access 97, Microsoft PowerPoint 97

   I took the 18-week CIS33 (Microcomputer Database Management) class and gained the knowledge of how to design and manage data in Access.

3. Source Data

   Ms. Nancy King gathered the student information through counseling course COUN58. Both of us spent time to enter the data together.

Design

The project is divided into two linked files. One is called Students Data.mdb, which contains seven tables. The tables are Academic, Goal, Immigration, LivingArrangement, Major, Personal, and Transportation. This file contains only the students' data. The other file is called Students.mdb, which contains the programs – queries, forms, reports, and macros. The benefit for separating the project into two files is that the user can change only the data file, not the program file. It is also easier for the programmer to modify the program file without the need to access the student data.

Users

International Students Counselors – Nancy King and Stanley Mbuthi

The project is designed according to the counselors' need to manage the international students' academic, personal, and immigration data.

User-friendly Project

In order for the user to run the project easily, a shortcut icon is created on the user's desktop. The user can click the icon and the project will be executed automatically in Access.

Program Structure

1. Main Form

   The program begins with a startup form showing the main screen. The user can choose one of the buttons to access information on Academic, Personal, Immigration, or Find Student by Social Security Number. (Form 1, p.25)

2. Academic Form

   On the Academic Form, the user can choose one of the buttons to find out students' goal, major, whether or not students are graduated from high school, and the college units they have completed. (Form 2, p.25)
   See (Academic Output 1-3, p.27) for the sample results of the selections.
   The Academic Report button will display a form for the user to select the different kinds of report on academic information.

3. Personal Form

   On the Personal Form, the user can choose one of the buttons to display students by age, gender, country, language, transportation, and living agreement. (Form 3, p.26) See (Personal Output 1-2, p.29) for sample results of the selections.
   The Personal Report button will display a form for the user to select the different kinds of report on personal information.

4. Immigration Form

The Immigration Form displays or prints the students' immigration data. (Form 4, p.26) (Immigration Report Output, p.34)

5. Find Student by SSN

The form allows the user to select a social security number from a combo box list. (Find Student Form, p.30) Then, the corresponding student is displayed with academic information. (Find Student Output 1, p.30) The user can also click the Personal button from this form to display the personal information on this student. (Find Student Output 2, p.31)

6. Academic Report

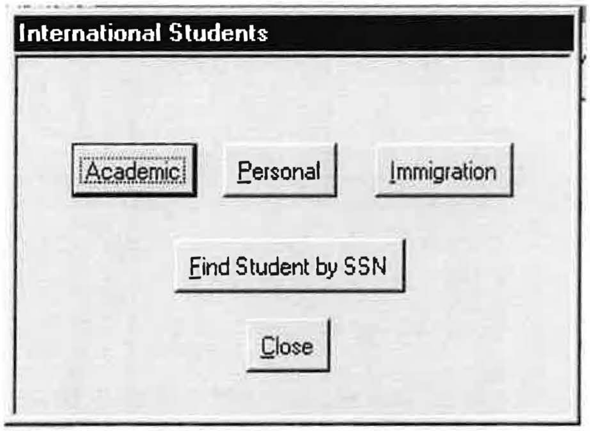
The Academic Report form allows the user to select different academic reports based on the order of ID, Last Name, Major, Goal, or Units. (Form 5, p.27) (Academic Report Output, p.32)

7. Personal Report

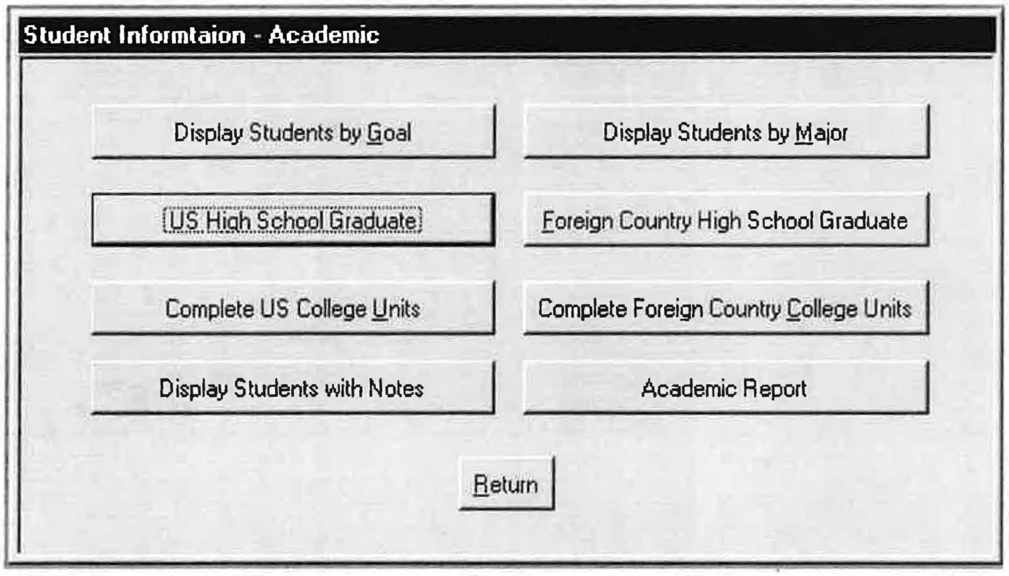
The Personal Report form allows the user to select different personal reports based on the order of ID, Last Name, Gender, Phone, or Language. (Form 6, p.27) (Personal Report Output, p.33)

Result

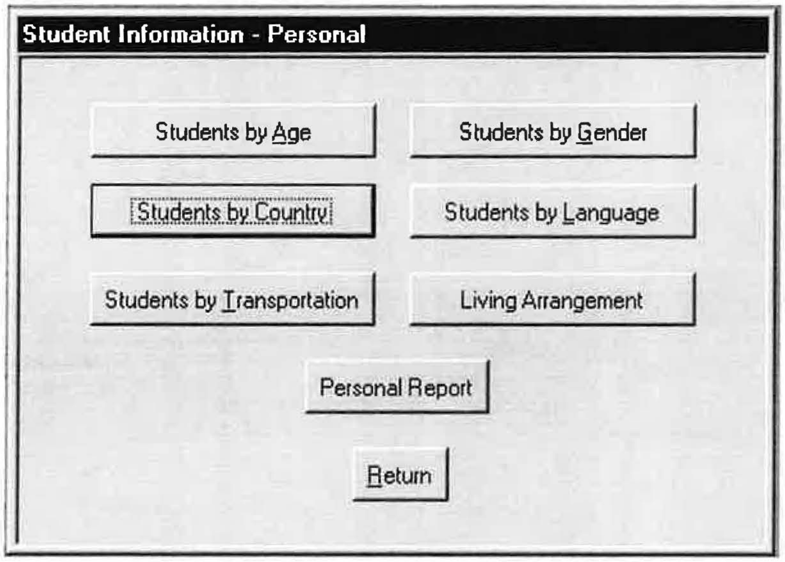
The international database project is stored in a 3 ½" disk. (Appendix F, p.142)


International Students
Academic
Personal
Immigration
Find Student by SSN
Close

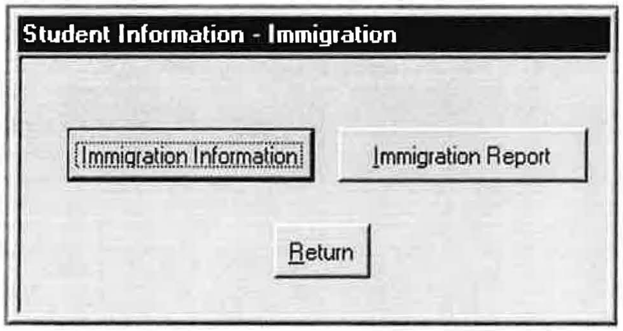
Form 1


Student Informtaion - Academic
Display Students by Goal
Display Students by Major
US High School Graduate
Foreign Country High School Graduate
Complete US College Units
Complete Foreign Country College Units
Display Students with Notes
Academic Report
Return

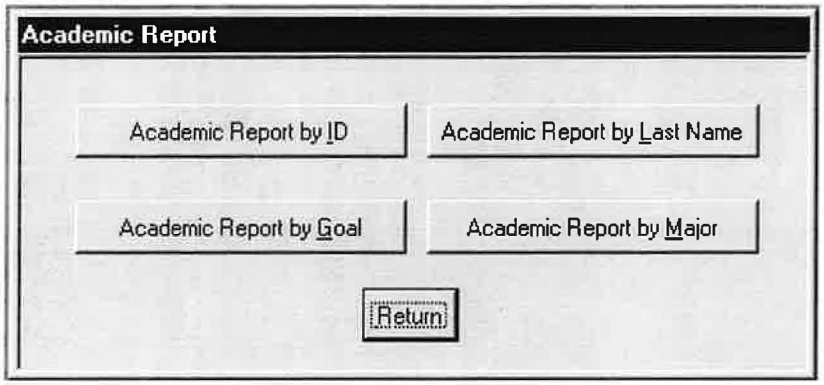
Form 2


Student Information - Personal
Students by Age
Students by Gender
Students by Country
Students by Language
Students by Transportation
Living Arrangement
Personal Report
Return

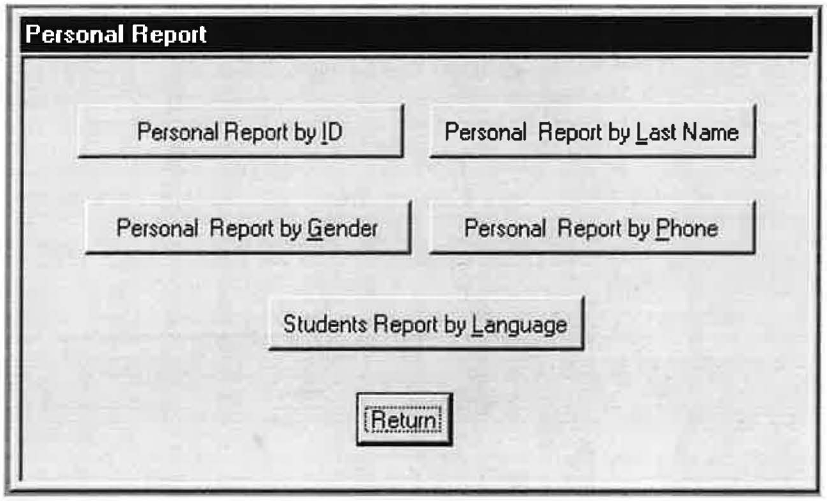
Form 3


Student Information - Immigration
Immigration Information
Immigration Report
Return

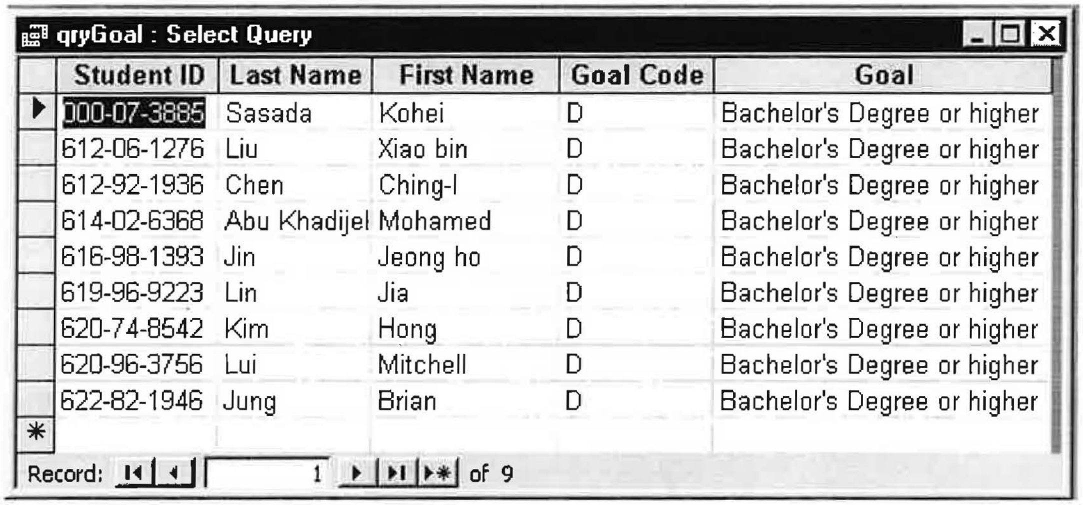
Form 4


Academic Report
Academic Report by ID
Academic Report by Last Name
Academic Report by Goal
Academic Report by Major
Return

Form 5

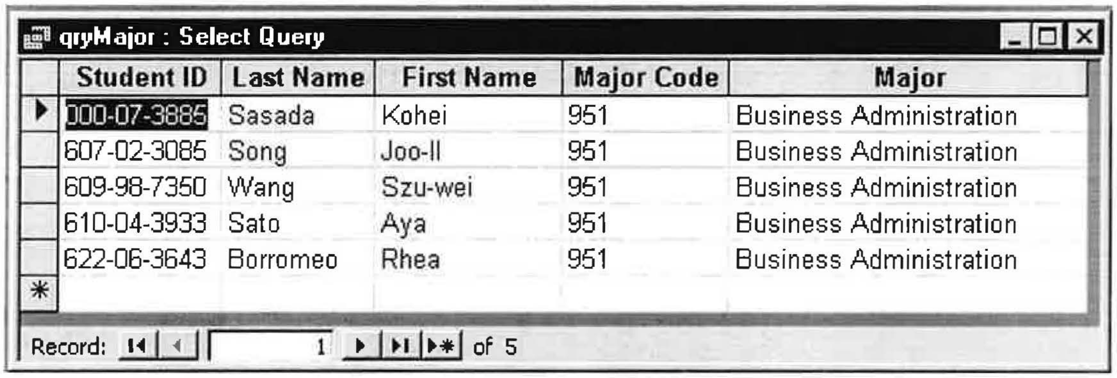

Personal Report
Personal Report by ID
Personal Report by Last Name
Personal Report by Gender
Personal Report by Phone
Students Report by Language
Return

Form 6



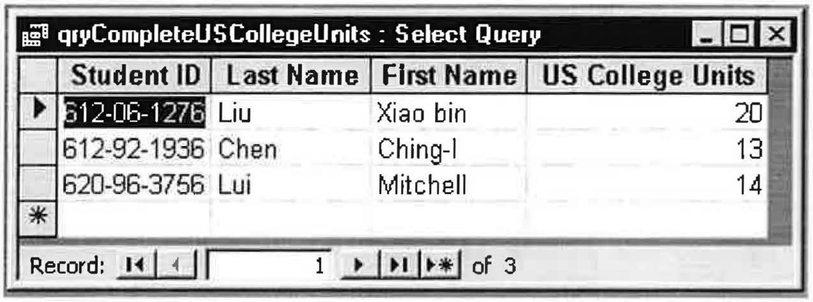
Student ID	Last Name	First Name	Goal Code	Goal
000-07-3885	Sasada	Kohei	D	Bachelor's Degree or higher
612-06-1276	Liu	Xiao bin	D	Bachelor's Degree or higher
612-92-1936	Chen	Ching-I	D	Bachelor's Degree or higher
614-02-6368	Abu Khadijel	Mohamed	D	Bachelor's Degree or higher
616-98-1393	Jin	Jeong ho	D	Bachelor's Degree or higher
619-96-9223	Lin	Jia	D	Bachelor's Degree or higher
620-74-8542	Kim	Hong	D	Bachelor's Degree or higher
620-96-3756	Lui	Mitchell	D	Bachelor's Degree or higher
622-82-1946	Jung	Brian	D	Bachelor's Degree or higher

Academic output 1 – select students by Goal (goal code "D")



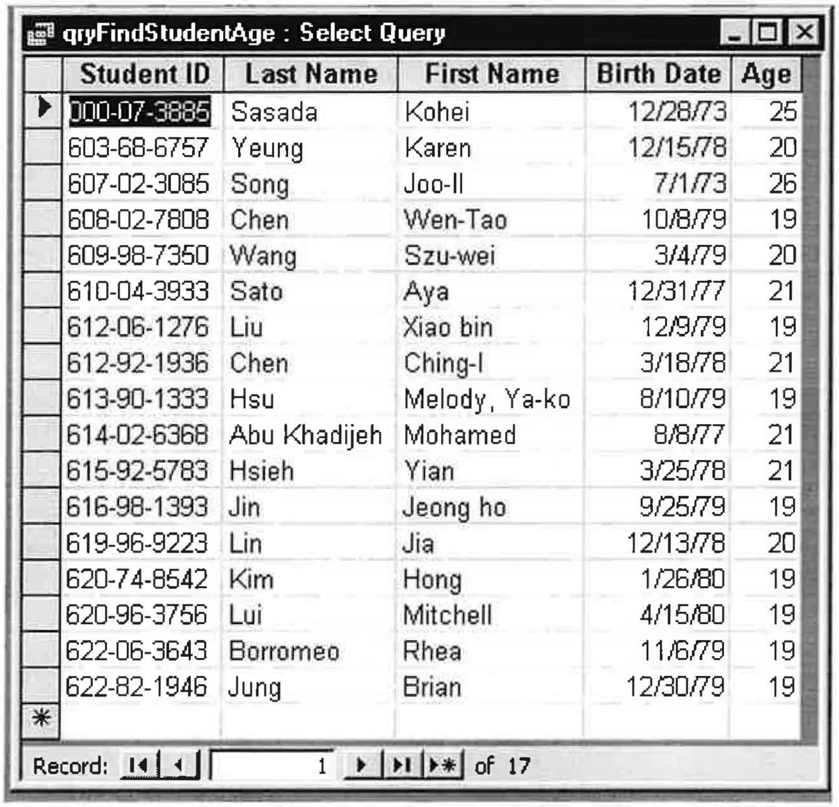
Student ID	Last Name	First Name	Major Code	Major
000-07-3885	Sasada	Kohei	951	Business Administration
607-02-3085	Song	Joo-Il	951	Business Administration
609-98-7350	Wang	Szu-wei	951	Business Administration
610-04-3933	Sato	Aya	951	Business Administration
622-06-3643	Borromeo	Rhea	951	Business Administration

Academic output 2 – select students by Major (major code "951")



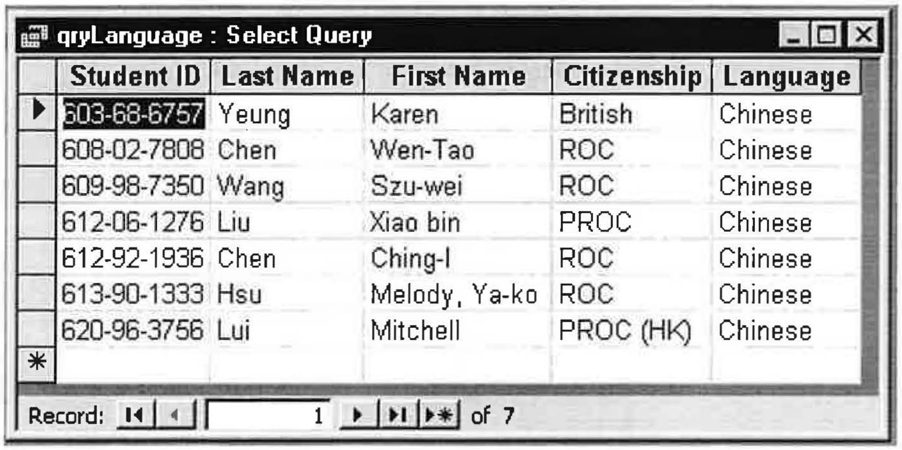
Student ID	Last Name	First Name	US College Units
612-06-1276	Liu	Xiao bin	20
612-92-1936	Chen	Ching-I	13
620-96-3756	Lui	Mitchell	14

Academic output 3 – US college units completed



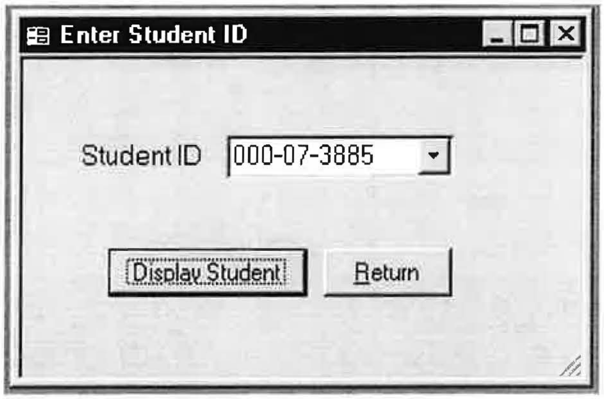
Student ID	Last Name	First Name	Birth Date	Age
000-07-3885	Sasada	Kohei	12/28/73	25
603-68-6757	Yeung	Karen	12/15/78	20
607-02-3085	Song	Joo-Il	7/1/73	26
608-02-7808	Chen	Wen-Tao	10/8/79	19
609-98-7350	Wang	Szu-wei	3/4/79	20
610-04-3933	Sato	Aya	12/31/77	21
612-06-1276	Liu	Xiao bin	12/9/79	19
612-92-1936	Chen	Ching-I	3/18/78	21
613-90-1333	Hsu	Melody, Ya-ko	8/10/79	19
614-02-6368	Abu Khadijeh	Mohamed	8/8/77	21
615-92-5783	Hsieh	Yian	3/25/78	21
616-98-1393	Jin	Jeong ho	9/25/79	19
619-96-9223	Lin	Jia	12/13/78	20
620-74-8542	Kim	Hong	1/26/80	19
620-96-3756	Lui	Mitchell	4/15/80	19
622-06-3643	Borromeo	Rhea	11/6/79	19
622-82-1946	Jung	Brian	12/30/79	19

Personal output 1 – Students' age



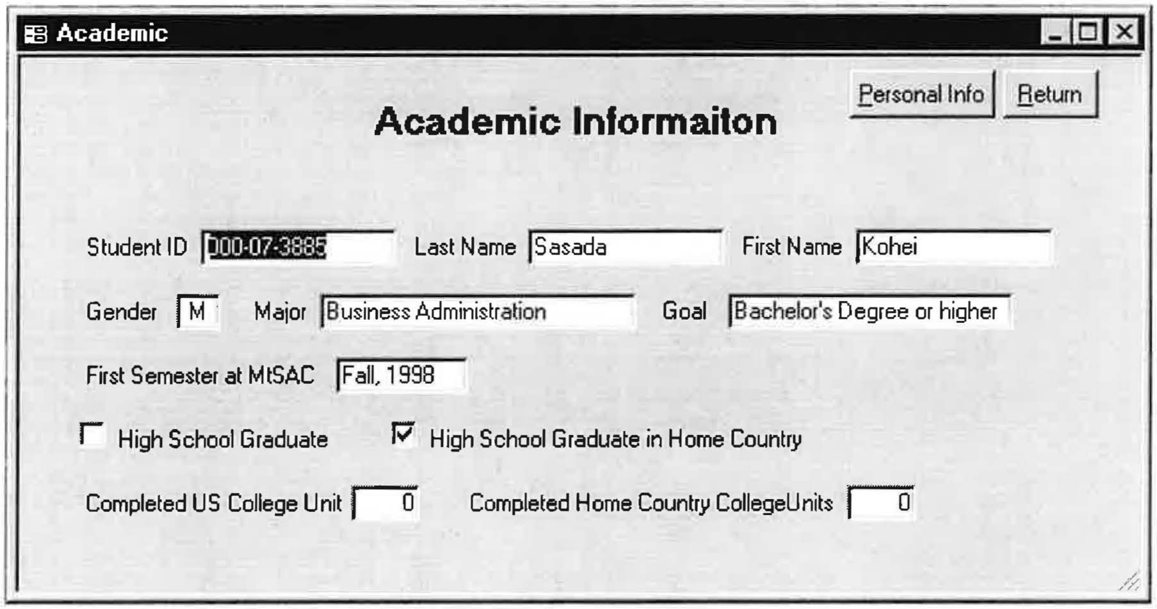
Student ID	Last Name	First Name	Citizenship	Language
603-68-6757	Yeung	Karen	British	Chinese
608-02-7808	Chen	Wen-Tao	ROC	Chinese
609-98-7350	Wang	Szu-wei	ROC	Chinese
612-06-1276	Liu	Xiao bin	PROC	Chinese
612-92-1936	Chen	Ching-I	ROC	Chinese
613-90-1333	Hsu	Melody, Ya-ko	ROC	Chinese
620-96-3756	Lui	Mitchell	PROC (HK)	Chinese

Personal output 2 – select student by language ("Chinese" is entered)



Student ID 000-07-3885

Find Student by SSN form



Academic Informaiton

Student ID 000-07-3885 Last Name Sasada First Name Kohei

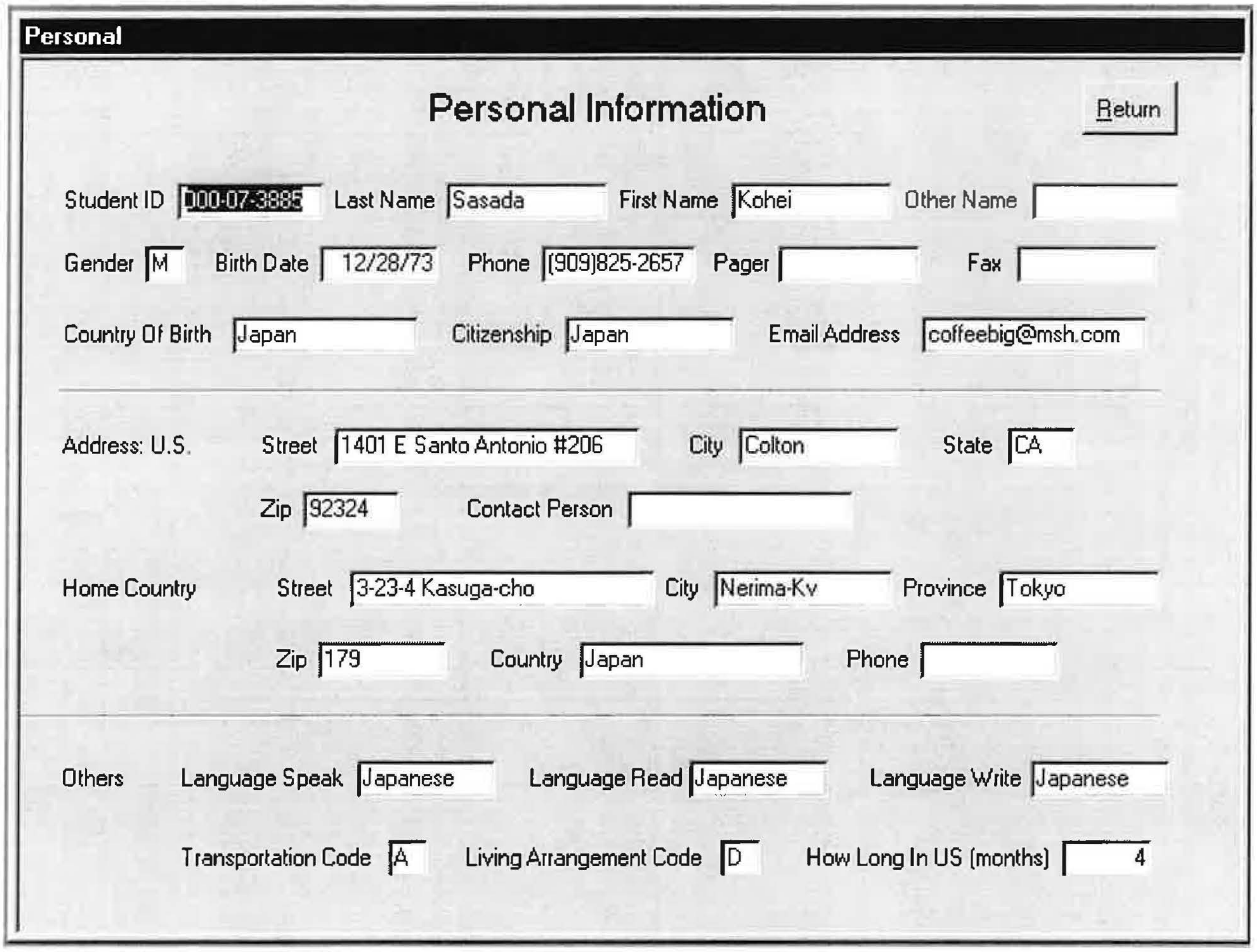
Gender M Major Business Administration Goal Bachelor's Degree or higher

First Semester at MtSAC Fall, 1998

□ High School Graduate ☑ High School Graduate in Home Country

Completed US College Unit 0 Completed Home Country CollegeUnits 0

Find Student by SSN output 1



Personal Information

Student ID 000-07-3885 Last Name Sasada First Name Kohei Other Name

Gender M Birth Date 12/28/73 Phone (909)825-2657 Pager Fax

Country Of Birth Japan Citizenship Japan Email Address coffeebig@msh.com

Address: U.S. Street 1401 E Santo Antonio #206 City Colton State CA

Zip 92324 Contact Person

Home Country Street 3-23-4 Kasuga-cho City Nerima-Kv Province Tokyo

Zip 179 Country Japan Phone

Others Language Speak Japanese Language Read Japanese Language Write Japanese

Transportation Code A Living Arrangement Code D How Long In US (months) 4

Find student by SSN output 2

Academic Report by Major

Major	*Student ID*	*Name*	*Gender*	*First Semester*	*Units*
Business Accounting	612-06-1276	Liu, Xiao bin	F		20
Business Administration	610-04-3933	Sato, Aya	F	Fall, 1998	0
	000-07-3885	Sasada, Kohei	M	Fall, 1998	0
	609-98-7350	Wang, Szu-wei	F	Fall. 1998	0
	622-06-3643	Borromeo, Rhea	F	Fall, 1998	0
	607-02-3085	Song, Joo-Il	M	Fall, 1998	0
Business, transfer	608-02-7808	Chen, Wen-Tao	M	Fall, 1998	0
Communication	615-92-5783	Hsieh, Yian	M	Fall, 1998	0
	612-92-1936	Chen, Ching-I	F	Fall, 1997	13
Computer Information System	614-02-6368	Abu Khadijeh, Mohamed	M	Fall, 1998	0
Computer Science	619-96-9223	Lin, Jia	F	Fall, 1998	0
Engineering	620-96-3756	Lui, Mitchell	M	Fall, 1998	14
Small Business Management	603-68-6757	Yeung, Karen	F	Fall, 1998	0
Undecided	613-90-1333	Hsu, Melody, Ya-ko	F	Fall, 1998	0
	622-82-1946	Jung, Brian	M	Fall, 1998	0
	616-98-1393	Jin, Jeong ho	M	Fall, 1998	0
	620-74-8542	Kim, Hong	M	Fall, 1998	0

Personal Report By Last Name

Student ID	Name	Gender	Phone	Address	Language
614-02-6368	Abu Khadijeh, Mohamed	M	(626)915-9997	1408 N. Grand Ave Covina, CA 91724	Arabic
622-06-3643	Borromeo, Rhea	F	(626)964-2914	19017 E. Hollingworth St. West Covina, CA 91792	
608-02-7808	Chen, Wen-Tao	M	(626)333-5637	2790 Jurado Ave. Hacienda Hts., CA 91745	Chinese
612-92-1936	Chen, Ching-I	F	(626)732-0268	1414 N. Grand Ave. Covina, CA 91724	Chinese
615-92-5783	Hsieh, Yian	M	(562)694-3489	18225 E. Wellington Ln. Rowland Heights, CA 91748	Chinese
613-90-1333	Hsu, Melody, Ya-ko	F	(626)965-0867	17800 E Colima Rd. #566 Rowland Heights, CA 91748	Chinese
616-98-1393	Jin, Jeong ho	M	(909)396-4700	1826 Los Cerros Dr. Diamond Bar, CA 91765	Korean
622-82-1946	Jung, Brian	M	(626)912-9944	19050 Colima Rd. #217 Roland Heights, CA 91748	Korean
620-74-8542	Kim, Hong	M	(909)860-5939	1328 Diamond Bar Blvd. #B Diamond Bar, CA 91765	Korean
619-96-9223	Lin, Jia	F	(909)594-7816	310 Abogado Ave. Walnut, CA 91789	Chinese
612-06-1276	Liu, Xiao bin	F	(626)810-5286	2339 Nogales Street Roland Heights, CA 91748	Chinese
620-96-3756	Lui, Mitchell	M	(626)330-3729	2441 Mountain Brook Dr. Hacienda Heighs, CA 91789	Chinese
000-07-3885	Sasada, Kohei	M	(909)825-2657	1401 E Santo Antonio #206 Colton, CA 92324	Japanese
610-04-3933	Sato, Aya	F	(909)469-1434	1512 W. Mission Blvd #17 Pomona, CA 91766	Japanese
607-02-3085	Song, Joo-Il	M	(909)899-8661	8403 Etinanda Ave. #A Rancho Cucamong, CA 91734	Korean
609-98-7350	Wang, Szu-wei	F	(626)913-5028	17800 E. Colima Rd. #528 Roland Heights, CA 91748	Chinese

Immigration Report

Student ID	*Name*	*Gender*	*I-20 Current*	*Passport Current*	*Visa Current*
000-07-3885	Sasada, Kohei	M	☑	☑	☑
603-68-6757	Yeung, Karen	F	☑	☑	☑
607-02-3085	Song, Joo-Il	M	☑	☑	☑
608-02-7808	Chen, Wen-Tao	M	☑	☑	☑
609-98-7350	Wang, Szu-wei	F	☑	☑	☑
610-04-3933	Sato, Aya	F	☑	☑	☑
612-06-1276	Liu, Xiao bin	F	☑	☑	☑
612-92-1936	Chen, Ching-I	F	☑	☑	☑
613-90-1333	Hsu, Melody, Ya-ko	F	☑	☑	☑
614-02-6368	Abu Khadijeh, Mohamed	M	☑	☑	☑
615-92-5783	Hsieh, Yian	M	☑	☑	☑
616-98-1393	Jin, Jeong ho	M	☑	☑	☑
619-96-9223	Lin, Jia	F	☑	☑	☑
620-96-3756	Lui, Mitchell	M	☑	☑	☑
622-06-3643	Borromeo, Rhea	F	☑	☑	☑
622-82-1946	Jung, Brian	M	☑	☑	☑

International Student Web Pages

Introduction

Use of Internet has become one of the most popular tools for information gathering. Reading in the student's native language about Mt. SAC will definitely attract more foreign students' attention. With the development of Mt. SAC web pages in native languages of our international student population, the history of Mt. SAC, admission procedures, application requirements counseling services and the pictures of our beautiful campus can be retrieved without difficulties.

Instrumentation

1. Hardware Used

   Office – IBM compatible 166MHz Computer, HP Deskjet 400

   Home – Compaq Deskpro 6000, HP scanner, HP LaserJet 4L.

2. Software Selection

   It was difficult for me to select between FrontPage and PageMill as the web design software. With the concern that our web pages will be under the main server, and the suggestion from Sheryl Hullings (Information Technology Department), I concluded that PageMill will be more appropriate for the purpose of this project.

In order to be proficient and knowledgeable of PageMill, and with the permission from instructor Mr. Joe Vasquez, I sat through a 9-week Comp 13 (Using Web Page Software) class.

Other software used:

- Windows NT Workstation 4.0
- Adobe PhotoDeluxe Business Edition 1.0
- Microsoft Paint 4.0
- Microsoft Photo Editor 3.01
- Netscape Navigator 3.1 and Internet Explorer 5.0 browsers
- Microsoft Word 97 in both English and Chinese versions
- NJStar Communicator

Data Source

The Initial contents of the web pages were provided by Ms. Nancy King – Mt. SAC's International Student Counselor. The web pages you are going to see were concluded after hours of discussion and then finalized.

Procedures

1. General

    a. Overall Design

    In order to organize the pages into two languages, a home page was created to display only the title and the choices of the English or Chinese languages. Once a language is chosen, the user will see the next page with three frames (Table 1). From this page on, all the pages include title frame, table of contents frame, and the contents display frame corresponding to what item is selected from the table of contents. The table of contents frame also contains a Home button to return to the home page.

Title frame	
Table of Contents	Contents display frame

Table 1 – Web Page Design

b. Numbered Paragraphs

When a page contains numbered paragraphs, they don't line up nicely in PageMill, unless a table method is used. Using a table of two columns for the numbered paragraphs will solve the problem. The first column is used to enter the numbers.

c. Background Colors

Numerous hours were spent in selecting the background colors. To give a lively appearance, the Spectra color was used for the home page, the title and table frames. With the consideration that the user can print the contents easily, a solid light yellow color is chosen.

Different display resolutions affect the Spectra color output. To achieve consistent Spectra color on different resolution monitors, Adobe PhotoDeluxe software is used to change the size of the original image.

d. Scanning Pictures

Three campus photos from one of Mt.SAC brochures (Appendix E, p.139) are used in the web pages. By default, the scanning images appear huge on the screen. After many hours and lots of trial and error, the appropriate size and resolution of the final images are implemented.

e. The Home Button

After drawing several "Home" buttons myself, I decided to search a more professional looking button from the web. Once the button (it has the two words Home Button as text) was downloaded, Microsoft Paint and AdbobeDeluxe were used to remove the second word and the background behind it.

2. Chinese Section

1. Chinese Characters Viewer

In order for Chinese characters to be readable in an English system environment, the use of NJStar Communicator is necessary. This software allows the users to work with some Asian languages in an English system only environment.

2. Displaying Chinese Documents in PageMill

   Chinese Word documents are not exactly compatible with PageMill. Using copy and paste method between Word and PageMill gives the results of unreadable characters. Open the Word document directly from PageMill has the similar results. After hours of research and trials, I found out that the Word documents needed to be converted into html files before these files can be read in PageMill. To keep the text clean and acceptable, it is also necessary to delete the question marks that are embedded in the converted file.

3. Chinese Button

   The Chinese button on the home page was downloaded from the web, and then modified through Microsoft Photo Editor.

4. Users Have Problems to View Chinese

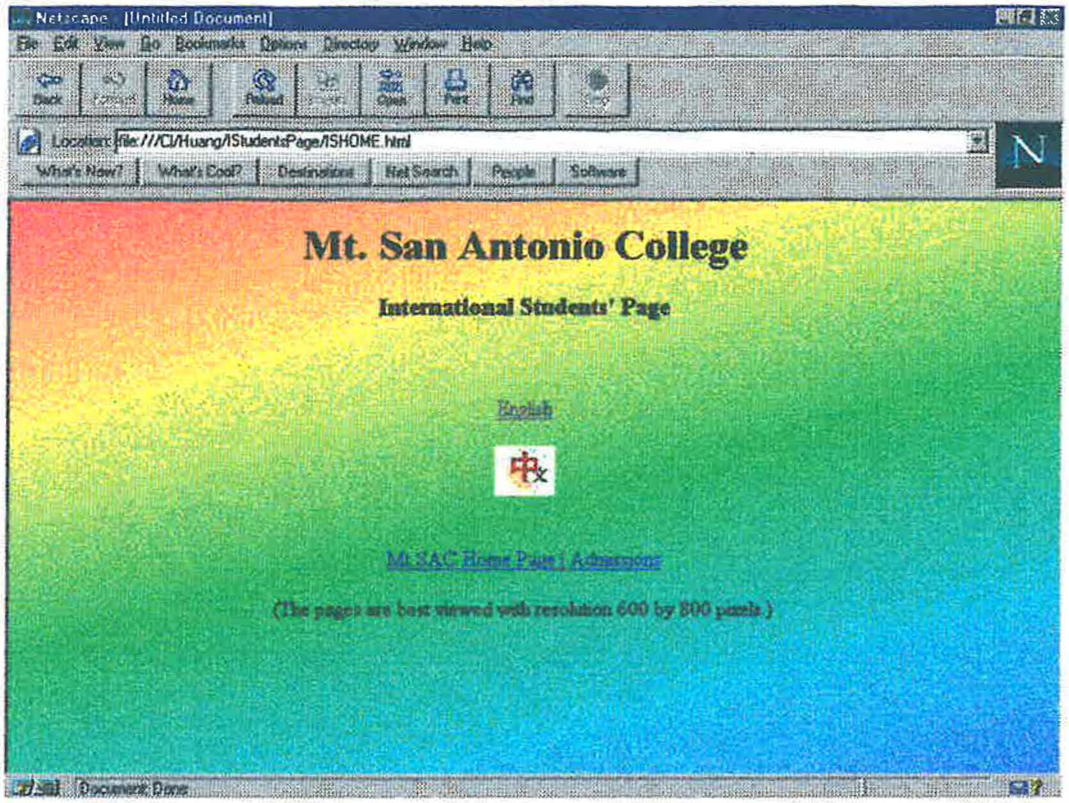
   Two links in the table of contents frame were added for users who do not have Chinese system and would like to view the web pages in Chinese. If the user has Internet Explorer browser 5.0, he or she could view Chinese pages by changing View – Encoding to Chinese. The other option the user has is to download NJStar Communicator by clicking the link and to follow the instruction from the web page.

Result

The international student web page project is stored in a 3 ½" disk. (Appendix F, p.142)

Acknowledgements

I thank Ms. Hannah Chi for her help to translate English into Chinese. Ms. Theresa Wu, who spent time entering text in Chinese Word 97, is greatly appreciated. I thank Mr. Joe Vasquez who gave me the opportunity to learn PageMill.


Netscape - [Untitled Document]
File Edit View Go Bookmarks Options Directory Window Help
Back Forward Home Reload Images Open Print Find Stop
What's New? What's Cool? Destinations Net Search People Software
N
Document: Done

Location: file:///C|/Huang/IStudentsPage/ISHOME.html

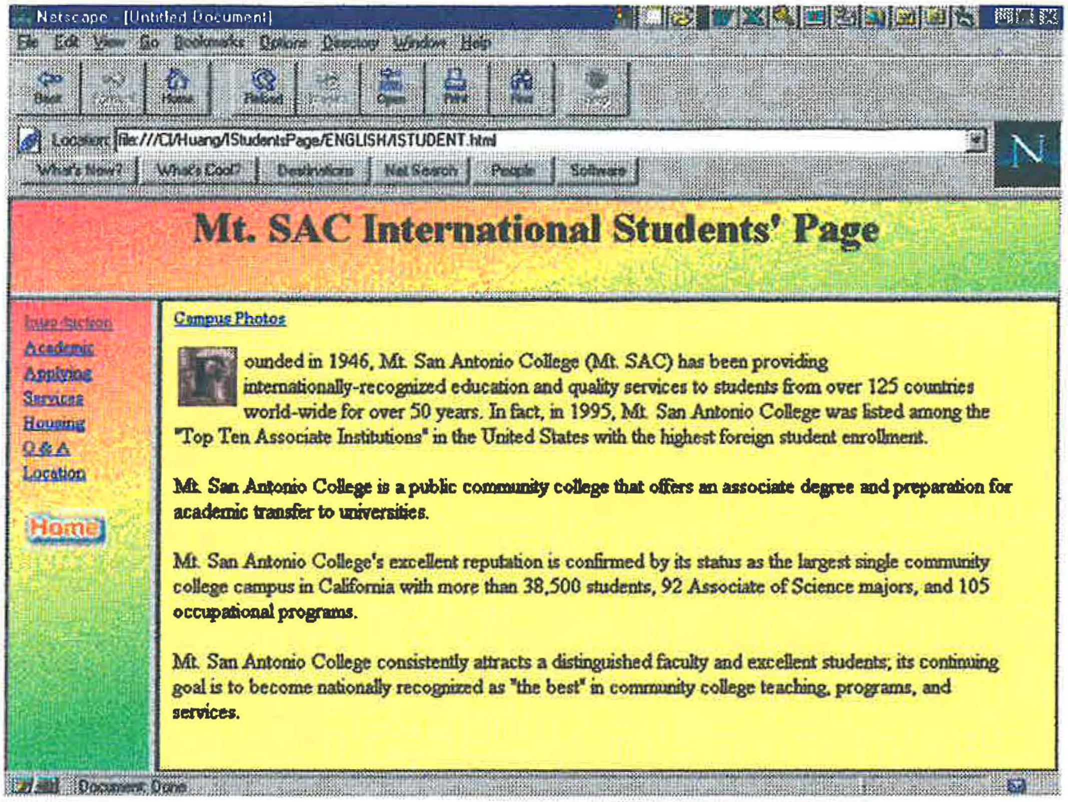
Mt. San Antonio College

International Students' Page

English

Mt.SAC Home Page | Admissions

(The pages are best viewed with resolution 600 by 800 pixels.)



Netscape - [Untitled Document]
File Edit View Go Bookmarks Options Directory Window Help
Back Forward Home Reload Images Open Print Find Stop
Location: file:///C|/Huang/IStudentsPage/ENGLISH/ISTUDENT.html
What's New? What's Cool? Destinations Net Search People Software
Introduction
Academic
Applying
Services
Housing
Q & A
Location
Home
Campus Photos
Document: Done

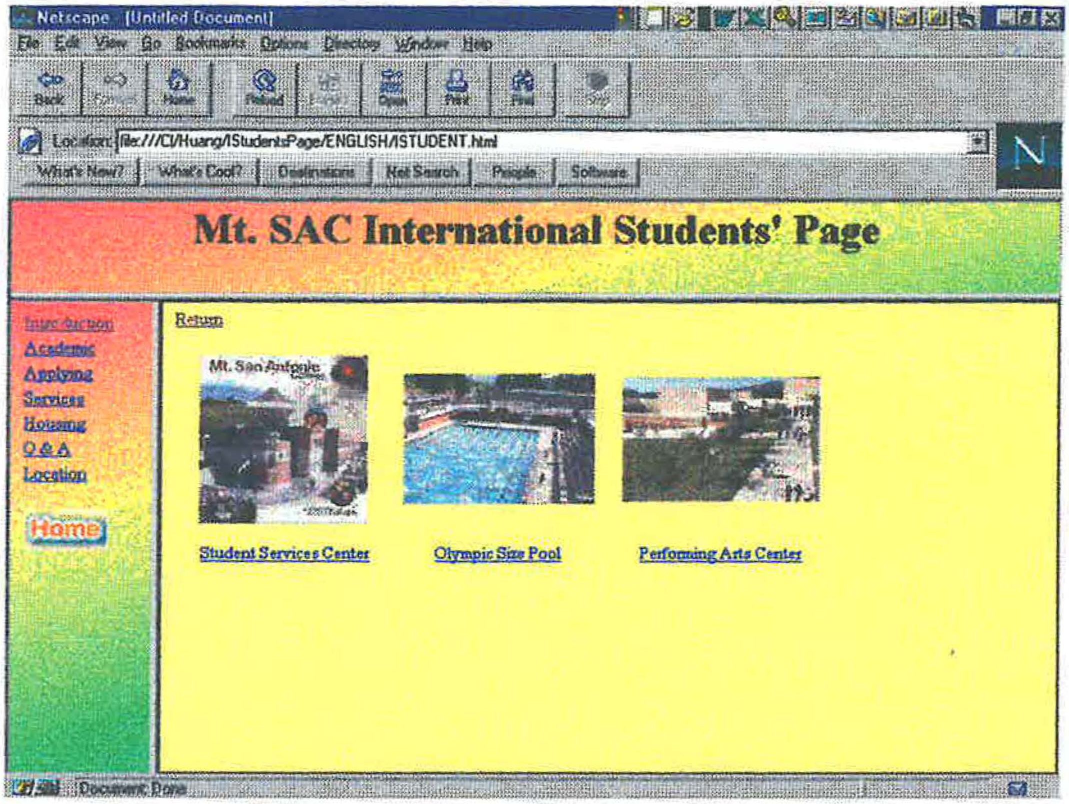
Mt. SAC International Students' Page

ounded in 1946, Mt. San Antonio College (Mt. SAC) has been providing internationally-recognized education and quality services to students from over 125 countries world-wide for over 50 years. In fact, in 1995, Mt. San Antonio College was listed among the "Top Ten Associate Institutions" in the United States with the highest foreign student enrollment.

Mt. San Antonio College is a public community college that offers an associate degree and preparation for academic transfer to universities.

Mt. San Antonio College's excellent reputation is confirmed by its status as the largest single community college campus in California with more than 38,500 students, 92 Associate of Science majors, and 105 occupational programs.

Mt. San Antonio College consistently attracts a distinguished faculty and excellent students; its continuing goal is to become nationally recognized as "the best" in community college teaching, programs, and services.


Netscape - [Untitled Document]
File Edit View Go Bookmarks Options Directory Window Help
Back Forward Home Reload Images Open Print Find Stop
What's New? What's Cool? Destinations Net Search People Software
Mt. San Antonio
Home

Location: file:///C|/Huang/IStudentsPage/ENGLISH/ISTUDENT.html

Mt. SAC International Students' Page

Introduction
Academic
Applying
Services
Housing
Q & A
Location

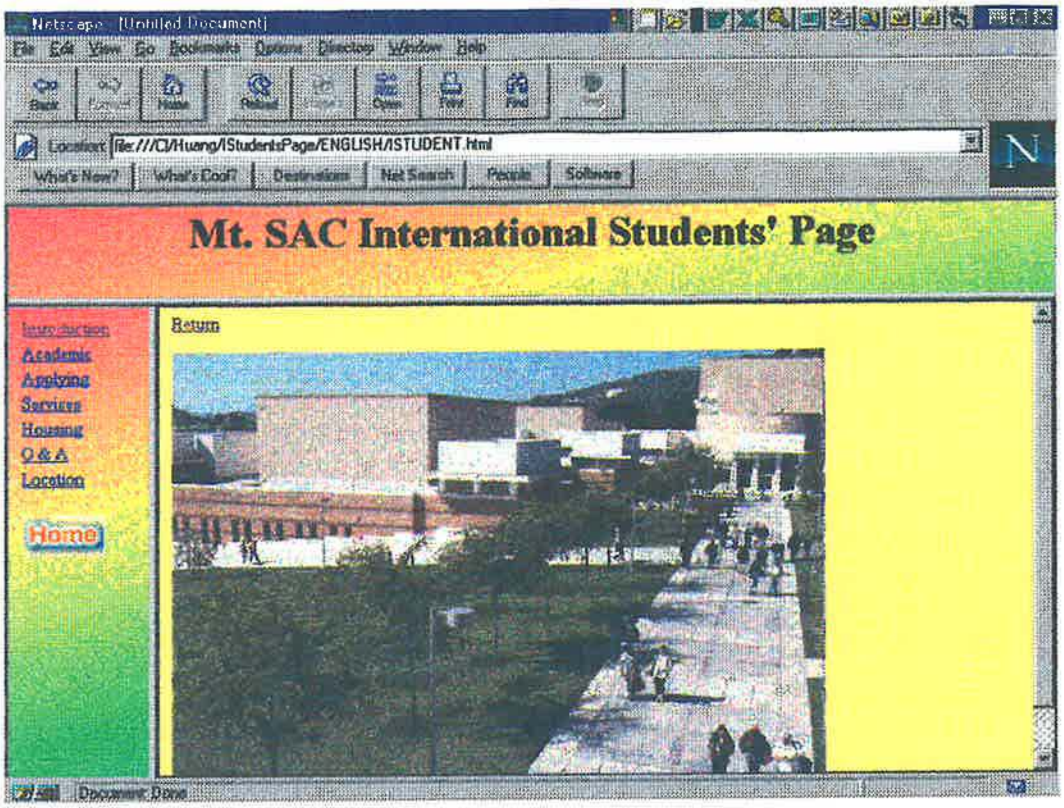
Return

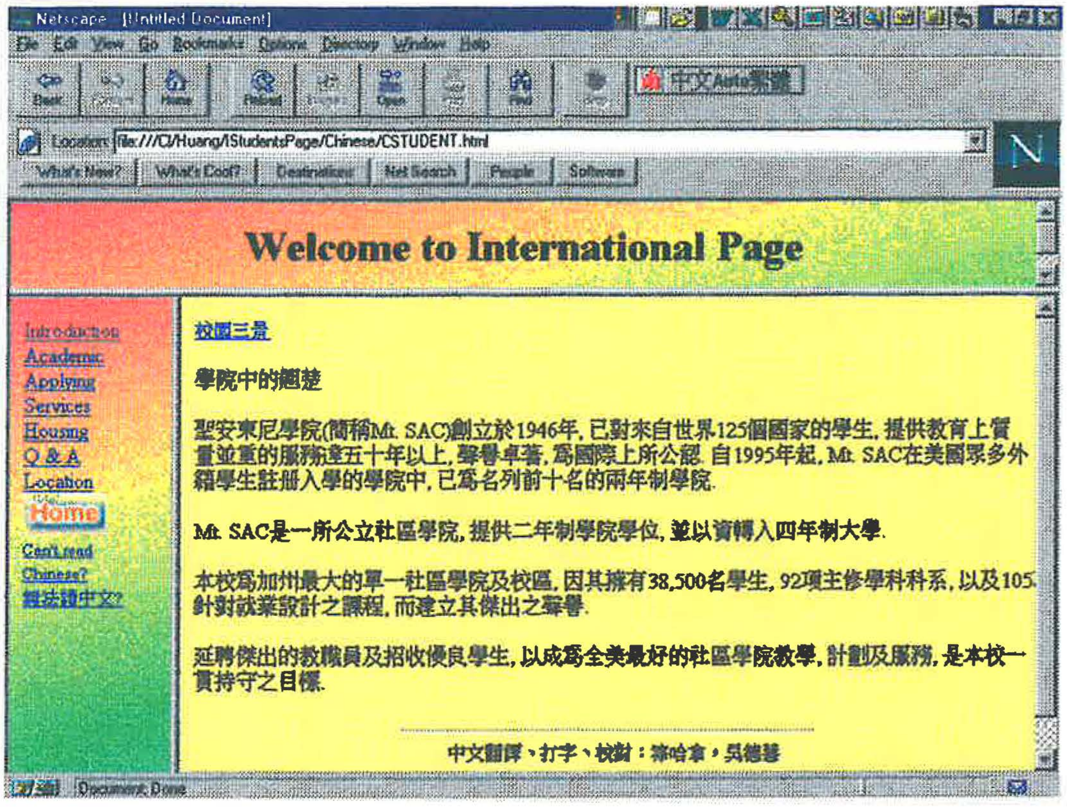
Student Services Center

Olympic Size Pool

Performing Arts Center

Document: Done


Netscape - [Untitled Document]
File Edit View Go Bookmarks Options Directory Window Help
Back
Forward
Home
Reload
Images
Open
Print
Find
Stop
Location: file:///C|/Huang/IStudentsPage/ENGLISH/ISTUDENT.html
What's New?
What's Cool?
Destinations
Net Search
People
Software
Mt. SAC International Students' Page
Introduction
Academic
Applying
Services
Housing
Q & A
Location
Home
Return
Document: Done



Welcome to International Page

校園三景

學院中的翹楚

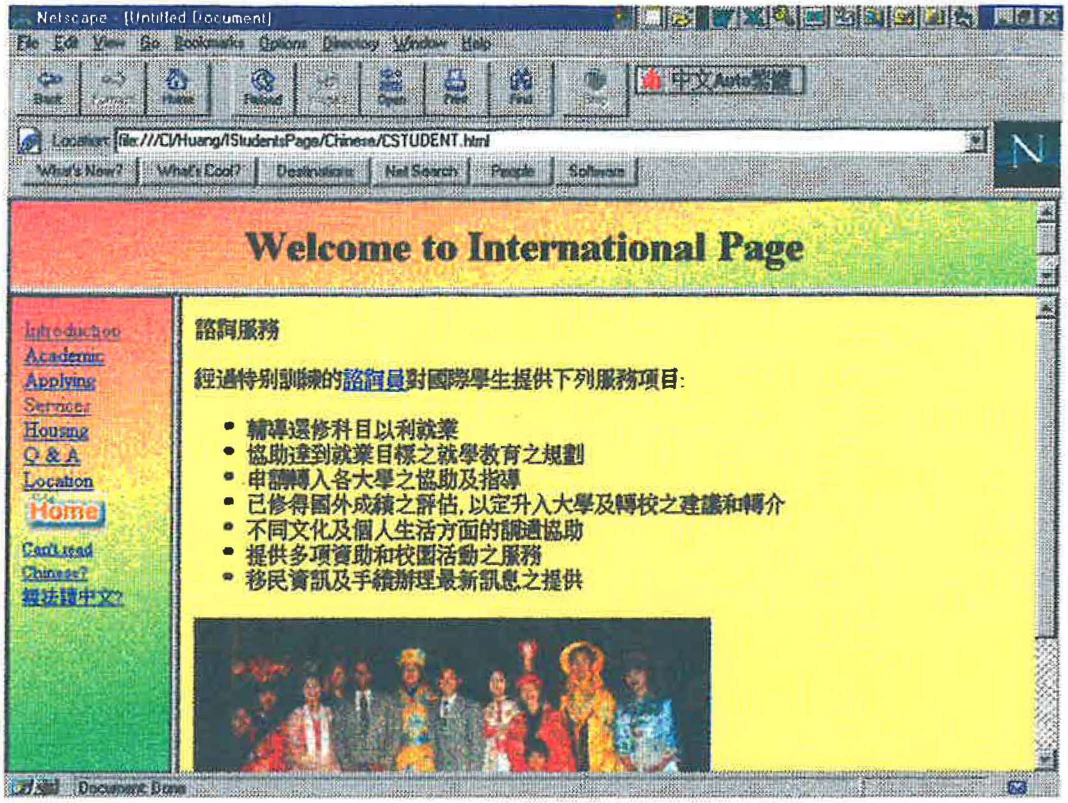
聖安東尼學院(簡稱Mt. SAC)創立於1946年, 已對來自世界125個國家的學生, 提供教育上質量並重的服務達五十年以上, 聲譽卓著, 爲國際上所公認. 自1995年起, Mt. SAC在美國眾多外籍學生註冊入學的學院中, 已爲名列前十名的兩年制學院.

Mt. SAC是一所公立社區學院, 提供二年制學院學位, 並以資轉入四年制大學.

本校爲加州最大的單一社區學院及校區, 因其擁有38,500名學生, 92項主修學科科系, 以及105針對就業設計之課程, 而建立其傑出之聲譽.

延聘傑出的教職員及招收優良學生, 以成爲全美最好的社區學院教學, 計劃及服務, 是本校一貫持守之目標.

中文翻譯、打字、校對：蔣哈拿，吳德慧


Netscape - [Untitled Document]
File Edit View Go Bookmarks Options Directory Window Help
Back Forward Home Reload Images Open Print Find Stop
中文Auto繁體
Location: file:///C|/Huang/IStudentsPage/Chinese/CSTUDENT.html
What's New? What's Cool? Destinations Net Search People Software
Document: Done

Welcome to International Page

Introduction
Academic
Applying
Services
Housing
Q & A
Location
Home
Can't read Chinese?
無法讀中文?

諮詢服務

經過特別訓練的諮詢員對國際學生提供下列服務項目:

- 輔導選修科目以利就業
- 協助達到就業目標之就學教育之規劃
- 申請轉入各大學之協助及指導
- 已修得國外成績之評估, 以定升入大學及轉校之建議和轉介
- 不同文化及個人生活方面的調適協助
- 提供多項資助和校園活動之服務
- 移民資訊及手續辦理最新訊息之提供

Summary

Through collaboration with Ms. Nancy King, Mt. SAC's International Student Counselor, we have developed the contents for Mt. SAC's international web pages and database management system. With my computer technical background and advanced training (taking the classes), I have finished designing, testing, and documenting the two projects.

The international student web pages project provides a highly visible bridge between Mt.SAC and the potential foreign students around the world. The web pages give our potential students the information they need before they make their school decisions. With increasing enrollments of Chinese students on campus, I have decided to start our first foreign language - Chinese. In the future, it would only be logical to continue adding other languages such as Japanese and Spanish.

The international student Access project will ease the counselors' work in getting the appropriate information. With this program, the counselor can save time on clerical work and serve the international students better. Instead of spending time on clerical work, with this database management program, the counselors will have more time for the international students.

While accumulating my professional skills in the computer field, the completed web pages and the database management program will provide the students in the department with practical experience and problem solving skills. Besides, the two useful projects will help our CIS department become more visible in our college.

With the increasing class demands of advanced Access and Visual Basic courses, and due to the retirement of some faculty members; my study program of sabbatical leave definitely meets the department's growing needs. My sabbatical leave has given me the opportunity to take more responsibilities to teach more advanced classes. After finishing my sabbatical leave, I am much more competent to contribute my expanded knowledge toward my students, my department, and my college.

Appendix A

CIS33 Assignments

100

Assignment 1

1. What are the major characteristics of a database application?

   A database application allows the users to store, retrieve, and update data that support critical decisions in a business.

2. Name at least three applications that could use a DBMS.

   Personnel information, warehouse inventory, and customer billing system.

3. What types of objects comprise an Access application?

   Tables, Queries, Forms, Reports, Macros, and Modules.

4. What is the purpose of the Database window?

   Database window is used to create new objects, and view or modify existing objects.

5. What is the purpose of the Startup window?

   The Startup window is used to specify a default menu or form to open when the application starts.

6. When is the Repair Database command issued?

   (1) When a database is corrupted, or
   (2) Before compacting a database.

7. What is the purpose of the Compact Database command?

   The Compact Database command should be used periodically to defragment the database.

8. What is Datasheet view? What Access objects support Datasheet view?

   The Datasheet view displays contents of a table or a query result to the user. The objects Tables, Queries, and Forms support Datasheet view.

9. What is the purpose of Design view?

   The Design view lets the user to create or modify a table structure, specify what data to be retrieved in a select query, and create or modify a form structure.

10. What information is displayed in a property sheet?

A property sheet displays information about the characteristics of a control, such as the format styles, data location, control's behaviors, and what macro or procedure to execute when an event occurs.

11. What is the difference between a standard module and a class module?

Standard module - It is listed on the Modules tab of the Database window. This type of module contains subroutines and functions that can be run at any time, as long as the application is open.

Class module - It contains the definition for a new object. Class module exists both independently and in association with forms and reports.

12. What is the difference between a module and a procedure?

Module is a collection of VBA declarations, statements, and procedures stored together as a unit. A procedure is a sequence of declarations and statements in a module that are executed as a unit.

13. What columns are available in the Macro window?

Macro Name, Condition, Action, and Comment.

14. What is the purpose of the Conditions column in the Macro window?

Conditions column contains a statement that can be evaluated as True or False. When the condition is True, the associated action is executed.

15. What is the purpose of the Action column in the Macro window?

It lists the action to perform when the Condition column evaluates to True.

16. What triggers a macro's execution in Access?

Macros are executed by clicking a menu command or toolbar button. They are triggered mostly in response to events.

100

Assignment 2

17. What is the purpose of the Expression Builder?

    It helps to build different types of Access expressions, including reference to controls on forms, references to reports, and function calls.

18. Using Access Help, describe the purpose of the SetValue action.

    The action is used to set the value of a Microsoft Access field, control, or property on a form, a form datasheet, or a report.

19. Using Access Help, describe the purpose of the Beep action.

    You can use this action to sound a beep through the computer's speaker.

20. Using Access Help, describe the purpose of the SetFocus method.

    The SetFocus method moves the focus to the specified form, the specified control on the active form, or the specified field on the active datasheet.

21. Using Access Help, describe the purpose of the Switchboard Manager.

    You can use the Switchboard Manager to create a Switchboard that helps you to navigate around the database. This switchboard has buttons that you can click to open forms and reports, open other switchboards, quit Microsoft Access, or customize the switchboard.

22. To learn more about the MU-Dsci application, open forms and search for information, using the menus, combo boxes, buttons, and toolbars. In particular, see if you can answer the following questions:

    - What is the status of Martin Lopez's application?

      Martin Lopez's application is in pending status.

    - Can Timothy Bennifield drop a course?

      No.

- What Leveling courses does Cynthia Morney need?

  MGMT 602
  FINC 629
  MGMT 631
  ACCT 640
  MKTG 641
  MGMT 680

- Who is the primary contact for Bana 320?

  Dr. William Korst.

- Finally, using the Prospects form, add yourself as a prospect.

  Done. DO YOU THINK THEY WILL ACCEPT YOU?

23. The Northwind Traders database is applied with Microsoft Access. It is usually located in a samples directory inside Access or the Microsoft office group. Open Northwind Traders and answer the following questions.

- What is the unit price of the Chang Beverage?

  $19

- What is the address of the supplier, Grandma Kelly's Homestead?

  707 Oxford Rd., Ann Arbor, MI 48104

100%

Assignment 3

Page 129 Question 1-20

1. Why is Access described as object-based but not object-oriented?

   Access is described as object-based because it supports object properties, object methods, collections, and objects that contain other object. However, Access is not described as object-oriented because it does not support class inheritance.

2. What is collection?

   An object that contains a group of related objects.

3. What are DAO objects? What are Microsoft Access objects? Identify at least three objects in each category.

   DAO objects - objects related to data storage and retrieval. TableDefs, QueryDefs, and Recordsets are DAO objects.

   Microsoft Access objects - objects related to the user interface and modules. Forms, reports, and modules are the examples.

4. How can an object own a collection?

   For example, a report object owns a Controls collection that contains all the controls in the report.

5. What objects are members of the Forms collection?

   All of the currently open forms objects in an Access database.

6. What objects are members of the TableDefs collection?

   All of the TableDef objects.

7. What property indicates whether a control can be seen?

   Visible

8. What properties can be used to prevent a user from entering a value in a text box?

   Enabled

9. What property indicates the source of a bound form?

   Record Source

10. What property indicates the source of data for a control?

    Control Source

11. What properties are used to remove the navigation buttons and record selector buttons form a form?

    Navigation Buttons and Record Selectors

12. What does it mean when a form's Modal property is set to Yes?

    The user must respond to the current form in some way, usually by clicking a command button before switching to other form. The user cannot respond to any other form until the modal form is closed.

13. Suppose you have a report that contains a repeating field. What property can be used to suppress the printing of the data in a control when the prior record printed contained the same value?

    Hide Duplicates

14. What is the purpose of the Dirty property?

    The property is used to check whether the current record has been modified since it was last saved.

15. What is the purpose of the NewRecord property?

    You can use it to determine whether the current record is a new record.

16. What object and property can be used to refer to the form that is currently active?

    You can use the Screen object together with the ActiveForm property to identify to the form that has the focus.

17. What is the DBEngine?

    It is an object contains and controls all other objects in the hierarchy of DAO objects. The DBEngine represents the jet database engine that stores and retrieves data.

18. What is the difference between a Recordset object and a TableDef object?

Recordset - a copy of the **data** in memory retrieved from a table or query. It provides access to the data stored in a database file on a disk.

TableDef - an object represents the stored **structure** (definition) of a table.

19. What is the difference between table-type, dynaset-type and snapshot-type recordset objects?

Table-type Recordset represents data contained in a single nonlinked table. The data in table-type recordset can be modified.

Dynaset-type Recordset represents the data retrieved from queries or linked tables. The data in dynaset-type recordset can be modified by adding, deleting, or updating records.

Snapshot-type A snapshot-type recordset object is a static set of records that you can use to examine data in an underlying table or tables. The data in snapshot-type recordset cannot be modified.

20. What is the purpose of the Properties collection? What objects own a properties collection?

The properties collection contains property objects. The user can use the property objects to store information that describe an object such as caption, enabled, visible, etc.

Most objects own a properties collection.

100%

Assignment 4

Chapter 4 Page 159

1. What is an event?

   An event is an action associated with an object (control, form, or report) through an event property.

2. What is an event property?

   An event property is a property of an object that associates an event that is recognized by the object with a procedure or macro.

3. Describe the relationship between an event, and event property, and a macro.

   If an event occurs, the macro identified by the event property is executed.

4. Describe the relationship between an event, and event property, and a function.

   A function is used when more than one event triggers the same VBA code. If the events occur, the function identified by the event property is executed.

5. Describe the relationship between an event, and event property, and an event procedure.

   If an event occurs, the event procedure identified by the event property is executed.

6. What is the difference between a Click event and a MouseDown event?

   Click - is triggered when a user presses and releases the left mouse button while the cursor is over a control or form.

   MouseDown - occurs when the user presses a mouse button.

7. Identify at least one use for a Click event.

   It is often used with command buttons to open forms or reports.

8. What are the differences between a control's Change, BeforeUpdate, and AfterUpdate events?

Change - occurs for each keystroke in a text box or combo box.

BeforeUpdate - occurs after a user changes a value but before the value is written to the current record buffer or before the record is moved to the database.

AfterUpdate - occurs after the change has been written to the current record buffer or the record has been written to the database.

9. Describe the relationship between a form's BeforeInsert, AfterInsert, BeforeUpdate, and AfterUpdate events.

The BeforeInsert event occurs as soon as the first character is typed in a new record but before the record is actually created. Next, the BeforeUpdate event occurs before the value is written to the current record buffer or before the record is moved to the database. Then, AfterUpdate event occurs after the change has been written to the current record buffer or the record has been written to the database. Finally, the AfterInsert event occurs after a new record is added to a database.

The sequence for the events:

BeforeInsert → BeforeUpdate → AfterUpdate → AfterInsert

10. What events occur when the cursor on a form is moved to another record?

AfterInsert

11. Does the Delete event occur before or after a record is actually deleted?

Before

12. Describe the relationship between the Delete, BeforeDelConfirm, and AfterDelConfirm events.

After a record is deleted, it's stored in a temporary buffer. The BeforeDelConfirm event occurs after the Delete event, but before the Delete Confirm dialog box is displayed. Canceling the BeforeDelConfirm event restores the record or records from the buffer and prevents the Delete Confirm dialog box from being displayed. The AfterDelConfirm event occurs after a record or records are actually deleted or after a deletion or deletions are canceled.

If you cancel the Delete event, the BeforeDelConfirm and AfterDelConfirm events don't occur and the Delete Confirm dialog box isn't displayed.

The sequence for the events:

Delete → BeforeDelConfirm → AfterDelConfirm

13. What does it mean to "get the focus"? Describe some ways that a control or form can get the focus.

When an object can receive user input through mouse or keyboard actions, the object gets the focus. A control can get the focus if the user clicks the control or presses the Tab key into this control.

14. What is the difference between the GotFocus and Enter events?

The Enter event is triggered just before a control on the form receives the focus. Once the control received the focus, the GotFocus event occurs.

15. What is the difference between the Load event and the Activate event for a form?

The Load event occurs when a form is opened and its data is displayed. Then, the Activate event occurs when a form or report receives the focus.

16. Identify a common use for the Activate and Deactivate events.

Activate - to display toolbars or forms that must be displayed with the form that is active.

Deactivate - to close toolbars or other forms that are only displayed when the deactivated form is displayed.

17. What sequence of events occur when a form is opened?

Form: Open → Form: Load → Form: Activate → Form: Current
First Control: → Enter → First Control: GotFocus

18. What sequence of events occur when a form is closed?

Control:Exit → Control: LostFocus → Form: Unload → Form: Deactivate
Form: Close

100%

Assignment 5
Chapter 6 Page 246

1. What is the difference between an independent class module and a form or report class module?

   Independent class module is very similar to a form or report class module except they are not tied to a form or report.

2. What is the difference between an independent class module and a standard module?

   An independent class module contains procedures that can be executed as long as the application is open. It is a custom-made object whose properties can be referenced and manipulated.

   A standard module is a module in which you can place Sub and Function procedures that you want to make available to other procedures throughout your database.

3. What are the roles of the Declarations section?

   This section is used to declare variables, constants and certain rules that govern how Access analyzes and executes your code.

4. How is DoCmd related to a macro action?

   It corresponds to an action listed in a macro's Action column.

5. What VBA statement is equivalent to the Condition column in a macro?

   If ...Then...Else statement.

6. What is the difference between a function and a Sub procedure?

   The function procedure returns value while a Sub procedure does not.

7. What is the difference between formal arguments and actual arguments?

   Formal arguments – arguments specified in which a function or sub procedure is defined.

   Actual arguments – arguments specified in which a function or sub procedure is called.

8. What is the difference between passing arguments “by reference” and “by value”?

By reference – the subprocedure will be sent the address of the argument’s value rather than a copy of the value.

By value – the subprocedure will be sent a copy of the argument’s value rather than the address of the value.

9. What is an event procedure?

An event procedure triggers when its associated action occurs.

10. When should a Variant data type be used?

If the data type is not certain when you declare the variable or if the variable need to be different data type during run time.

11. Write the statements that make the following declarations:

a. A function named Taxes with one argument, Value. Taxes return a Currency value. Value is Currency that is passed “by value.”

Private Function Taxes (ByVal Value as Currency) as Currency

b. A variable named FullName is a string variable.

Dim pstrFullName as String

c. A variable name ZipCode that can be used by any module in an application. ZipCode is a string variable.

Public gstrZipCode as string

d. A variable named CurrentBalance that is Currency value. It should maintain an accurate current value even when the procedure in which it is declared is not being executed.

Static pcurCurrentBalance as Currency

12. What is an array?

A variable contains a specified number of elements that have a common name and data type.

13. Write a statement that defines a private array that holds up to 10 integers. The values correspond to the ProjectID's of projects under construction during the current month.

    Dim pintProjectID(1 to 10) as Integer

14. What is a symbolic constant?

    A symbolic constant is a constant that is defined by the programmer.

15. What is an intrinsic constant?

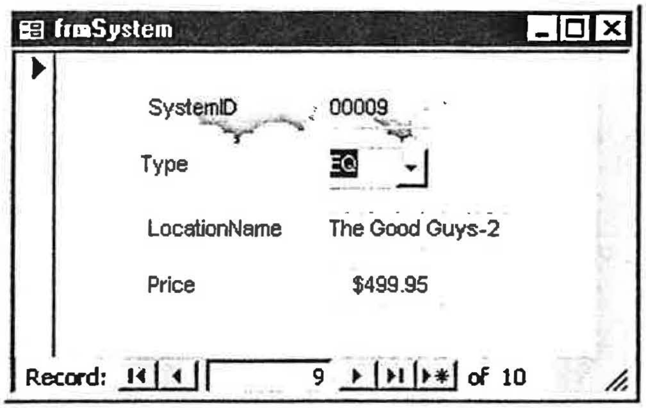
    An intrinsic constant is a predefined constant provided by Access.

Appendix B

CIS33 Projects

Systeml	00009
Type	EQ
LocationNam	The Good Guys-2
Price	$499.95

Printing from Access, but the combo box doesn't show.



SystemID 00009

Type EQ

LocationName The Good Guys-2

Price $499.95

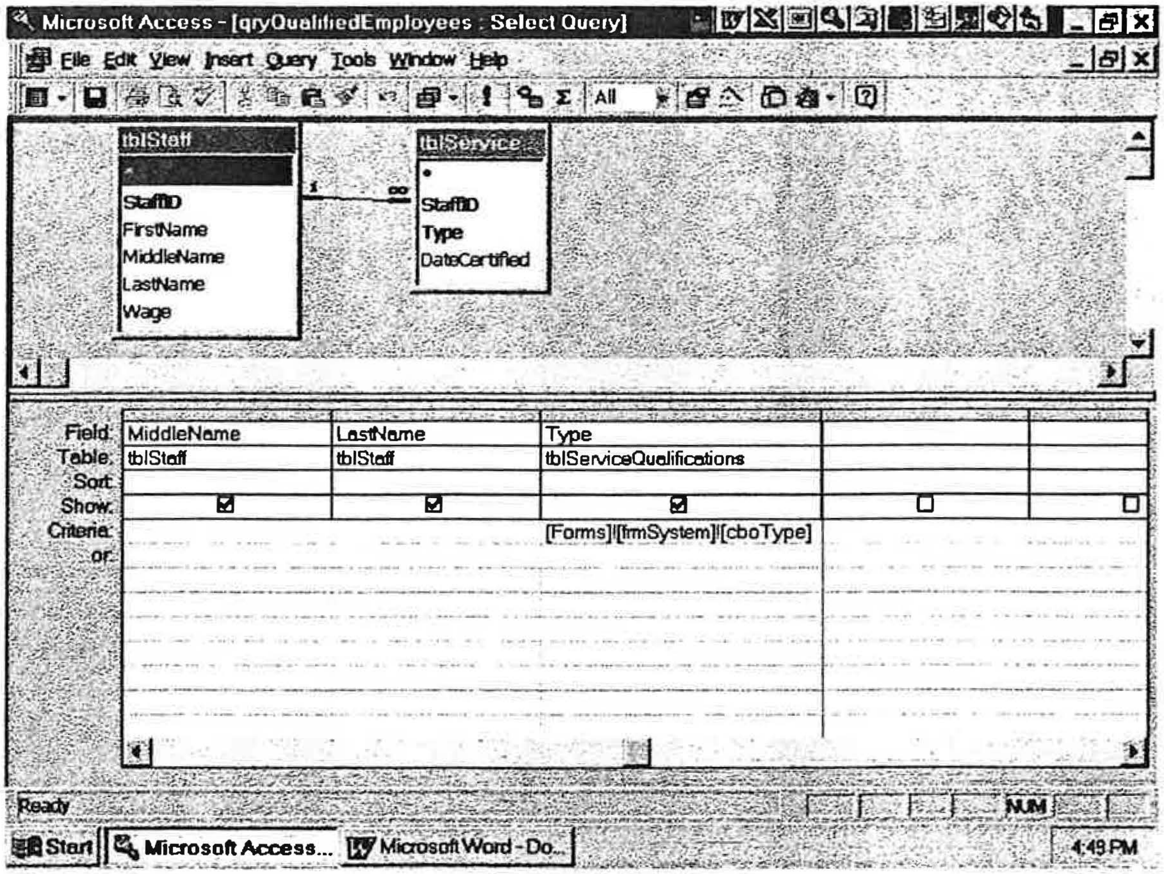
Record: 9 of 10

screen print

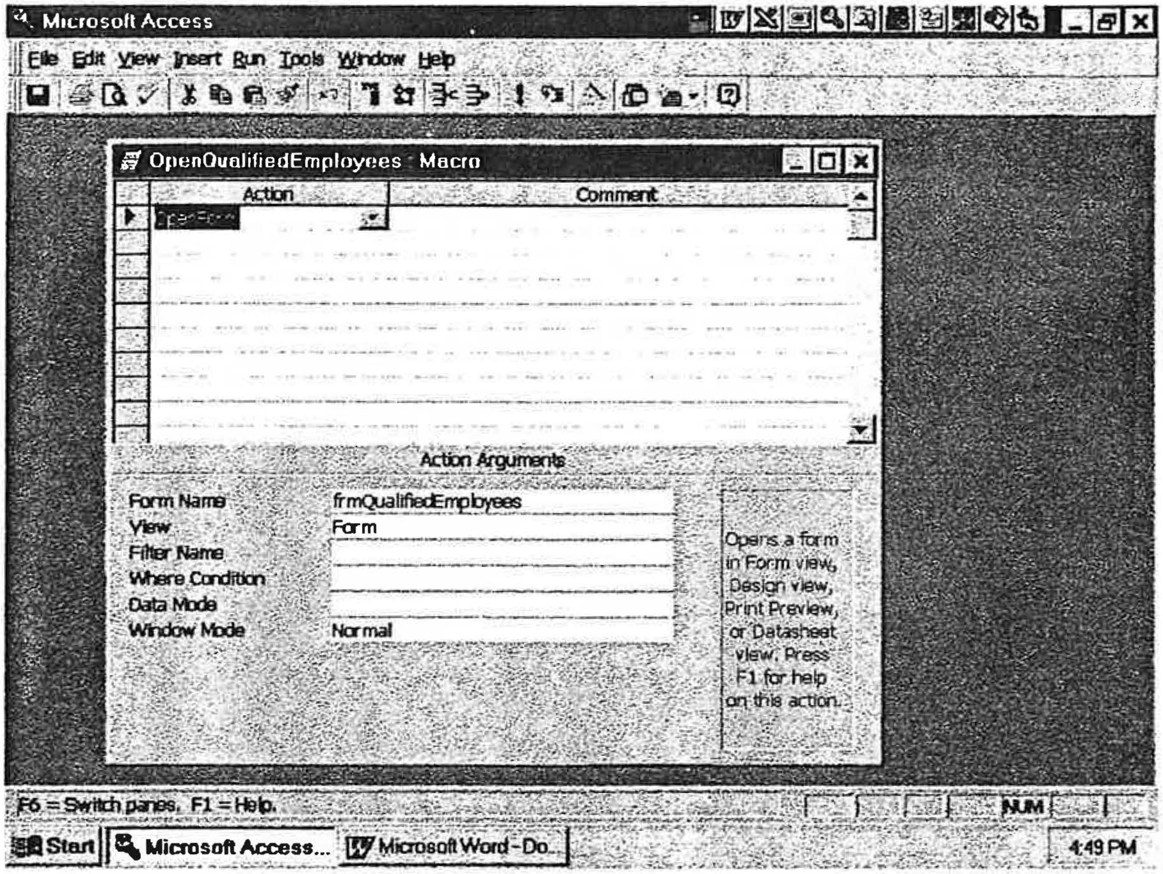
Sales By Location

City	*Location*	*Syste*	*Type*	*Price*	*Date*	*Warranty*	*Address*	*State*	*Zip*
Azusa	BestBuy-2	00007	NonCD	250.99	2/20/99	1	825 W. 8th Street	CA	91702
City of Industry	BestBuy-1	00001	NonCD	169.95	1/30/99	1	17545 Gale Ave.	CA	91748
	Circuit City-2	00010	Sat-3	608.35	2/26/99	1	17543 E. Gale Ave.	CA	91748
	Hollytron	00006	IR-2	599.75	2/18/99	1	106 Puente Hills Mall	CA	91748
	Sears-1	00002	EQ	450.99	2/ 2/99	5	100 Puente Hills Mall	CA	91748
Montebello	Sears-2	00008	Sat-2	487.36	2/25/99	5	1401 N. Montebello Blvd.	CA	90640
Monterey Park	The Good Guys-2	00009	EQ	499.95	2/25/99	1	2345 s. atlantic Blvd.	CA	91754
West Covina	Circuit City-1	00003	IR-1	612.59	2/ 5/99	1	339 N. Azusa Ave.	CA	91791
	Macy's	00004	Sat-1	999.99	2/10/99	1	1200 W. Covina Parkway	CA	91790
	The Good Guys-1	00005	CD	238.95	2/15/99	5	1000 W. Covina Parkway	CA	91790

100



Field:	MiddleName	LastName	Type		
Table:	tblStaff	tblStaff	tblServiceQualifications		
Sort:					
Show:	☑	☑	☑	□	□
Criteria:			[Forms]![frmSystem]![cboType]		
or:					


Microsoft Access
File Edit View Insert Run Tools Window Help
F6 = Switch panes. F1 = Help.
NUM
Start
Microsoft Access...
Microsoft Word - Do...
4:49 PM

OpenQualifiedEmployees : Macro

Action	Comment
OpenForm	

Action Arguments

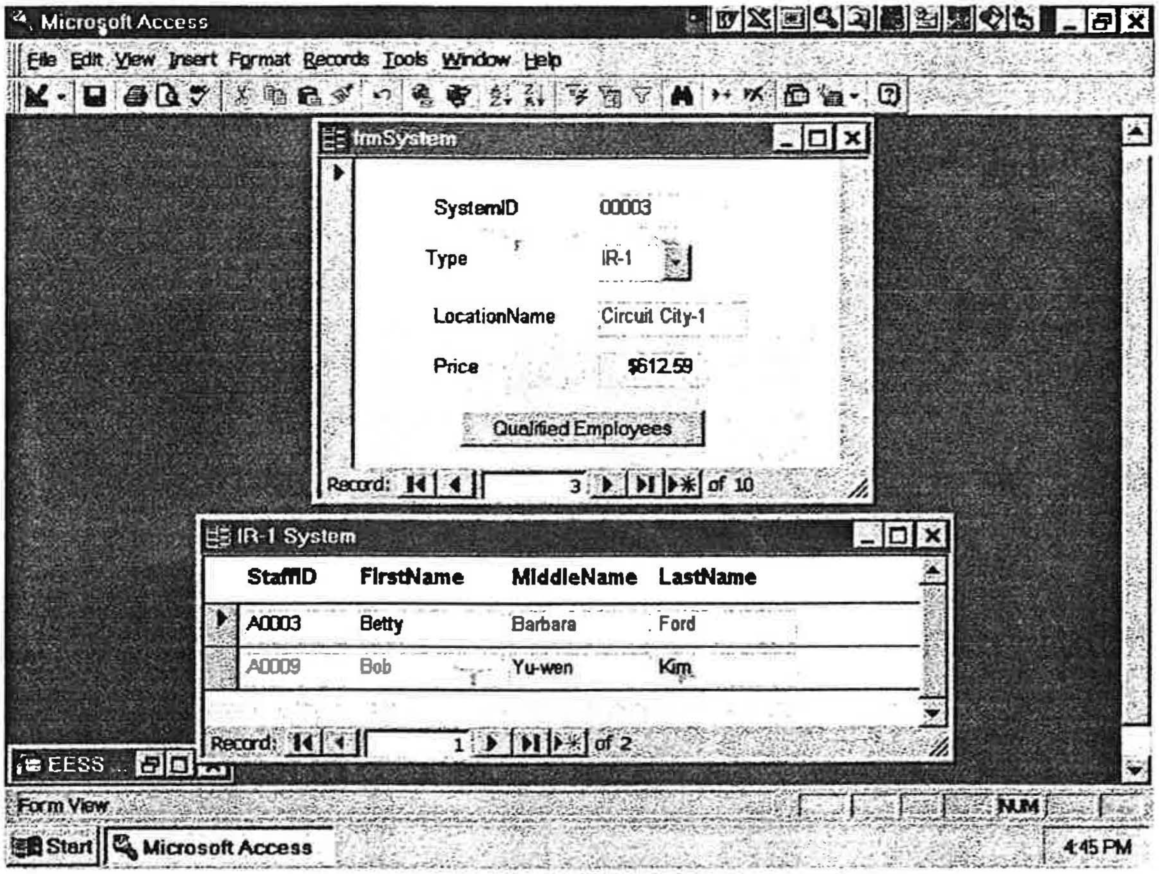
Form Name	frmQualifiedEmployees
View	Form
Filter Name	
Where Condition	
Data Mode	
Window Mode	Normal

Opens a form in Form view, Design view, Print Preview, or Datasheet view. Press F1 for help on this action.

VBA program

```
Option Compare Database
Option Explicit

Private Sub Form_Open(Cancel As Integer)
    Forms![frmSystem]![cboType].SetFocus
    Me.Caption = Forms![frmSystem]![cboType].Text & " System"
End Sub
```



SystemID 00003

Type IR-1

LocationName Circuit City-1

Price $612.59

StaffID	FirstName	MiddleName	LastName
A0003	Betty	Barbara	Ford
A0009	Bob	Yu-wen	Kim

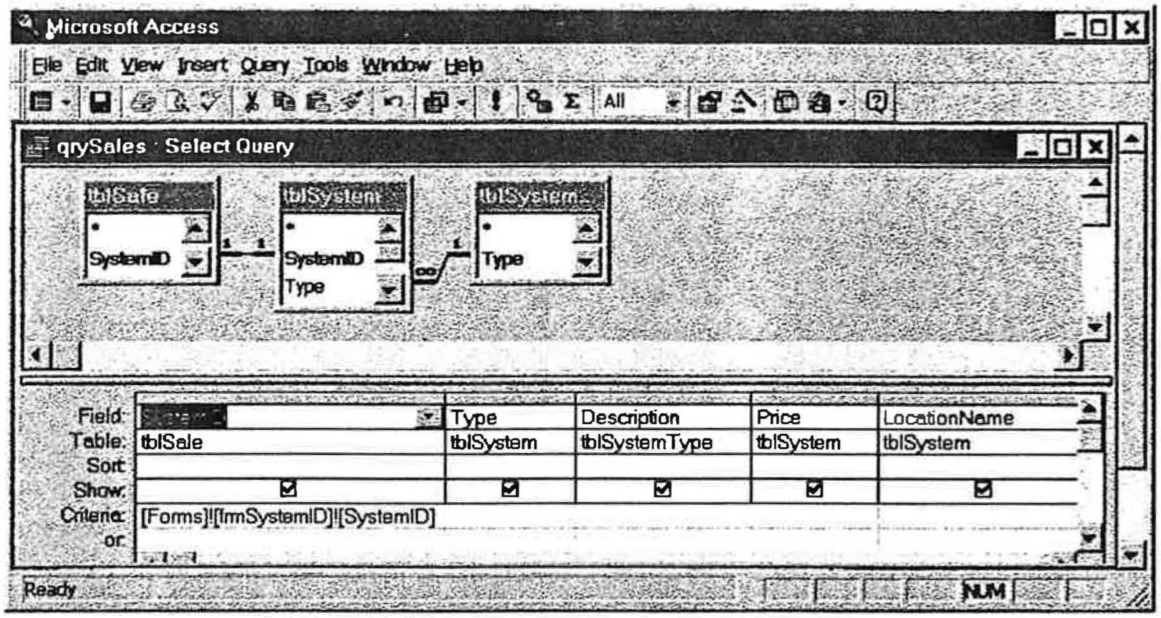
The [illegible] ed Employee button [illegible]

[illegible] f[illegible]

Chapter 9

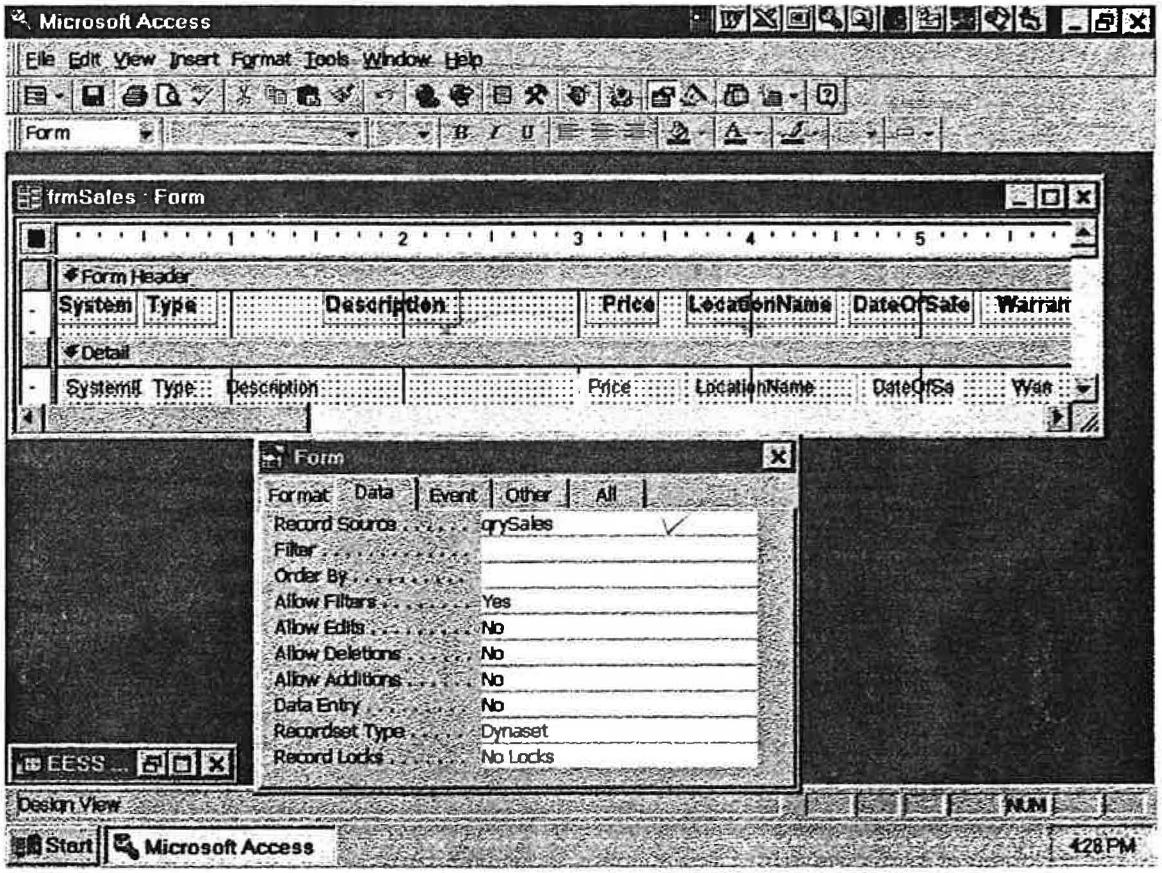
100%

Create a query qrySales



Field:	SystemID	Type	Description	Price	LocationName
Table:	tblSale	tblSystem	tblSystemType	tblSystem	tblSystem
Sort:					
Show:	☑	☑	☑	☑	☑
Criteria:	[Forms]![frmSystemID]![SystemID]				
or:					

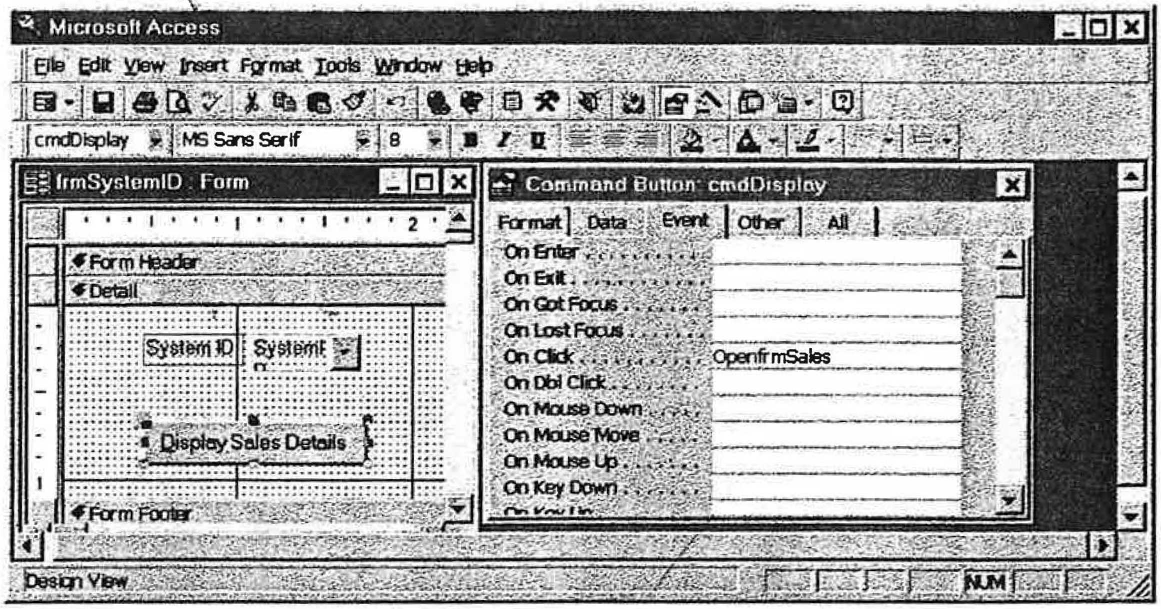
Create a form frmSales based on qrySales


Microsoft Access
File Edit View Insert Format Tools Window Help
Form
frmSales : Form
Form Header
System Type Description Price LocationName DateOfSale Warran
Detail
SystemI Type Description Price LocationName DateOfSa Warr
Form
Format Data Event Other All
EESS ...
Design View
NUM
Start
Microsoft Access
4:28 PM

Record Source	qrySales
Filter	
Order By	
Allow Filters	Yes
Allow Edits	No
Allow Deletions	No
Allow Additions	No
Data Entry	No
Recordset Type	Dynaset
Record Locks	No Locks

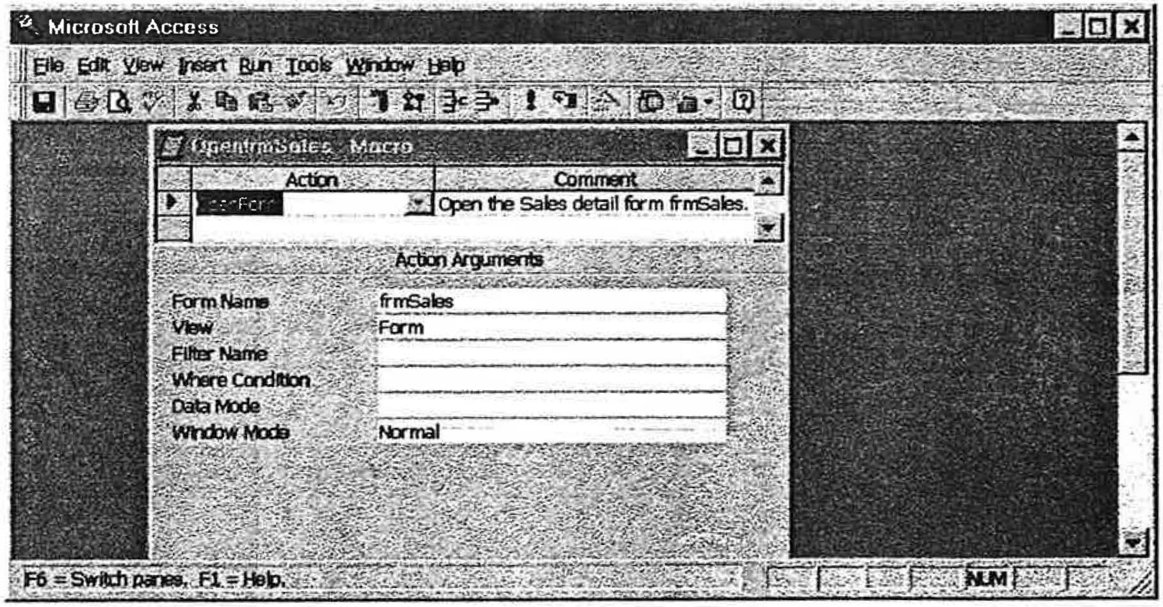
Create form SystemID

OnClick event for the command button



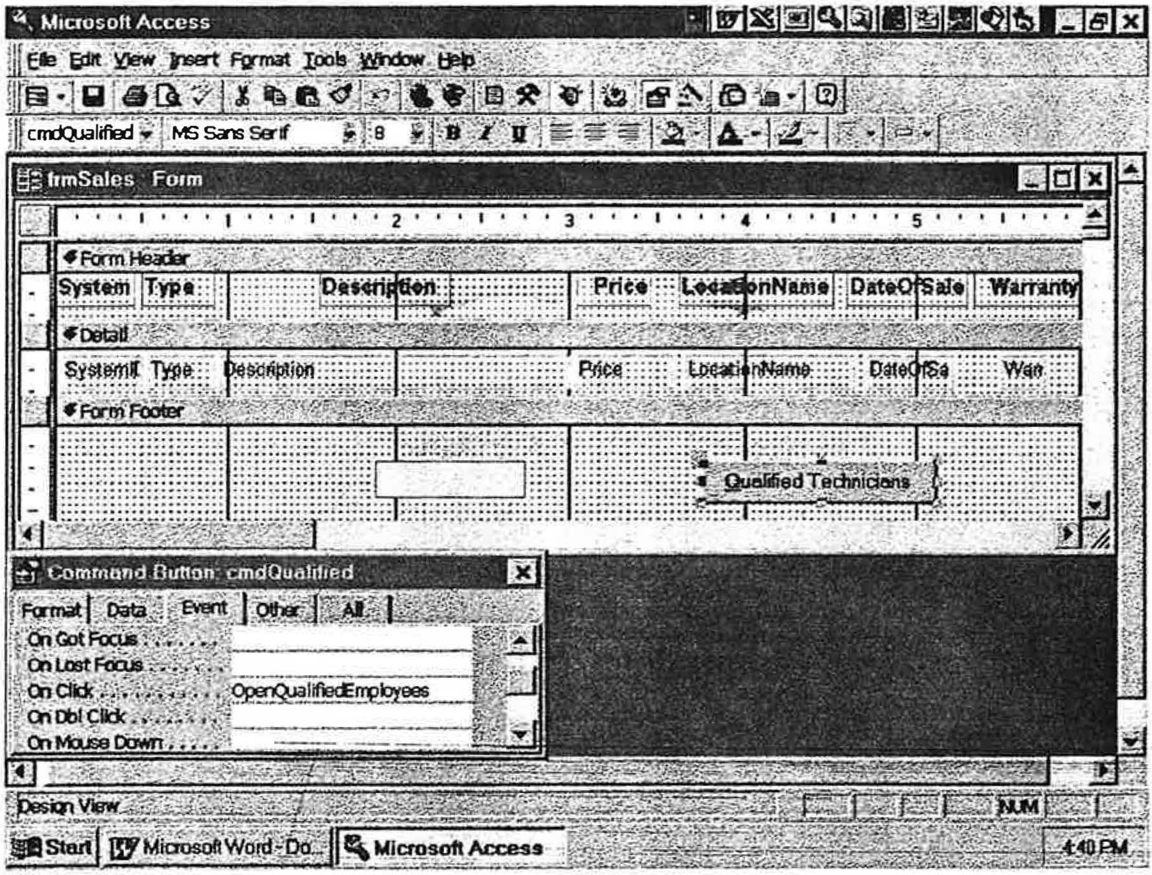
On Enter	
On Exit	
On Got Focus	
On Lost Focus	
On Click	OpenfrmSales
On Dbl Click	
On Mouse Down	
On Mouse Move	
On Mouse Up	
On Key Down	

Openform Sales Macro


Microsoft Access
File Edit View Insert Run Tools Window Help
OpenfrmSales : Macro
Action Arguments
F6 = Switch panes. F1 = Help.
NUM

Action	Comment
OpenForm	Open the Sales detail form frmSales.

Form Name	frmSales
View	Form
Filter Name	
Where Condition	
Data Mode	
Window Mode	Normal



Command Button: cmdQualified

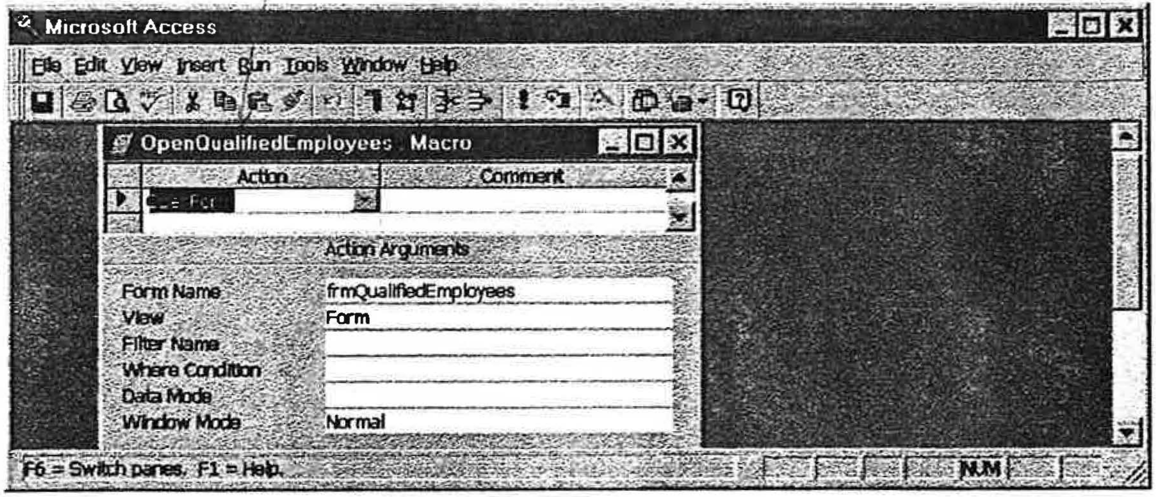
Format | Data | Event | Other | All

On Got Focus	
On Lost Focus	
On Click	OpenQualifiedEmployees
On Dbl Click	
On Mouse Down	

frmSales

← On Click event for the command button

OpenQualifiedEmployees macro

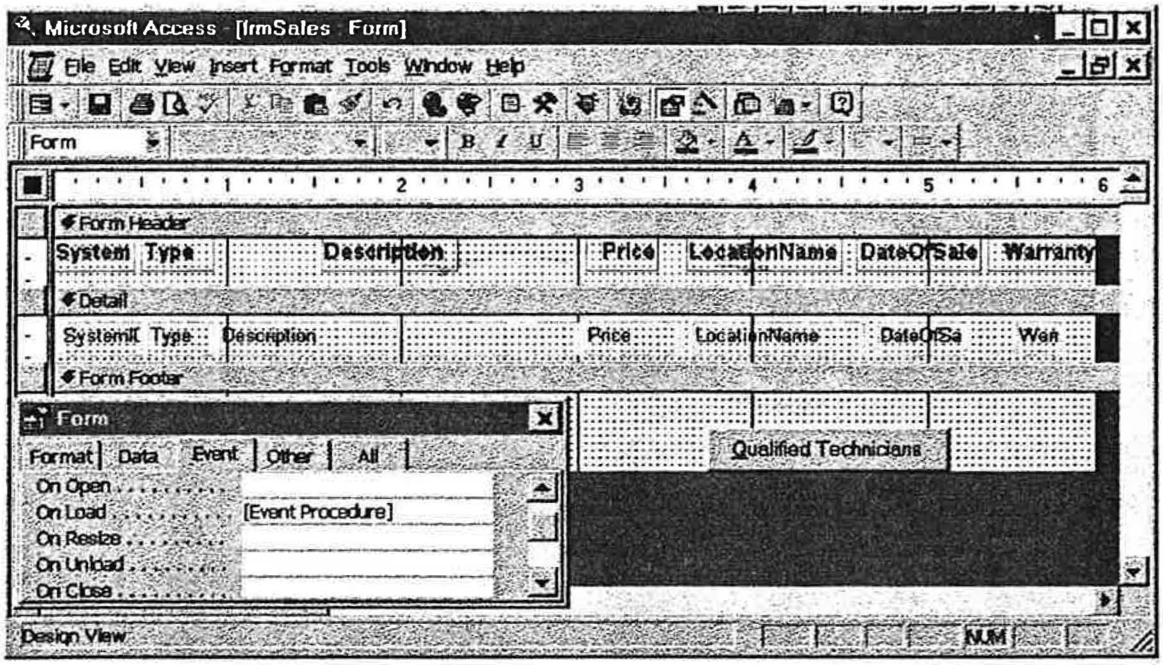

Microsoft Access
File Edit View Insert Run Tools Window Help
F6 = Switch panes. F1 = Help.
NUM

OpenQualifiedEmployees : Macro

Action	Comment
OpenForm	

Action Arguments

Form Name	frmQualifiedEmployees
View	Form
Filter Name	
Where Condition	
Data Mode	
Window Mode	Normal

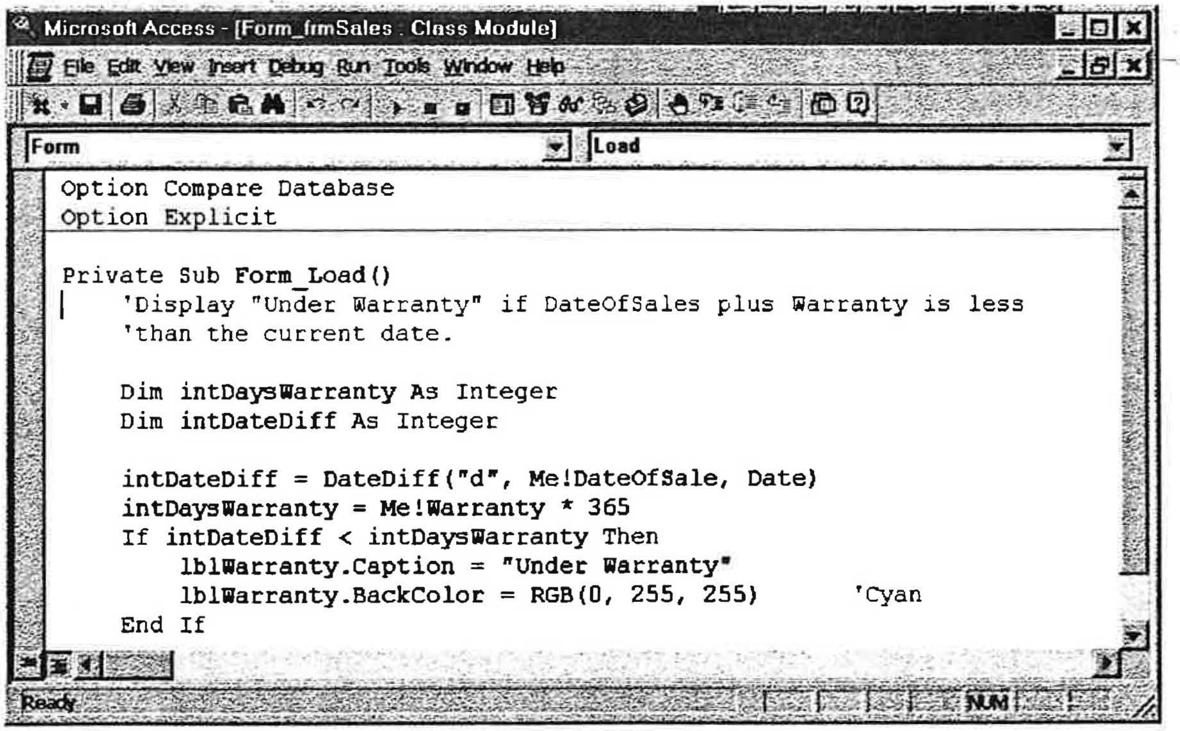


On Open
On Load [Event Procedure]
On Resize
On Unload
On Close

frm Sales

←

On Load
Event trigger
the event
procedure


Microsoft Access - [Form_frmSales : Class Module]
File Edit View Insert Debug Run Tools Window Help
Form
Load
Ready
NUM

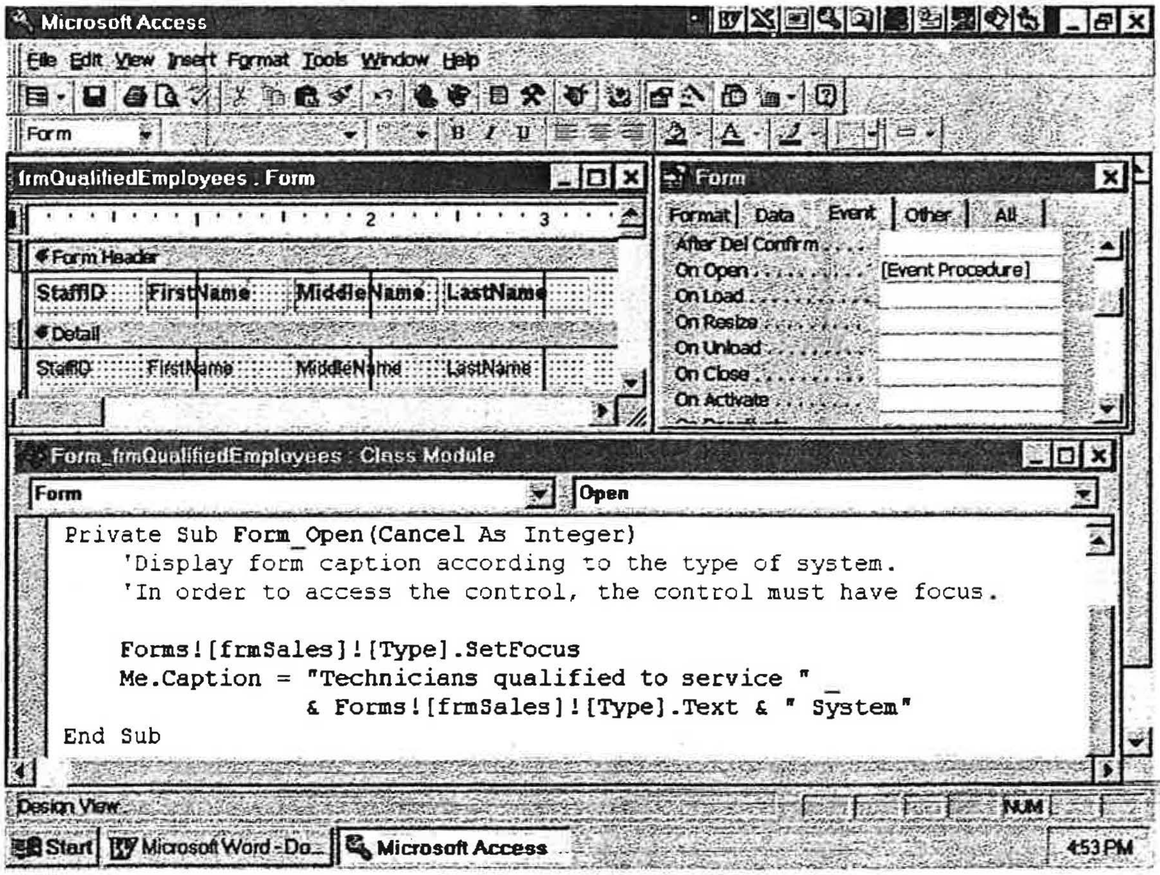
```
Option Compare Database
Option Explicit

Private Sub Form_Load()
    'Display "Under Warranty" if DateOfSales plus Warranty is less
    'than the current date.

    Dim intDaysWarranty As Integer
    Dim intDateDiff As Integer

    intDateDiff = DateDiff("d", Me!DateOfSale, Date)
    intDaysWarranty = Me!Warranty * 365
    If intDateDiff < intDaysWarranty Then
        lblWarranty.Caption = "Under Warranty"
        lblWarranty.BackColor = RGB(0, 255, 255)      'Cyan
    End If
```

frm QualifiedEmployees


Microsoft Access
File Edit View Insert Format Tools Window Help
Form
B I U
frmQualifiedEmployees : Form
Form Header
StaffID FirstName MiddleName LastName
Detail
StaffID FirstName MiddleName LastName
Form
Format Data Event Other All
Form_frmQualifiedEmployees : Class Module
Design View
NUM
Start
Microsoft Word - Do...
Microsoft Access
4:53 PM

After Del Confirm
On Open [Event Procedure]
On Load
On Resize
On Unload
On Close
On Activate

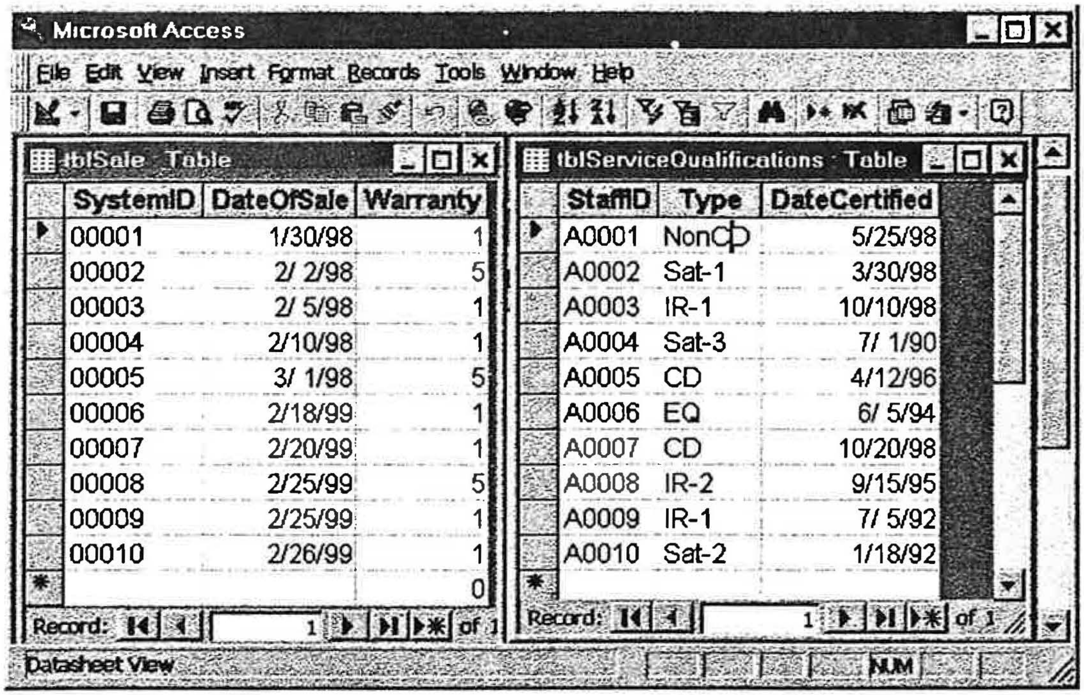
Form — Open

```
Private Sub Form_Open(Cancel As Integer)
    'Display form caption according to the type of system.
    'In order to access the control, the control must have focus.

    Forms![frmSales]![Type].SetFocus
    Me.Caption = "Technicians qualified to service " _
                 & Forms![frmSales]![Type].Text & " System"
End Sub
```

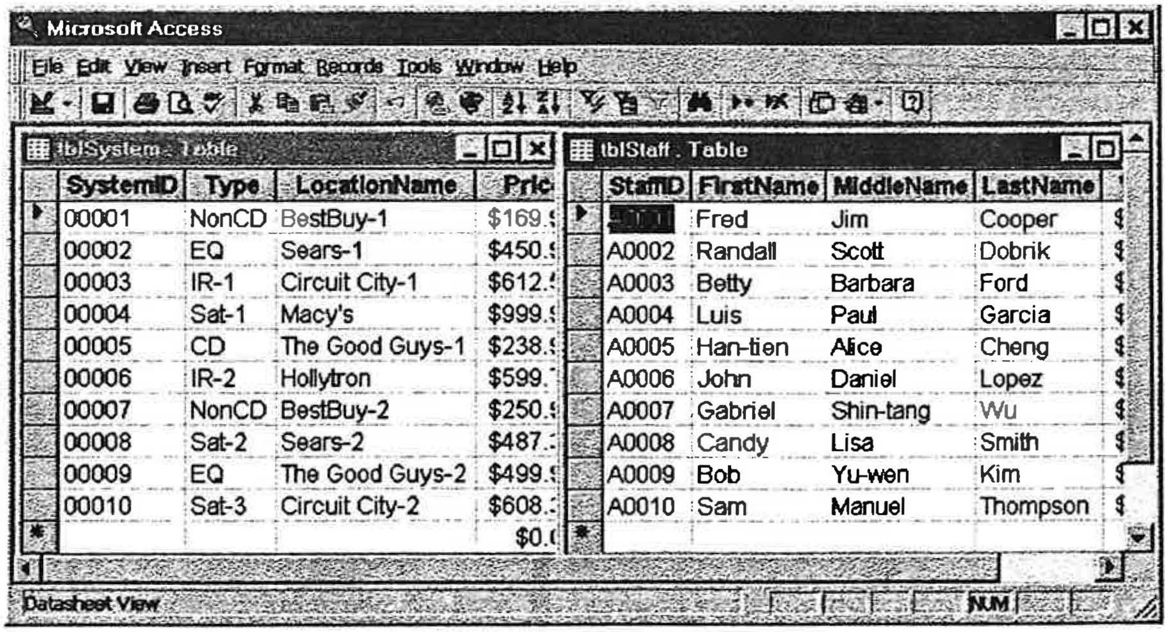
On Open ev
← triggers
the event to
open
frmQualifiedEmp

Data for sale, ServiceQualifications,
System, staff tables


Microsoft Access
File Edit View Insert Format Records Tools Window Help
tblSale : Table
tblServiceQualifications : Table
Record: 1 of 1
Datasheet View
NUM

SystemID	DateOfSale	Warranty
00001	1/30/98	1
00002	2/ 2/98	5
00003	2/ 5/98	1
00004	2/10/98	1
00005	3/ 1/98	5
00006	2/18/99	1
00007	2/20/99	1
00008	2/25/99	5
00009	2/25/99	1
00010	2/26/99	1
*		0

StaffID	Type	DateCertified
A0001	NonCD	5/25/98
A0002	Sat-1	3/30/98
A0003	IR-1	10/10/98
A0004	Sat-3	7/ 1/90
A0005	CD	4/12/96
A0006	EQ	6/ 5/94
A0007	CD	10/20/98
A0008	IR-2	9/15/95
A0009	IR-1	7/ 5/92
A0010	Sat-2	1/18/92
*		

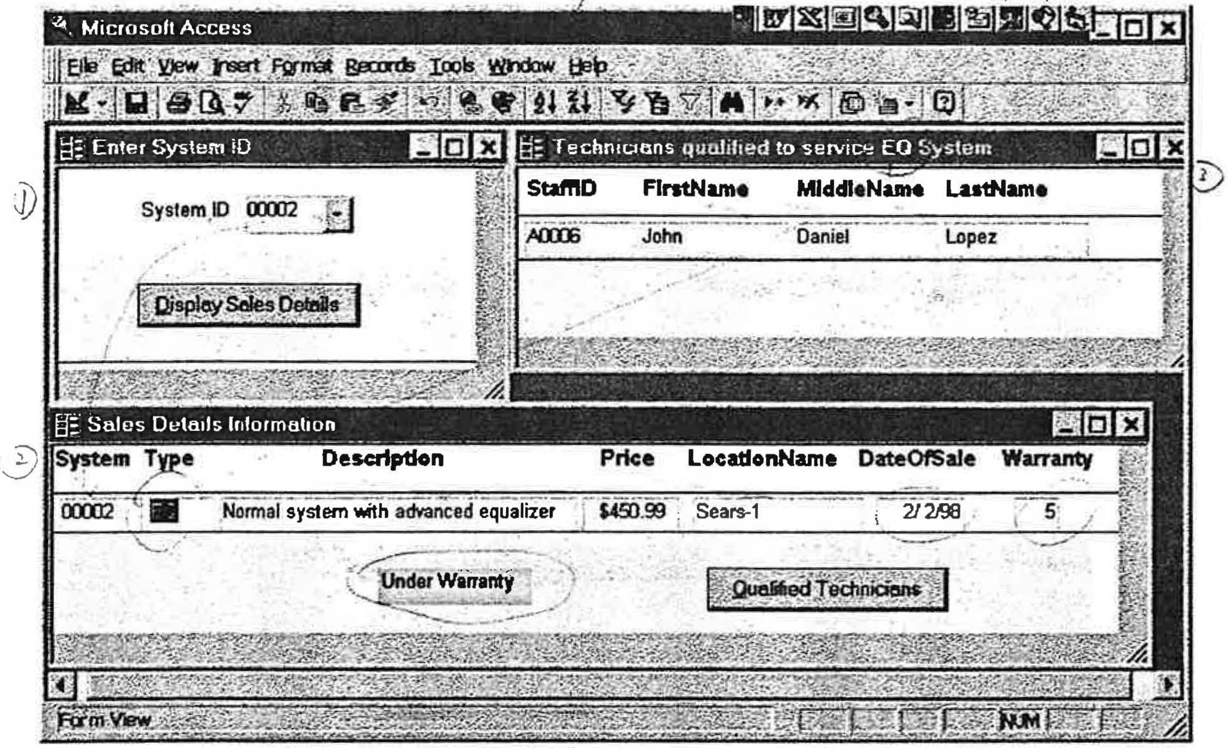


SystemID	Type	LocationName	Pric
00001	NonCD	BestBuy-1	$169.
00002	EQ	Sears-1	$450.
00003	IR-1	Circuit City-1	$612.
00004	Sat-1	Macy's	$999.
00005	CD	The Good Guys-1	$238.
00006	IR-2	Hollytron	$599.
00007	NonCD	BestBuy-2	$250.
00008	Sat-2	Sears-2	$487.
00009	EQ	The Good Guys-2	$499.
00010	Sat-3	Circuit City-2	$608.
*			$0.0

StaffID	FirstName	MiddleName	LastName	
A0001	Fred	Jim	Cooper	$
A0002	Randall	Scott	Dobrik	$
A0003	Betty	Barbara	Ford	$
A0004	Luis	Paul	Garcia	$
A0005	Han-tien	Alice	Cheng	$
A0006	John	Daniel	Lopez	$
A0007	Gabriel	Shin-tang	Wu	$
A0008	Candy	Lisa	Smith	$
A0009	Bob	Yu-wen	Kim	$
A0010	Sam	Manuel	Thompson	$
*				

Execution Samples:

When system ID 00002 is entered, detail info for this sale displayed, (under warranty shows) then technicians qualified to service EQ displayed



Enter System ID

System ID 00002

Display Sales Details

Technicians qualified to service EQ System

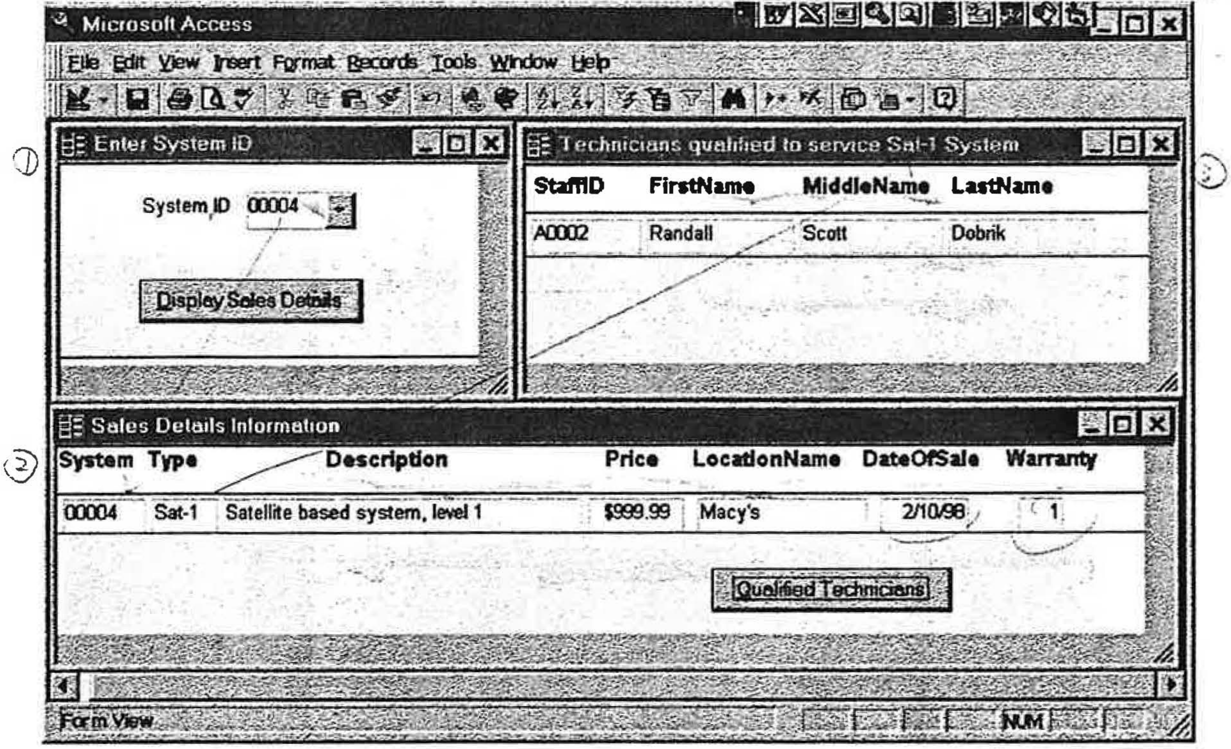
StaffID	FirstName	MiddleName	LastName
A0006	John	Daniel	Lopez

Sales Details Information

System	Type	Description	Price	LocationName	DateOfSale	Warranty
00002	■	Normal system with advanced equalizer	$450.99	Sears-1	2/2/98	5

Under Warranty

Qualified Technicians



Enter System ID

System ID 00004

Display Sales Details

Technicians qualified to service Sat-1 System

StaffID	FirstName	MiddleName	LastName
A0002	Randall	Scott	Dobrik

Sales Details Information

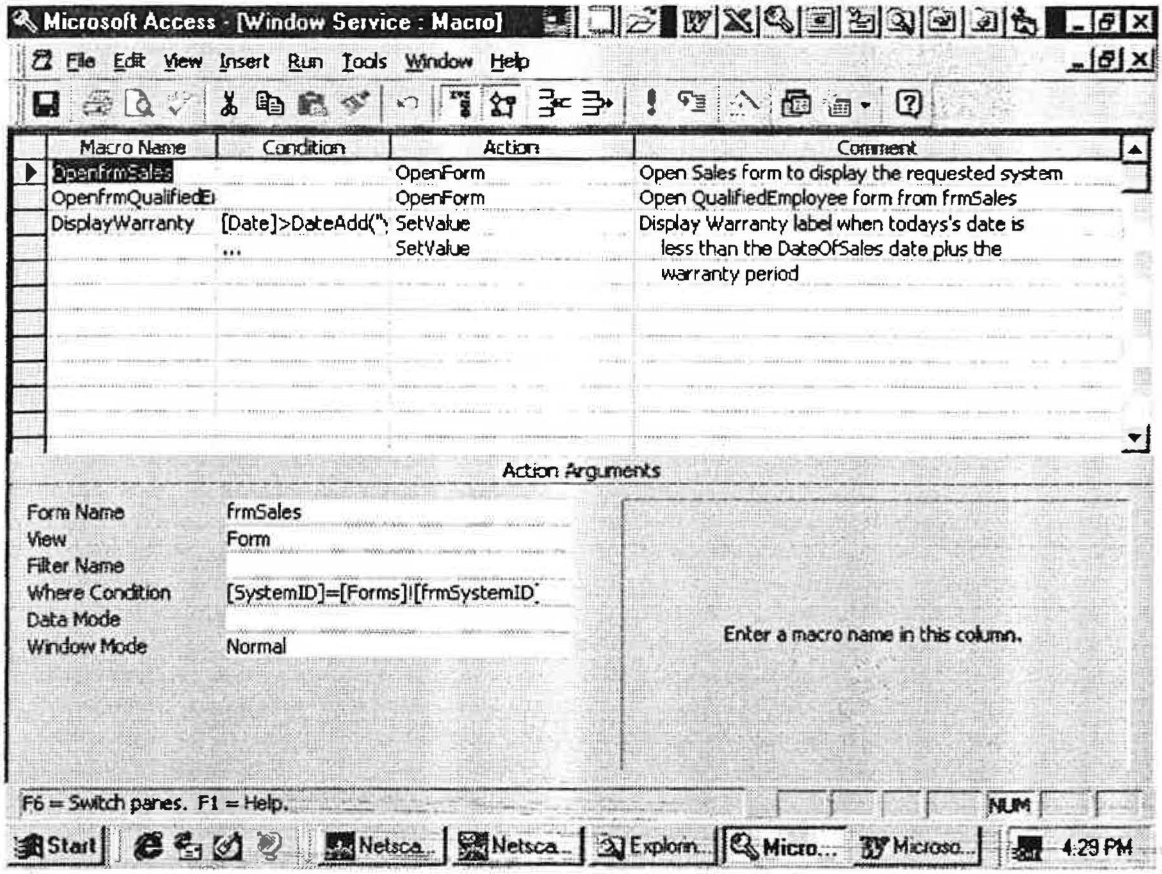
System	Type	Description	Price	LocationName	DateOfSale	Warranty
00004	Sat-1	Satellite based system, level 1	$999.99	Macy's	2/10/98	1

Qualified Technicians

When system ID 00004 is entered.

(please refer to the steps of execution & results) chapter 5

100%.


Microsoft Access - [Window Service : Macro]
File Edit View Insert Run Tools Window Help

Macro Name	Condition	Action	Comment
OpenfrmSales		OpenForm	Open Sales form to display the requested system
OpenfrmQualifiedE		OpenForm	Open QualifiedEmployee form from frmSales
DisplayWarranty	[Date]>DateAdd("	SetValue	Display Warranty label when todays's date is less than the DateOfSales date plus the warranty period
	...	SetValue	

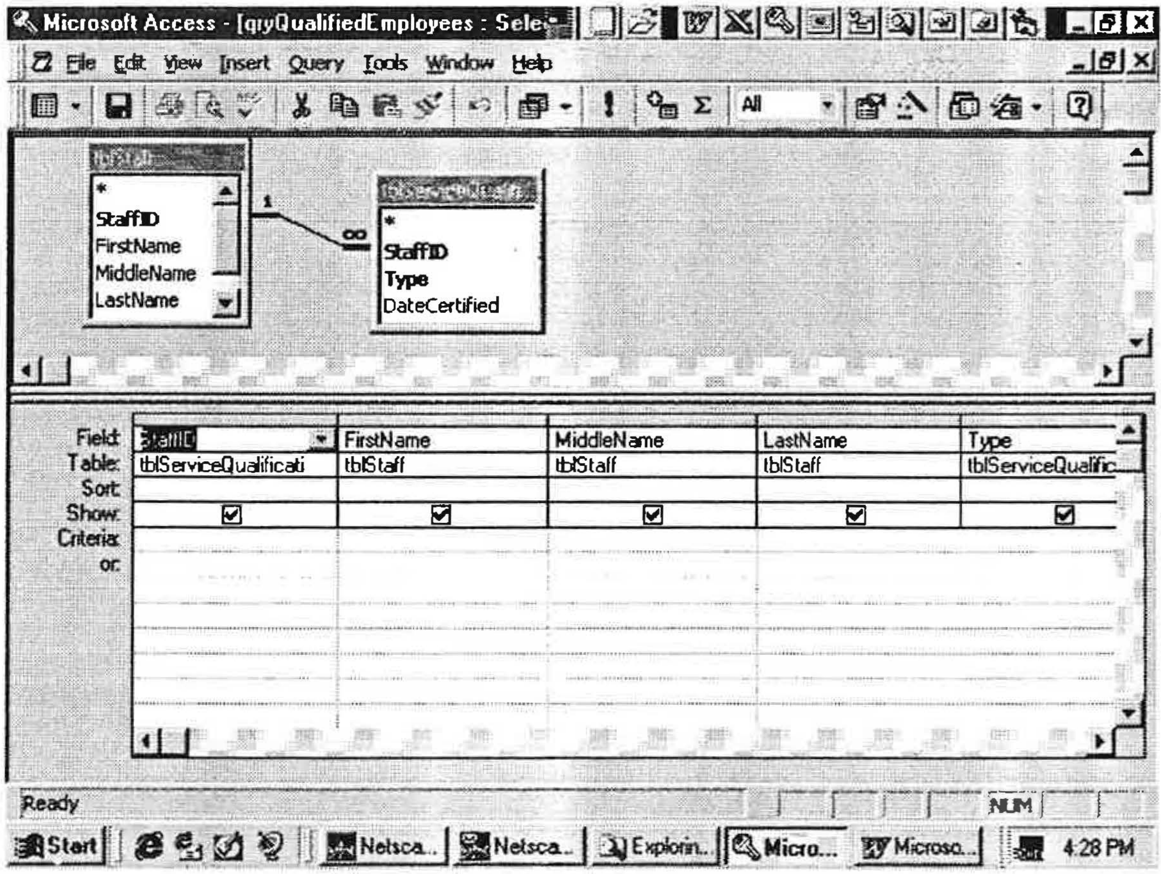
Action Arguments

Form Name	frmSales
View	Form
Filter Name	
Where Condition	[SystemID]=[Forms]![frmSystemID
Data Mode	
Window Mode	Normal

Enter a macro name in this column.

F6 = Switch panes. F1 = Help.

step 1 & 2

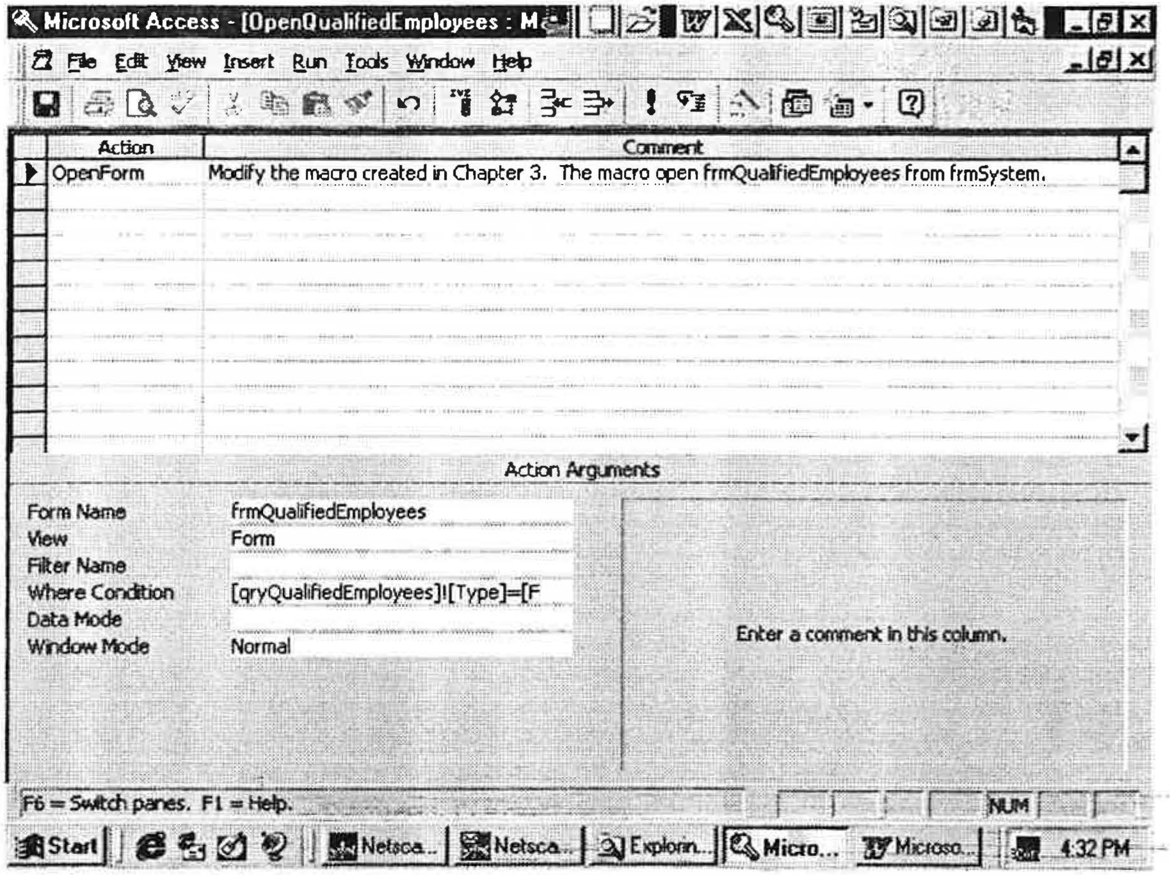

Microsoft Access - [qryQualifiedEmployees : Sele
File Edit View Insert Query Tools Window Help
All
*
StaffID
FirstName
MiddleName
LastName
1
∞
*
StaffID
Type
DateCertified

Field:	StaffID	FirstName	MiddleName	LastName	Type
Table:	tblServiceQualificati	tblStaff	tblStaff	tblStaff	tblServiceQualific
Sort:					
Show:	☑	☑	☑	☑	☑
Criteria:					
or:					

Ready

step 3

← no criteria


Microsoft Access - [OpenQualifiedEmployees : M
File Edit View Insert Run Tools Window Help
Start
Netsca...
Netsca...
Explorin...
Micro...
Microso...
4:32 PM

Action	Comment
OpenForm	Modify the macro created in Chapter 3. The macro open frmQualifiedEmployees from frmSystem.

Action Arguments

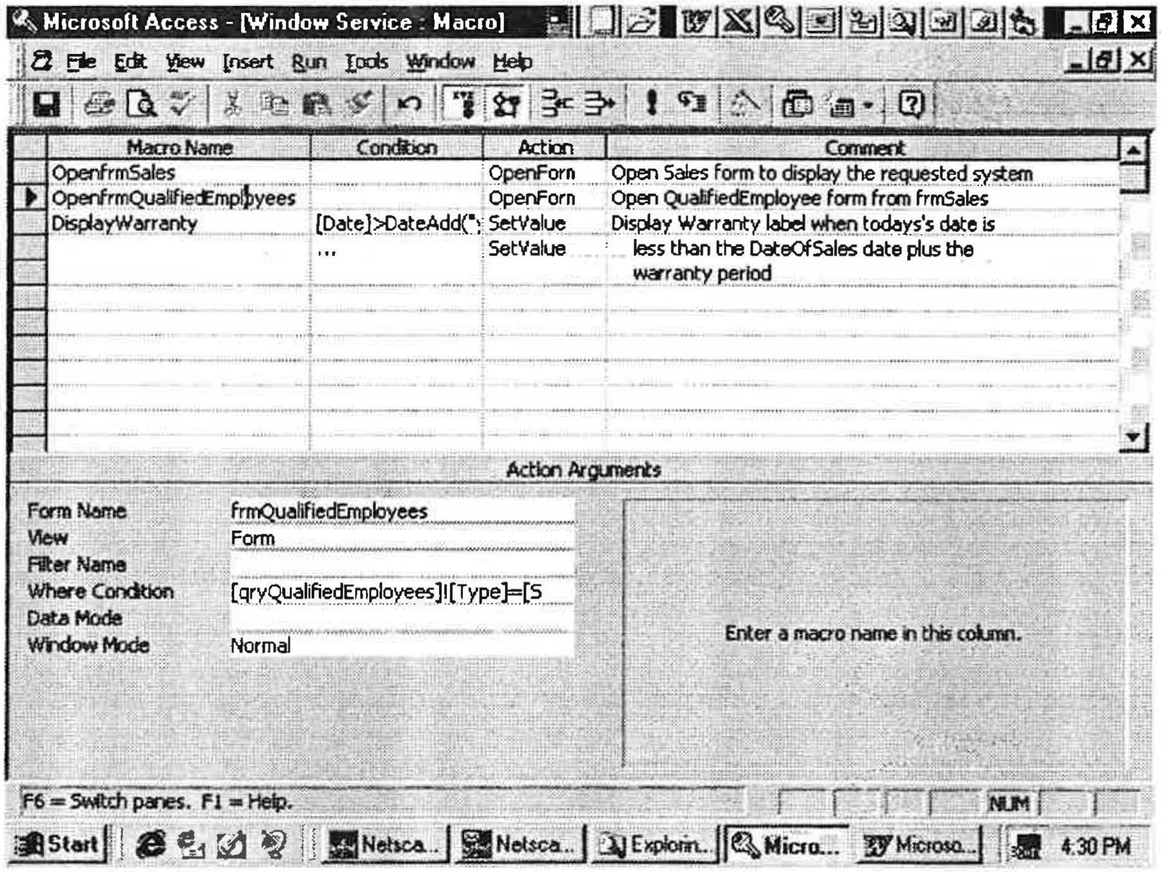
Form Name	frmQualifiedEmployees
View	Form
Filter Name	
Where Condition	[qryQualifiedEmployees]![Type]=[F
Data Mode	
Window Mode	Normal

Enter a comment in this column.

F6 = Switch panes. F1 = Help.

NUM

step 4


Microsoft Access - [Window Service : Macro]
File Edit View Insert Run Tools Window Help
Start
Netsca...
Netsca...
Explorin...
Micro...
Microso...
4:30 PM

Macro Name	Condition	Action	Comment
OpenfrmSales		OpenForn	Open Sales form to display the requested system
OpenfrmQualifiedEmployees		OpenForn	Open QualifiedEmployee form from frmSales
DisplayWarranty	[Date]>DateAdd("	SetValue	Display Warranty label when todays's date is
	...	SetValue	less than the DateOfSales date plus the warranty period

Action Arguments

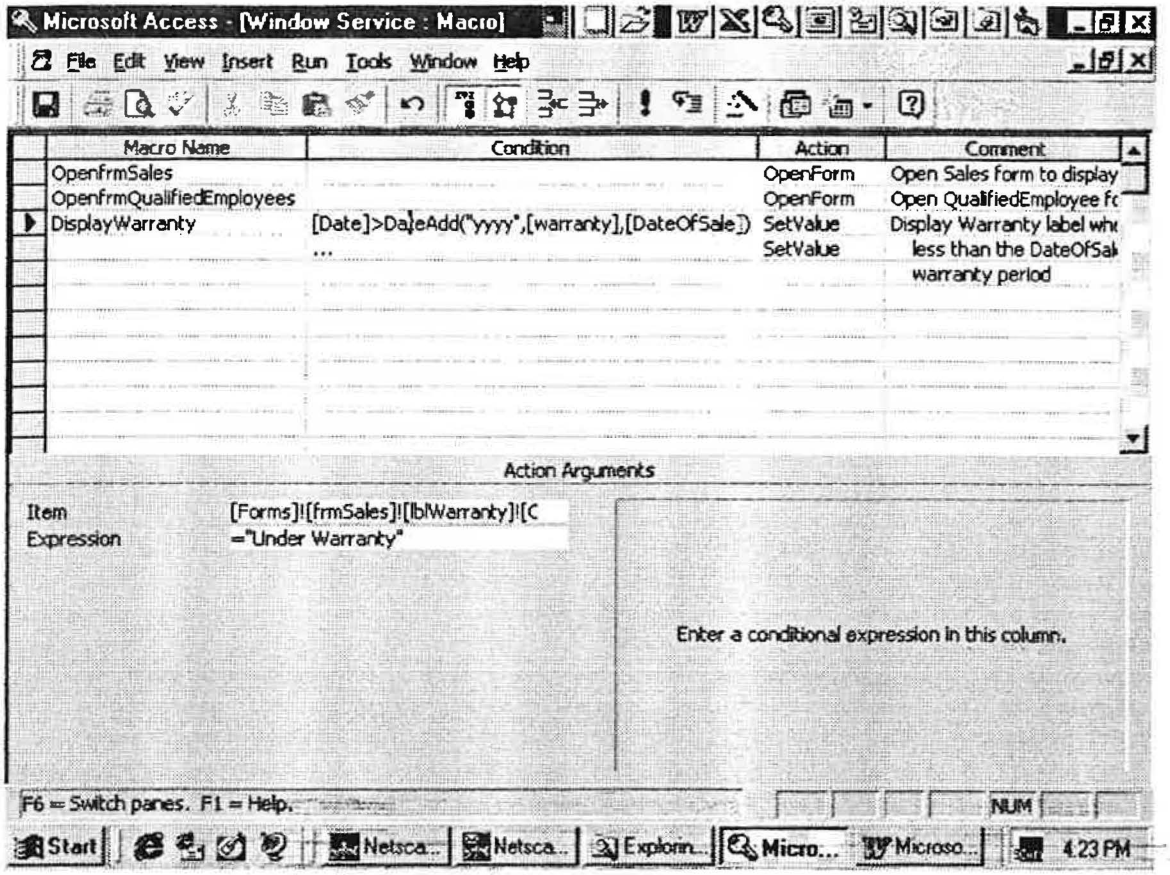
Form Name	frmQualifiedEmployees
View	Form
Filter Name	
Where Condition	[qryQualifiedEmployees]![Type]=[S
Data Mode	
Window Mode	Normal

Enter a macro name in this column.

F6 = Switch panes. F1 = Help.

NUM

step 5


Microsoft Access - [Window Service : Macro]
File Edit View Insert Run Tools Window Help
Start
Netsca...
Netsca...
Explorin...
Micro...
Microso...
4:23 PM
NUM

Macro Name	Condition	Action	Comment
OpenfrmSales		OpenForm	Open Sales form to display
OpenfrmQualifiedEmployees		OpenForm	Open QualifiedEmployee fo
DisplayWarranty	[Date]>DateAdd("yyyy",[warranty],[DateOfSale])	SetValue	Display Warranty label whe
	...	SetValue	less than the DateOfSal
			warranty period

Action Arguments

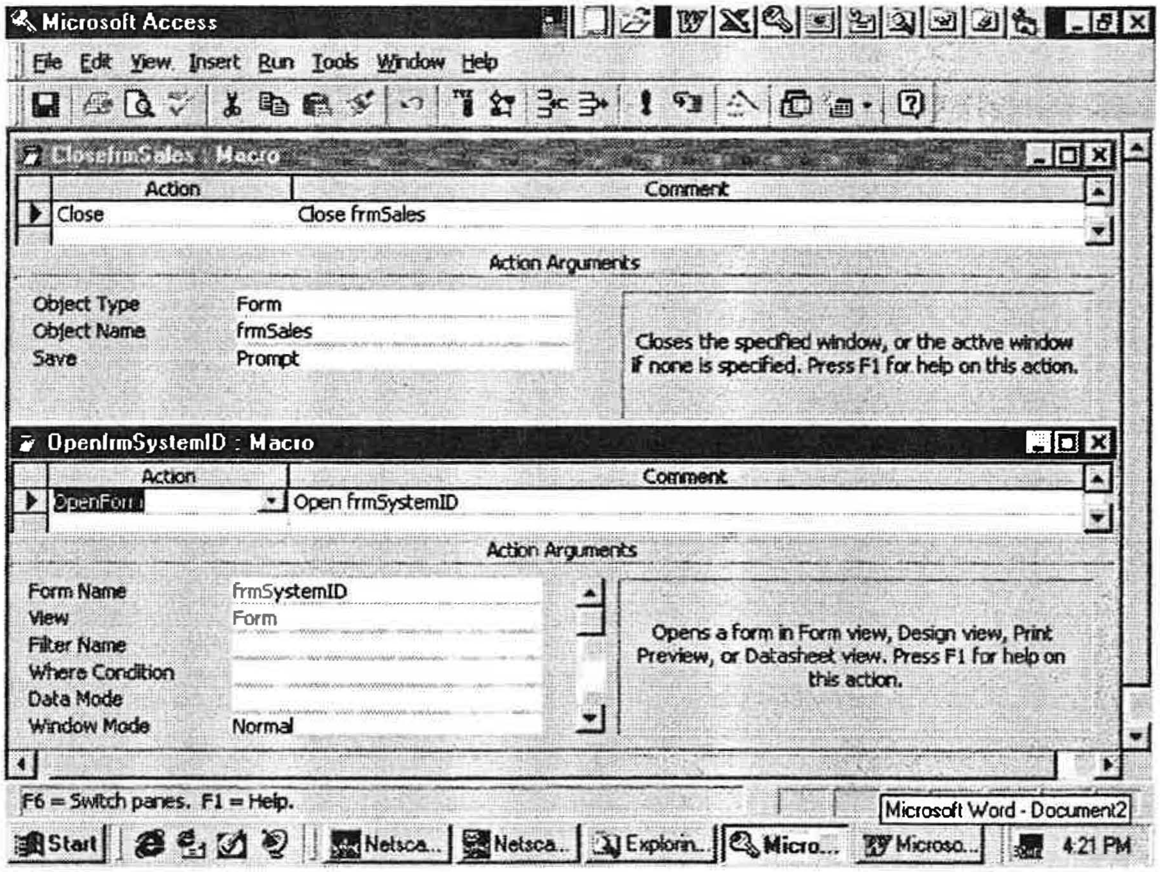
Item [Forms]![frmSales]![lblWarranty]![C

Expression ="Under Warranty"

Enter a conditional expression in this column.

F6 = Switch panes. F1 = Help.

step 6



ClosefrmSales : Macro

Action	Comment
Close	Close frmSales

Action Arguments

Object Type Form

Object Name frmSales

Save Prompt

Closes the specified window, or the active window if none is specified. Press F1 for help on this action.

OpenfrmSystemID : Macro

Action	Comment
OpenForm	Open frmSystemID

Action Arguments

Form Name frmSystemID

View Form

Filter Name

Where Condition

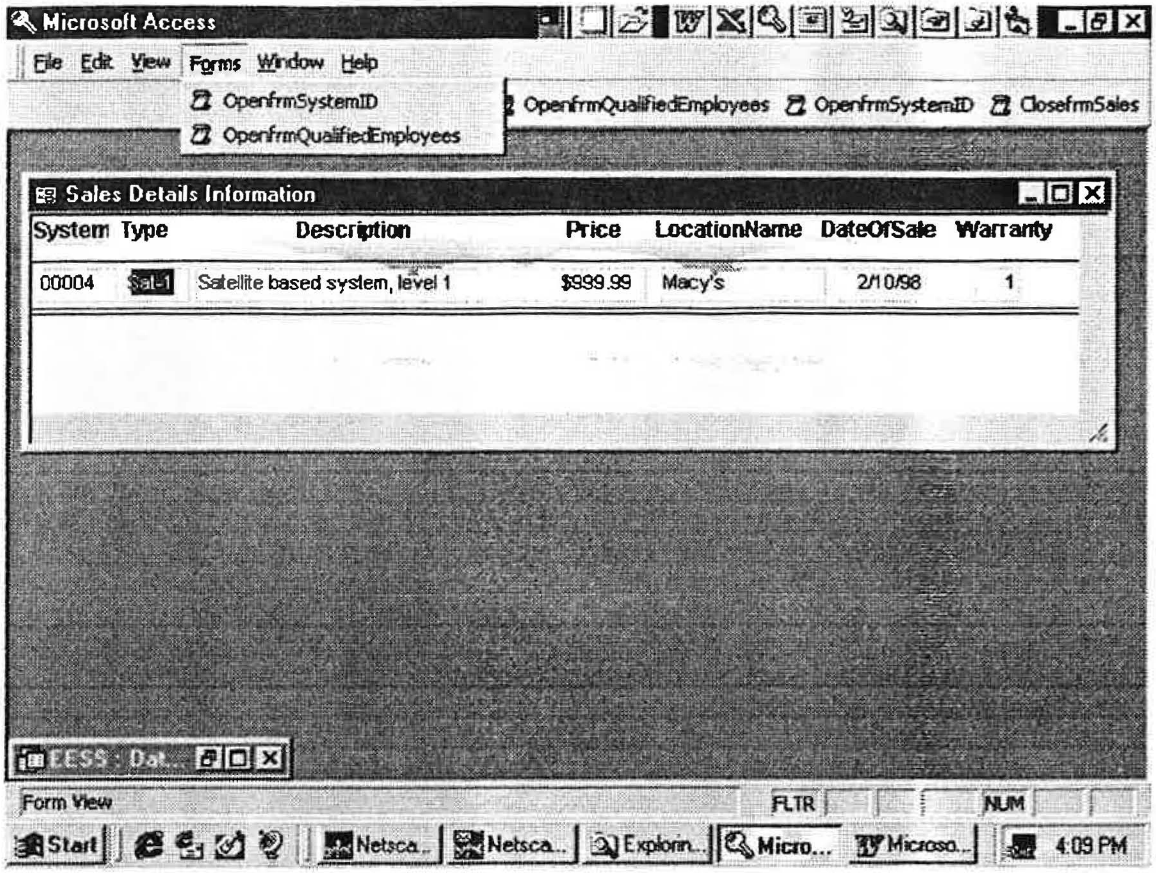
Data Mode

Window Mode Normal

Opens a form in Form view, Design view, Print Preview, or Datasheet view. Press F1 for help on this action.

F6 = Switch panes. F1 = Help.

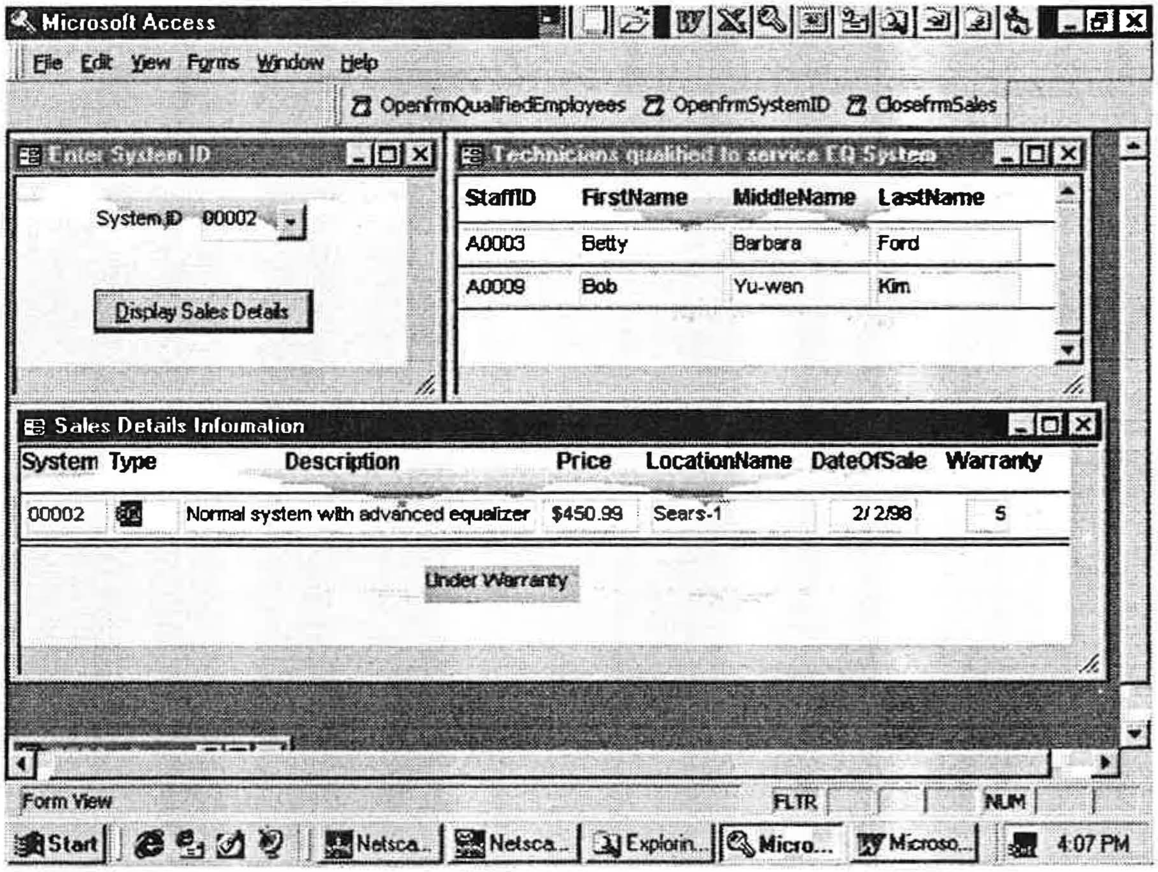
Macros needed for menubar & toolbar



Sales Details Information

System	Type	Description	Price	LocationName	DateOfSale	Warranty
00004	Sat-1	Satellite based system, level 1	$999.99	Macy's	2/10/98	1

step 7



Enter System ID

System ID 00002

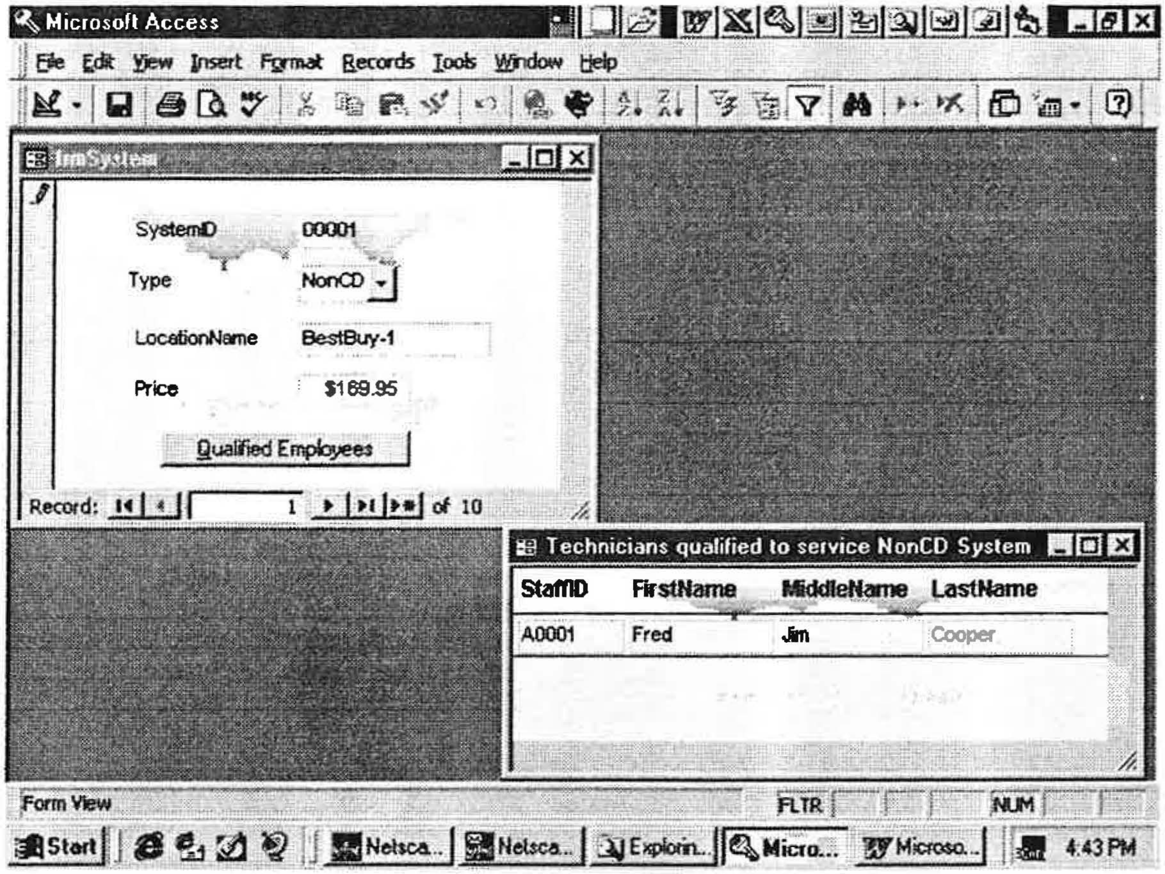
Technicians qualified to service EQ System

StaffID	FirstName	MiddleName	LastName
A0003	Betty	Barbara	Ford
A0009	Bob	Yu-wen	Kim

Sales Details Information

System	Type	Description	Price	LocationName	DateOfSale	Warranty
00002	[illegible]	Normal system with advanced equalizer	$450.99	Sears-1	2/2/98	5

Execution example for step 2 & 5-6



SystemID	00001
Type	NonCD
LocationName	BestBuy-1
Price	$169.95

Qualified Employees

Record: 1 of 10

Technicians qualified to service NonCD System

StaffID	FirstName	MiddleName	LastName
A0001	Fred	Jim	Cooper

Execution example for step 4

Appendix C

CIS38 Projects

Visual Basic

Project Evaluation Sheet

Programmer Name: shui-lien Huang   Project Number: 1

Disk Folder Name: project1   Project (.vbp) file: BnB.vbp

Date Due: 1-19-99   Date Turned In: 1-19-99

Above to be completed by student

Points (100 Possible)

Project Performance:

Project works according to specifications

Output is accurate

Logic is efficient

Meets all requirements

Disk Storage:

Folder is present and contains all necessary project files (no extra files)

Folder name and file names match those listed above

Project runs as submitted

User Interface:

Professional appearance

Layout, placement, spelling

Follows Windows standards

Keyboard access keys (& in Caption, Default and Cancel Set)

Tab order is correct

I like the Huang B'n B

External documentation:

Professional presentation

Folder: outside label, evaluation sheet, divider tabs, and disk securely attached

Printout of all forms, form text, and code

Standards and conventions:

Object names

Identifiers for Variables and Named Constants

Option Explicit / All variables declared

Remarks in General Declarations section

Programmer name

Project number

Date

Project description

Descriptive remarks in every procedure

Proper indentation

Proper blank lines

No unused procedures included

(See note on code)

Timeliness:

Extra credit:

Project Score: 100

```
'Name:           Shui-lien Huang
'Date:           Jan 12, 1999
'Folder:         Project1
'File Name:      BnBSplash
'Description:    Display the splash screen while the main form is being loaded

Private Sub cmdOK_Click()
    'Unload the splash form and display the main form

    Unload Me
    frmBnBMain.Show
End Sub
```

```
'Name:           Shui-lien Huang
'Date:           Jan 12, 1999
'Folder:         Project1
'File Name:      BnBMain
'Description:    The main form displays Huang Bed and Breakfast reservation
'                menu system

Private Sub mnuFileExit_Click()
    'Exit the project

    End
End Sub

Private Sub mnuHelpAbout_Click()
    'Display the About form

    frmAbout.Show
End Sub
```

```
'Name:           Shui-lien Huang
'Date:           Jan 12, 1999
'Folder:         Project1
'File Name:      BnBAbout
'Description:    The form displays the information about the programmar,
'                the date created, and the version number.

Private Sub cmdOK_Click()
    'Hide the current form

    Me.Hide
End Sub

Private Sub Form_Load()
    'Center the current form

    frmAbout.Top = (Screen.Height - frmAbout.Height) / 2
    frmAbout.Left = (Screen.Width - frmAbout.Width) / 2
End Sub
```

in VB and Data - use StartUp Position property

```
'Name:           Shui-lien Huang
'Date:           Jan 12, 1999
'Folder:         Project1
'File Name:      BnBModule
'Description:    Display the splash screen and load the main form

Sub Main()
    'Display the splash form while the main form is being loaded

    frmSplash.Show
    Load frmBnBMain
End Sub
```

✓

Visual Basic

Project Evaluation Sheet

Programmer Name: Shui-lien Huang Project Number: 2

Disk Folder Name: project 2 Project (.vbp) file: SnB.vbp

Date Due: 2/2/99 Date Turned In: 1/28/99

Above to be completed by student

Points (100 Possible)

Project Performance:

Project works according to specifications ________

Output is accurate ________

Logic is efficient ________

Meets all requirements ________

Disk Storage:

Folder is present and contains all necessary project files (no extra files) ________

Folder name and file names match those listed above ________

Project runs as submitted ________

User Interface:

Professional appearance ________

Layout, placement, spelling ________

Follows Windows standards ________

Keyboard access keys (& in Caption, Default and Cancel Set)

Tab order is correct

External documentation:

Professional presentation ________

Folder: outside label, evaluation sheet, divider tabs, and disk securely attached ________

Printout of all forms, form text, and code ________

Standards and conventions:

Object names ________

Identifiers for Variables and Named Constants ________

Option Explicit / All variables declared ________

Remarks in General Declarations section ________

Programmer name

Project number

Date

Project description

Descriptive remarks in every procedure ________

Proper indentation ________

Proper blank lines ________

No unused procedures included ________

Timeliness: ________

Extra credit: ________

Only also [illegible]

Boolean [illegible]

[illegible]

Only early [illegible]

Project Score: 100

89

```
'Name:          Shui-lien Huang
'Date:          Jan. 25, 1999
'Folder:        Project2
'File Name:     BnBRoom
'Description:   The Room project lets the user to retrive and modify
'               the contents of a sequential file through an array.
'               The user uses the tabbed dialog control to display
'               and update data.  The only fields can be changed by
'               the user are the price fields.

Option Explicit

Private Type RoomRecord
    strRoom             As String
    optIndex            As Integer
    chkJacuzziValue     As Integer
    chkBathValue        As Integer
    chkFirePlaceValue   As Integer
    curLowPrice         As Currency
    curHighPrice        As Currency
End Type

Dim mudtRooms(0 To 4)   As RoomRecord
Dim mblnSaved           As Boolean
Dim mblnChanged         As Boolean

Private Sub cmdOK_Click()
    'Check to see if the changes are saved before returning to main form

    Dim intResponse As Integer
    If Not mblnSaved Then
        intResponse = MsgBox("Save Changes?", vbOKCancel, "Unsaved Changes")
        If intResponse = vbOK Then
            SaveData
        End If
    End If

    Me.Hide
End Sub

Private Sub cmdSave_Click()
    'Call the procedure to save data

    SaveData
End Sub
```

90

```
Private Sub Form_Load()
    'Center the form and load data from the file into the array.
    Dim intIndex As Integer

    Me.Top = (Screen.Height - Me.Height) / 2
    Me.Left = (Screen.Width - Me.Width) / 2

    Open App.Path & "\rooms.txt" For Input As #1
    Do Until EOF(1)
        Input #1, mudtRooms(intIndex).strRoom, _
                  mudtRooms(intIndex).optIndex, _
                  mudtRooms(intIndex).chkJacuzziValue, _
                  mudtRooms(intIndex).chkBathValue, _
                  mudtRooms(intIndex).chkFirePlaceValue, _
                  mudtRooms(intIndex).curLowPrice, _
                  mudtRooms(intIndex).curHighPrice
                  TabRooms.TabCaption(intIndex) = mudtRooms(intIndex).strRoom
        intIndex = intIndex + 1
    Loop
    Close
    mblnSaved = True
    DisplayTabData (0)
End Sub

Private Sub DisplayTabData(intIndex As Integer)
    'Display rooms information from the array.
    'Data has not been changed at this time.

    mblnChanged = False
    optBed(mudtRooms(intIndex).optIndex).Value = True
    chkJacuzzi.Value = mudtRooms(intIndex).chkJacuzziValue
    chkBath.Value = mudtRooms(intIndex).chkBathValue
    chkFirePlace.Value = mudtRooms(intIndex).chkFirePlaceValue
    txtLowPrice.Text = mudtRooms(intIndex).curLowPrice
    txtHighPrice.Text = mudtRooms(intIndex).curHighPrice
End Sub

Private Sub SaveData()
    'Save everthing from the array back to the sequential file.
    Dim intIndex As Integer

    Open App.Path & "\rooms.txt" For Output As #1
    For intIndex = 0 To 4
        Write #1, mudtRooms(intIndex).strRoom, _
                  mudtRooms(intIndex).optIndex, _
                  mudtRooms(intIndex).chkJacuzziValue, _
                  mudtRooms(intIndex).chkBathValue, _
                  mudtRooms(intIndex).chkFirePlaceValue, _
                  mudtRooms(intIndex).curLowPrice, _
                  mudtRooms(intIndex).curHighPrice
    Next intIndex
    Close
    mblnSaved = True
End Sub
```

```
Private Sub TabRooms_Click(PreviousTab As Integer)
    'Display different information on the selected tab

    DisplayTabData (TabRooms.Tab)
End Sub

Private Sub txtHighPrice_GotFocus()
    'Highlight the high price text box

    txtHighPrice.SelStart = 0
    txtHighPrice.SelLength = Len(txtHighPrice.Text)
    txtHighPrice.SetFocus
    mblnChanged = True
End Sub

Private Sub txtHighPrice_Change()
    'High price has been changed

    If mblnChanged Then
        mblnSaved = False
        mudtRooms(TabRooms.Tab).curHighPrice = Val(txtHighPrice.Text)
    End If
End Sub

Private Sub txtLowPrice_GotFocus()
    'Highlight the low price text box

    txtLowPrice.SelStart = 0
    txtLowPrice.SelLength = Len(txtLowPrice.Text)
    txtLowPrice.SetFocus
    mblnChanged = True
End Sub

Private Sub txtLowPrice_Change()
    'Low price has been changed

    If mblnChanged Then
        mblnSaved = False
        mudtRooms(TabRooms.Tab).curLowPrice = Val(txtLowPrice.Text)
    End If
End Sub
```

Visual Basic

Project Evaluation Sheet

Programmer Name: Shui-lien Huang ✓ Project Number: 3

Disk Folder Name: Project3 Project (.vbp) file: BnB.vbp

Date Due: 2/23/99 Date Turned In: 2/23/99

Above to be completed by student

Points (100 Possible)

Project Performance:

Project works according to specifications ______

Output is accurate ______

Logic is efficient ______

Meets all requirements ______

Disk Storage:

Folder is present and contains all necessary project files (no extra files) ______

Folder name and file names match those listed above ______

Project runs as submitted ______

User Interface:

Professional appearance ______

Layout, placement, spelling ______

Follows Windows standards ______

Keyboard access keys (& in Caption, Default and Cancel Set)

Tab order is correct

External documentation:

Professional presentation ______

Folder: outside label, evaluation sheet, divider tabs, and disk securely attached ______

Printout of all forms, form text, and code ______

Standards and conventions:

Object names ______

Identifiers for Variables and Named Constants ______

Option Explicit / All variables declared ______

Remarks in General Declarations section ______

Programmer name

Project number

Date

Project description

Descriptive remarks in every procedure ______

Proper indentation ______

Proper blank lines ______

No unused procedures included ______

Timeliness: -3

Extra credit: ______

Project Score: 97

93

```
'Name:          Shui-lien Huang
'Date:          Feb 23, 1999
'Folder:        Project3
'File Name:     frmBnBRoom
'Description:   Add error handling and QueryUnload event to this form
'               which was created in Project2.

Option Explicit

Private Type RoomRecord
    strRoom             As String
    optIndex            As Integer
    chkJacuzziValue     As Integer
    chkBathValue        As Integer
    chkFirePlaceValue   As Integer
    curLowPrice         As Currency
    curHighPrice        As Currency
End Type

Dim mudtRooms(0 To 4)   As RoomRecord
Dim mblnSaved           As Boolean
Dim mblnChanged         As Boolean

Private Sub cmdOK_Click()
    'Check to see if the changes are saved before returning to main form

    Dim intResponse As Integer
    If Not mblnSaved Then
        intResponse = MsgBox("Save Changes?", vbOKCancel, "Unsaved Changes")
        If intResponse = vbOK Then
            SaveData
        End If
    End If

    Me.Hide
End Sub

Private Sub cmdSave_Click()
    'Call the procedure to save data

    SaveData
End Sub
```

```
Private Sub Form_Load()
    'Center the form and load data from the file into the array.
    Dim intIndex As Integer

    Me.Top = (Screen.Height - Me.Height) / 2
    Me.Left = (Screen.Width - Me.Width) / 2

    On Error GoTo HandleErrors
    Open App.Path & "\rooms.txt" For Input As #1
    Do Until EOF(1)
        Input #1, mudtRooms(intIndex).strRoom, _
                  mudtRooms(intIndex).optIndex, _
                  mudtRooms(intIndex).chkJacuzziValue, _
                  mudtRooms(intIndex).chkBathValue, _
                  mudtRooms(intIndex).chkFirePlaceValue, _
                  mudtRooms(intIndex).curLowPrice, _
                  mudtRooms(intIndex).curHighPrice
                  TabRooms.TabCaption(intIndex) = mudtRooms(intIndex).strRoom
        intIndex = intIndex + 1
    Loop
    Close
    mblnSaved = True
    DisplayTabData (0)
    Exit Sub

HandleErrors:
    Select Case Err.Number
        Case 53
            MsgBox "You may mispell the filename", , "File not found"
            End
        Case 71
            MsgBox "Please insert the correct disk", , "Disk not ready"
            Resume
        Case 76
            MsgBox "Please check the path", , "Path not found"
            Resume
        Case Else
            On Error GoTo 0
    End Select
End Sub

Private Sub DisplayTabData(intIndex As Integer)
    'Display rooms information from the array.
    'Data has not been changed at this time.

    mblnChanged = False
    optBed(mudtRooms(intIndex).optIndex).Value = True
    chkJacuzzi.Value = mudtRooms(intIndex).chkJacuzziValue
    chkBath.Value = mudtRooms(intIndex).chkBathValue
    chkFirePlace.Value = mudtRooms(intIndex).chkFirePlaceValue
    txtLowPrice.Text = mudtRooms(intIndex).curLowPrice
    txtHighPrice.Text = mudtRooms(intIndex).curHighPrice
End Sub
```

```
Private Sub SaveData()
    'Save everthing from the array back to the sequential file.
    Dim intIndex As Integer

    Open App.Path & "\rooms.txt" For Output As #1
    For intIndex = 0 To 4
        Write #1, mudtRooms(intIndex).strRoom, _
                  mudtRooms(intIndex).optIndex, _
                  mudtRooms(intIndex).chkJacuzziValue, _
                  mudtRooms(intIndex).chkBathValue, _
                  mudtRooms(intIndex).chkFirePlaceValue, _
                  mudtRooms(intIndex).curLowPrice, _
                  mudtRooms(intIndex).curHighPrice
    Next intIndex
    Close
    mblnSaved = True
End Sub

Private Sub Form_QueryUnload(Cancel As Integer, UnloadMode As Integer)
    'Check to see if the user has changed the data

    cmdOK_Click
End Sub

Private Sub TabRooms_Click(PreviousTab As Integer)
    'Display different information on the selected tab

    DisplayTabData (TabRooms.Tab)
End Sub

Private Sub txtHighPrice_GotFocus()
    'Highlight the high price text box

    txtHighPrice.SelStart = 0
    txtHighPrice.SelLength = Len(txtHighPrice.Text)
    txtHighPrice.SetFocus
    mblnChanged = True
End Sub

Private Sub txtHighPrice_Change()
    'High price has been changed

    If mblnChanged Then
        mblnSaved = False
        mudtRooms(TabRooms.Tab).curHighPrice = Val(txtHighPrice.Text)
    End If
End Sub

Private Sub txtLowPrice_GotFocus()
    'Highlight the low price text box

    txtLowPrice.SelStart = 0
    txtLowPrice.SelLength = Len(txtLowPrice.Text)
    txtLowPrice.SetFocus
    mblnChanged = True
End Sub
```

```
Private Sub txtLowPrice_Change()
    'Low price has been changed

    If mblnChanged Then
        mblnSaved = False
        mudtRooms(TabRooms.Tab).curLowPrice = Val(txtLowPrice.Text)
    End If
End Sub
```

```
'Name:         Shui-lien Huang
'Date:         Feb 20, 1999
'Folder:       Project3
'File Name:    BnBGuestMain.frm
'Description:  Maintain a list of guests

Option Explicit

Private Sub cmdAdd_Click()
    'Display the Guest file and add the item to the list

    With frmGuest
        .Action = "A"
        .Show vbModal
    End With
End Sub

Private Sub cmdBrowse_Click()
    'Display first Guest in list

    With lstGuest
        .ListIndex = 0                      'Browse from the first record
        frmGuest.Key = Trim(Str(.ItemData(0)))
    End With

    With frmGuest
        .Action = "B"
        .Show vbModal
    End With
End Sub

Private Sub cmdClose_Click()
    'Unload the Guest forms

    Unload frmGuest
    Unload Me
End Sub

Private Sub cmdDisplay_Click()
    'Display the Guest member

    frmGuest.Action = "D"
    DisplayForm
End Sub

Private Sub cmdEdit_Click()
    'Edit the Guest member

    frmGuest.Action = "E"
    DisplayForm
End Sub
```

```
Private Sub cmdRemove_Click()
    'Remove the Guest member

    frmGuest.Action = "R"
    DisplayForm
End Sub

Private Sub Form_Load()
    'Force file creation from Guest form

    Load frmGuest

    Me.Top = (Screen.Height - Me.Height) / 2
    Me.Left = (Screen.Width - Me.Width) / 2
End Sub

Private Sub DisplayForm()
    'Set key and show frmGuest
    Dim strMsg As String

    If lstGuest.ListIndex <> -1 Then
        With lstGuest
            frmGuest.Key = Trim(Str(.ItemData(.ListIndex)))
        End With
        frmGuest.Show vbModal
    Else
        strMsg = "Please select a Guest from the list"
        MsgBox strMsg, vbInformation, "Guest File"
    End If
End Sub
```

```
'Name:          Shui-lien Huang
'Date:          Feb 20, 1999
'Folder:        Project3
'File Name:     BnBGuest.frm
'Description:   The form is used to add, display, edit, browse
'               or remove a guest from a random file.

Option Explicit
Private mGuests      As CGuests
Private mstrKey      As String       'Key property of form
Private mstrAction   As String       'Action property of form

Private Sub cmdCancel_Click()
    'Return to main form with no action

    Me.Hide
End Sub

Private Sub cmdOK_Click()
    'Choose action depending upon the seleted command
    Dim strName     As String
    Dim strKey      As String
    Dim strMsg      As String
    Dim intResponse As Integer
    Dim intKey      As Integer

    Select Case mstrAction
        Case "A"                          'Add to listbox and set ItemData
            strName = Trim(txtLastName) & ", " & txtFirstName
            With frmGuestMain.lstGuest
                .AddItem strName
                intKey = Val(mGuests.HighestKey) + 1
                .ItemData(.NewIndex) = intKey
            End With

            'Add to collection and file
            strKey = Trim(Str(intKey))
            mGuests.Add txtLastName, txtFirstName, txtStreet, txtCity, _
                            txtState, txtZip, txtCountry, txtPhone, _
                            txtEmail, txtLastVisitDate, txtRoom, strKey
            Me.Hide
        Case "D"                          'Display the selected guest
            Me.Hide
        Case "B"                          'Display next Guest
            With frmGuestMain.lstGuest
                If .ListIndex < .ListCount - 1 Then
                    .ListIndex = .ListIndex + 1
                    mstrKey = Trim(.ItemData(.ListIndex))
                    DisplayData
                Else
                    strMsg = "No more Guest records to display"
                    MsgBox strMsg, vbInformation, "Browse Guest Information"
                    Me.Hide
                End If
            End With
```

```
         Case "E"                         'Update list box
            With frmGuestMain.lstGuest
                .RemoveItem .ListIndex
                strName = Trim(txtLastName) & ", " & txtFirstName
                .AddItem strName
                .ItemData(.NewIndex) = Val(mstrKey)
            End With

            'Update object in collection and file
            With mGuests(mstrKey)
                .LastName = txtLastName
                .FirstName = txtFirstName
                .Street = txtStreet
                .City = txtCity
                .State = txtState
                .ZipCode = txtZip
                .Country = txtCountry
                .Phone = txtPhone
                .Email = txtEmail
                .LastVisitDate = txtLastVisitDate
                .Room = txtRoom
            End With

            mGuests.SaveRecord mstrKey    'Save changes in file
            Me.Hide
        Case "R"                          'Remove from list box
            With frmGuestMain.lstGuest
                .RemoveItem .ListIndex
            End With

            'Remove from cllection and file
            mGuests.Remove mstrKey
            Me.Hide
    End Select
End Sub

Private Sub Form_Activate()
    'Set up the form for the selected action

    Select Case mstrAction
        Case "A"
            UnlockTheControls
            ClearTextBoxes
            txtLastName.SetFocus
            lblCommand.Caption = "Add New Guest Information"
        Case "D"
            DisplayData
            LockTheControls
            lblCommand.Caption = "Display Guest Information"
        Case "B"
            DisplayData
            LockTheControls
            lblCommand.Caption = "Browse Guest Information"
        Case "E"
            DisplayData
            UnlockTheControls
            lblCommand.Caption = "Edit Guest Information"
        Case "R"
            DisplayData
            LockTheControls
            lblCommand.Caption = "Remove This Record?"
    End Select
End Sub

Private Sub Form_Load()
    'Create the Guest collection object

    Dim Guest       As CGuest
    Dim strName     As String
    Dim intResponse As Integer
    Dim strMsg      As String

    Set mGuests = New CGuests

    If mGuests.FileOpened Then
        For Each Guest In mGuests
            strName = Trim(Guest.LastName) & ", " & Guest.FirstName
            With frmGuestMain.lstGuest
                .AddItem strName
                .ItemData(.NewIndex) = (Val(Guest.GuestCode))
            End With
        Next
    Else
        strMsg = "File does not exist.  Create new file?"
        intResponse = MsgBox(strMsg, vbQuestion + vbYesNo, "Guest File")
        If intResponse = vbYes Then
            mGuests.OpenNewFile
        Else
            Set mGuests = Nothing
            Set Guest = Nothing
            End
        End If
    End If

    Me.Top = (Screen.Height - Me.Height) / 2
    Me.Left = (Screen.Width - Me.Width) / 2
End Sub

Private Sub Form_Unload(Cancel As Integer)
    'Remove the object from memory

    Set mGuests = Nothing
End Sub
```

```
Private Sub DisplayData()
    'Transfer from the selected object of the collection to text fields

    With mGuests(mstrKey)                     'Omit default property Item
        txtLastName = .LastName               'Place insertion point in object
mGuests.Item
        txtFirstName = .FirstName             'Tools - Procedure Attribute -
        txtStreet = .Street                   'Set Procedure ID to default
        txtCity = .City
        txtState = .State
        txtZip = .ZipCode
        txtCountry = .Country
        txtPhone = .Phone
        txtEmail = .Email
        txtLastVisitDate = .LastVisitDate
        txtRoom = .Room
    End With
End Sub

Private Sub ClearTextBoxes()
    'Clear all text boxes

    txtLastName = ""
    txtFirstName = ""
    txtStreet = ""
    txtCity = ""
    txtState = ""
    txtZip = ""
    txtCountry = ""
    txtPhone = ""
    txtEmail = ""
    txtLastVisitDate = ""
    txtRoom = ""
End Sub

Private Sub LockTheControls()
    'Do not allow changes

    txtLastName.Locked = True
    txtFirstName.Locked = True
    txtStreet.Locked = True
    txtCity.Locked = True
    txtState.Locked = True
    txtZip.Locked = True
    txtCountry.Locked = True
    txtPhone.Locked = True
    txtEmail.Locked = True
    txtLastVisitDate.Locked = True
    txtRoom.Locked = True
End Sub
```

```
Private Sub UnlockTheControls()
    'Do allow changes

    txtLastName.Locked = False
    txtFirstName.Locked = False
    txtStreet.Locked = False
    txtCity.Locked = False
    txtState.Locked = False
    txtZip.Locked = False
    txtCountry.Locked = False
    txtPhone.Locked = False
    txtEmail.Locked = False
    txtLastVisitDate.Locked = False
    txtRoom.Locked = False
End Sub

Public Property Let Key(ByVal strKey As String)
    'Write-only property to pass selected key value to form

    mstrKey = strKey
End Property

Public Property Let Action(ByVal strAction As String)
    'Write-only property to pass action to form

    mstrAction = strAction
End Property
```

```
'Name:          Shui-lien Huang
'Date:          Feb 20, 1999
'Folder:        Project3
'File Name:     CGuest.cls
'Description:   Declare properties for the CGuess class

Option Explicit

Private mstrLastName        As String
Private mstrFirstName       As String
Private mstrStreet          As String
Private mstrCity            As String
Private mstrState           As String
Private mstrZipCode         As String
Private mstrCountry         As String
Private mstrPhone           As String
Private mstrEmail           As String
Private mstrLastVisitDate   As String
Private mstrRoom            As String
Private mstrGuestCode       As String

Public Property Get LastName() As String
    'Retrieve the current value

    LastName = mstrLastName                 'LastName is an external name
End Property

Public Property Let LastName(ByVal strLastName As String)
    'Assign the property value

    mstrLastName = strLastName              'strLastName is local variable
End Property

Public Property Get FirstName() As String
    'Retrieve the current value

    FirstName = mstrFirstName
End Property

Public Property Let FirstName(ByVal strFirstName As String)
    'Assign the property value

    mstrFirstName = strFirstName
End Property

Public Property Get Street() As String
    'Retrieve the current value

    Street = mstrStreet
End Property

Public Property Let Street(ByVal strStreet As String)
    'Assign the property value

    mstrStreet = strStreet
End Property
```

```
Public Property Get City() As String
    'Retrieve the current value

    City = mstrCity
End Property

Public Property Let City(ByVal strCity As String)
    'Assign the property value

    mstrCity = strCity
End Property

Public Property Get State() As String
    'Retrieve the current value

    State = mstrState
End Property

Public Property Let State(ByVal strState As String)
    'Assign the property value

    mstrState = strState
End Property

Public Property Get ZipCode() As String
    'Retrieve the current value

    ZipCode = mstrZipCode
End Property

Public Property Let ZipCode(ByVal strZipCode As String)
    'Assign the property value

    mstrZipCode = strZipCode
End Property

Public Property Get Country() As String
    'Retrieve the current value

    Country = mstrCountry
End Property

Public Property Let Country(ByVal strCountry As String)
    'Assign the property value

    mstrCountry = strCountry
End Property

Public Property Get Phone() As String
    'Retrieve the current value

    Phone = mstrPhone
End Property
```

```
Public Property Let Phone(ByVal strPhone As String)
    'Assign the property value

    mstrPhone = strPhone
End Property

Public Property Get Email() As String
    'Retrieve the current value

    Email = mstrEmail
End Property

Public Property Let Email(ByVal strEmail As String)
    'Assign the property value

    mstrEmail = strEmail
End Property

Public Property Get LastVisitDate() As String
    'Retrieve the current value

    LastVisitDate = mstrLastVisitDate
End Property

Public Property Let LastVisitDate(ByVal strLastVisitDate As String)
    'Assign the property value

    mstrLastVisitDate = strLastVisitDate
End Property

Public Property Get Room() As String
    'Retrieve the current value

    Room = mstrRoom
End Property

Public Property Let Room(ByVal strRoom As String)
    'Assign the property value

    mstrRoom = strRoom
End Property

Public Property Get GuestCode() As String
    'Retrieve the current value

    GuestCode = mstrGuestCode
End Property

Public Property Let GuestCode(ByVal strGuestCode As String)
    'Assign the property value

    mstrGuestCode = strGuestCode
End Property
```

```
'Name:          Shui-lien Huang
'Date:          Feb 20, 1999
'Folder:        Project3
'File Name:     CGuests.cls
'Description:   Manage a collection of Guest objects

Option Explicit
Private mGuests              As Collection
Private mstrFilePath         As String
Private mstrHighestKey       As String
Private mblnFileOpened       As Boolean
Private mblnInitComplete     As Boolean
Private mintFileNumber       As Integer

Private Type udtGuest
    strLastName         As String * 15
    strFirstName        As String * 10
    strStreet           As String * 25
    strCity             As String * 15
    strState            As String * 2
    strZipCode          As String * 5
    strCountry          As String * 20
    strPhone            As String * 14
    strEmail            As String * 22
    strLastVisitDate    As String * 8
    strRoom             As String * 8
    strGuestCode        As String * 4
    strDeleteCode       As String * 1      'D=deleted, A=active
End Type

Dim mudtGuest As udtGuest

Public Sub Add(ByVal strLastName As String, _
            ByVal strFirstName As String, _
            ByVal strStreet As String, _
            ByVal strCity As String, _
            ByVal strState As String, _
            ByVal strZipCode As String, _
            ByVal strCountry As String, _
            ByVal strPhone As String, _
            ByVal strEmail As String, _
            ByVal strLastVisitDate As String, _
            ByVal strRoom As String, _
            ByVal strGuestCode As String)

    'Add a member to the collection
    Dim NewGuest As New CGuest
    With NewGuest
        .LastName = strLastName
        .FirstName = strFirstName
        .Street = strStreet
        .City = strCity
        .State = strState
        .ZipCode = strZipCode
        .Country = strCountry
        .Phone = strPhone
        .Email = strEmail
        .LastVisitDate = strLastVisitDate
        .Room = strRoom
        .GuestCode = Trim(strGuestCode)

        mGuests.Add NewGuest, .GuestCode  'Add the object to collection

        'Write new record to the disk
        If mblnInitComplete Then       'Don't write during initialization
            SetupRecord .GuestCode
            mudtGuest.strDeleteCode = "A" 'Active record
            WriteRecord .GuestCode
            mstrHighestKey = .GuestCode   'Save new key as highest
        End If
    End With
End Sub

Public Sub Remove(ByVal strKey As String)
    'Remove a guest

    'Mark record as deleted in disk file
    SetupRecord strKey
    mudtGuest.strDeleteCode = "D"
    WriteRecord strKey

    'Remove from collection (must do after disk write)
    mGuests.Remove strKey
End Sub

Public Sub SaveRecord(ByVal strKey As String)
    'Save changes in file

    SetupRecord strKey
    WriteRecord strKey
End Sub

Public Function Item(ByVal strKey As String) As CGuest
    'Select a member from the collection
    'Set procureID to make this the default property of the class

    Set Item = mGuests.Item(strKey)
End Function
```

```
Public Sub Class_Initialize()
    'Create the collection object & read the data file

    Set mGuests = New Collection
    ReadFile
    mblnInitComplete = True     'All future adds must be written to file
End Sub

Public Sub Class_Terminate()
    'Close the file and remove the collection from memory

    Close #mintFileNumber
    Set mGuests = Nothing
End Sub

Public Function NewEnum()
    'Allow for For Each...Next enumeration, the following step is needed
    'Place insertion point in the Function statement
    'Tools - Procedure Attributes - Set procedureID to -4

    Set NewEnum = mGuests.[_NewEnum]
End Function

Private Sub ReadFile()
    'read all Guest records into collection
    Dim intNumberRecords    As Integer
    Dim intIndex            As Integer

    'Get the file number and open the file
    mintFileNumber = FreeFile
    mstrFilePath = App.Path & "\Guest.txt"
    If Dir(mstrFilePath) <> "" Then       'The file exists
        On Error GoTo HandleErrors
        Open mstrFilePath For Random As #mintFileNumber Len=Len(mudtGuest)
        intNumberRecords = LOF(mintFileNumber) / Len(mudtGuest)
        For intIndex = 1 To intNumberRecords
            Get #mintFileNumber, , mudtGuest
                With mudtGuest
                    If .strDeleteCode <> "D" Then
                        Add .strLastName, .strFirstName, .strStreet, _
                              .strCity, .strState, .strZipCode, _
                              .strCountry, .strPhone, .strEmail, _
                              .strLastVisitDate, .strRoom, _
                              Trim(.strGuestCode)
                    End If
                End With
        Next intIndex
        mstrHighestKey = Trim(Str(intNumberRecords))
        mblnFileOpened = True
    Else
        mstrHighestKey = "0"
        mblnFileOpened = False
    End If
```

```
ReadFileExit:
    Exit Sub

HandleErrors:
    mstrHighestKey = "0"
    mblnFileOpened = False
    On Error GoTo 0                        'Turn off error handling
End Sub

Private Sub SetupRecord(strKey)
    'Setu up the record for the current Guest object

    With mGuests(strKey)
        mudtGuest.strLastName = .LastName
        mudtGuest.strFirstName = .FirstName
        mudtGuest.strStreet = .Street
        mudtGuest.strCity = .City
        mudtGuest.strState = .State
        mudtGuest.strZipCode = .ZipCode
        mudtGuest.strCountry = .Country
        mudtGuest.strPhone = .Phone
        mudtGuest.strEmail = .Email
        mudtGuest.strLastVisitDate = .LastVisitDate
        mudtGuest.strRoom = .Room
        mudtGuest.strGuestCode = .GuestCode
    End With
End Sub

Private Sub WriteRecord(strKey)
    'Write the record for one Guest
    Dim intRecNum As Integer

    intRecNum = Val(strKey)
    Put #mintFileNumber, intRecNum, mudtGuest
End Sub

Public Property Get HighestKey() As String
    'Return current highest key property - read only

    HighestKey = mstrHighestKey
End Property

Public Property Get FileOpened() As Boolean
    'Return current value

    FileOpened = mblnFileOpened
End Property

Public Sub OpenNewFile()
    'Open new empty file

    Open mstrFilePath For Random As #mintFileNumber Len = Len(mudtGuest)
End Sub
```

Visual Basic

Project Evaluation Sheet

Programmer Name: Shui Lin Huang Project Number: 4

Disk Folder Name: project 4 Project (.vbp) file: Bns

Date Due: 3-9-99 Date Turned In: 3-1-99

Above to be completed by student

Points (100 Possible)

Project Performance:

Project works according to specifications ______

Output is accurate ______

Logic is efficient ______

Meets all requirements ______

Disk Storage:

Folder is present and contains all necessary project files (no extra files) ______

Folder name and file names match those listed above ______

Project runs as submitted ______

User Interface:

Professional appearance ______

Layout, placement, spelling ______

Follows Windows standards ______

Keyboard access keys (& in Caption, Default and Cancel Set)

Tab order is correct

External documentation:

Professional presentation ______

Folder: outside label, evaluation sheet, divider tabs, and disk securely attached ______

Printout of all forms, form text, and code ______

Standards and conventions:

Object names ______

Identifiers for Variables and Named Constants ______

Option Explicit / All variables declared ______

Remarks in General Declarations section ______

Programmer name

Project number

Date

Project description

Descriptive remarks in every procedure ______

Proper indentation ______

Proper blank lines ______

No unused procedures included ______

Timeliness: +10

Extra credit: ______

Project Score: 110

112

```
'Name:          Shui-lien Huang
'Date:          Feb 24, 1999
'Folder:        Project4
'File Name:     BnBGuest.frm
'Description:   The form is used to work with guest records from a
'               database table.  The user can use menu and command buttons
'               to add, save, delete records.  Instead of using the data
'               control to navigate the guest records, the command buttons
'               are used to move next, previous, first and last record.

Option Explicit

Private Sub cmdAdd_Click()
    'Add a new Guest

    If cmdAdd.Caption = "&Add" Then
        datGuest.Recordset.AddNew
        txtLastName.SetFocus
        DisableButtons
        cmdSave.Enabled = True
        cmdAdd.Caption = "&Cancel"
        lblCommand.Caption = "Add a New Guest"
    Else
        CancelAdd
    End If
End Sub

Private Sub CancelAdd()
    'Cancel without adding the record

    datGuest.Recordset.CancelUpdate
    EnableButtons
    cmdSave.Enabled = False
    cmdAdd.Caption = "&Add"
    lblCommand.Caption = "List Guest"
End Sub

Private Sub cmdClose_Click()
    'Return to parent form

    Me.Hide
End Sub
```

```
Private Sub cmdDelete_Click()
    'Delete the current Guest
    Dim intResponse As Integer
    Dim strMsg      As String

    strMsg = "Are you sure?"
    intResponse = MsgBox(strMsg, vbYesNo, "Delete the current record")
    If intResponse = vbYes Then
        With datGuest.Recordset
            .Delete
            .MoveNext
            If .EOF Then
                .MovePrevious
                If .BOF Then
                    MsgBox "The recordset is empty", vbInformation, "No
Records"
                    DisableButtons
                End If
            End If
        End With
    End If
End Sub

Private Sub cmdFirst_Click()
    'Move to first record

    datGuest.Recordset.MoveFirst
End Sub

Private Sub cmdLast_Click()
    'Move to last record

    datGuest.Recordset.MoveLast
End Sub

Private Sub cmdNext_Click()
    'Move to next record

    With datGuest.Recordset
        .MoveNext
        If .EOF Then
            .MoveFirst
        End If
    End With
End Sub

Private Sub cmdPrevious_Click()
    'Move to previous record

    With datGuest.Recordset
        .MovePrevious
        If .BOF Then
            .MoveLast
        End If
    End With
End Sub
```

```
Private Sub cmdSave_Click()
    'Save the current Guest

    datGuest.Recordset.Update
    EnableButtons
    cmdSave.Enabled = False
    cmdAdd.Caption = "&Add"
    lblCommand.Caption = "Guest List"
End Sub

Private Sub Form_Load()
    'Center the Guest form

    Me.Top = (Screen.Height - Me.Height) / 2
    Me.Left = (Screen.Width - Me.Width) / 2
End Sub

Private Sub DisableButtons()
    'Disable navigation buttons

    cmdNext.Enabled = False
    cmdPrevious.Enabled = False
    cmdFirst.Enabled = False
    cmdLast.Enabled = False
    cmdDelete.Enabled = False
    cmdClose.Enabled = False
    mnuFileClose.Enabled = False
    mnuGuestNew.Enabled = False
End Sub

Private Sub EnableButtons()
     'Enable navigation buttons

    cmdNext.Enabled = True
    cmdPrevious.Enabled = True
    cmdFirst.Enabled = True
    cmdLast.Enabled = True
    cmdDelete.Enabled = True
    cmdClose.Enabled = True
    mnuFileClose.Enabled = True
    mnuGuestNew.Enabled = True
    mnuGuestReturning.Enabled = True
End Sub

Private Sub mnuFileClose_Click()
    'Call the cmdClose_click event

    cmdClose_Click
End Sub

Private Sub mnuGuestNew_Click()
    cmdAdd_Click            'Call the cmdAdd_click event
End Sub

Private Sub mnuGuestReturning_Click()
    CancelAdd               'Call the CancelAdd sub procedure
End Sub
```

Visual Basic

Project Evaluation Sheet

Programmer Name: shui-lien Huang Project Number: 5

Disk Folder Name: project 5 Project (.vbp) file: BnB

Date Due: 4/6/99 Date Turned In: 4/6/99

Above to be completed by student

Points (100 Possible)

Project Performance:

Project works according to specifications ______

Output is accurate ______

Logic is efficient ______

Meets all requirements ______

Disk Storage:

Folder is present and contains all necessary project files (no extra files) ______

Folder name and file names match those listed above ______

Project runs as submitted ______

User Interface:

Professional appearance ______

Layout, placement, spelling ______

Follows Windows standards ______

Keyboard access keys (& in Caption, Default and Cancel Set)

Tab order is correct

External documentation:

Professional presentation ______

Folder: outside label, evaluation sheet, divider tabs, and disk securely attached ______

Printout of all forms, form text, and code ______

Standards and conventions:

Object names ______

Identifiers for Variables and Named Constants ______

Option Explicit / All variables declared ______

Remarks in General Declarations section ______

Programmer name

Project number

Date

Project description

Descriptive remarks in every procedure ______

Proper indentation ______

Proper blank lines ______

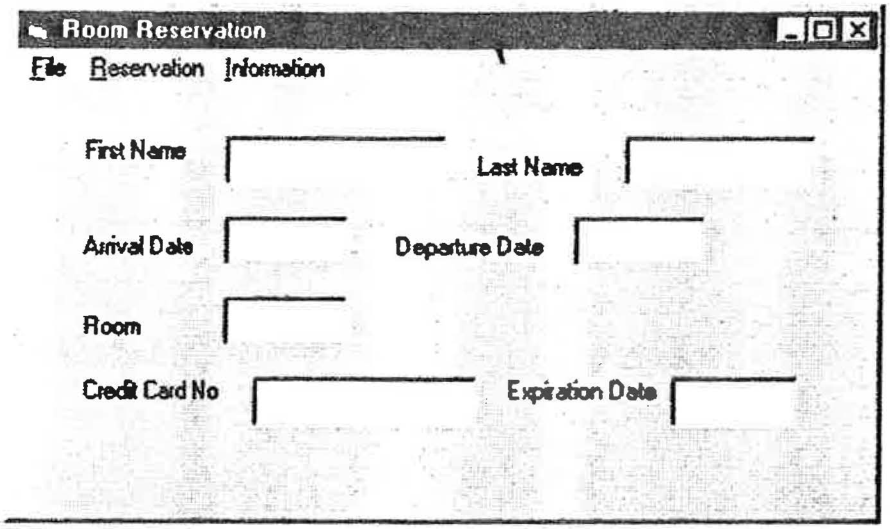
No unused procedures included ______

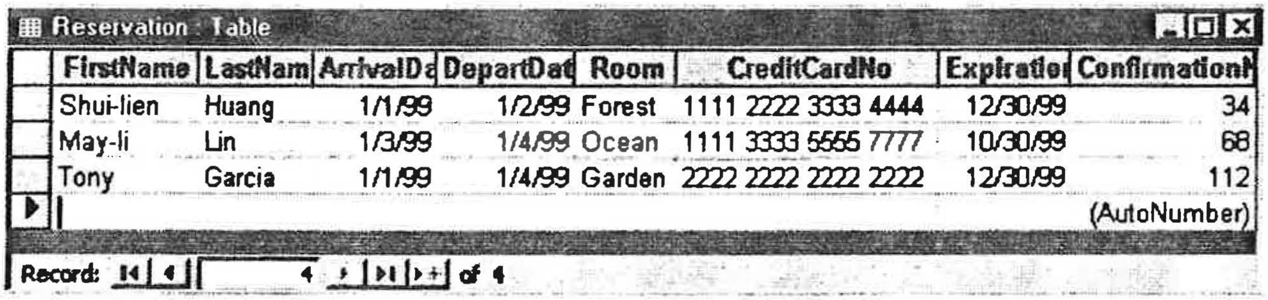
Timeliness: -2

Extra credit: ______

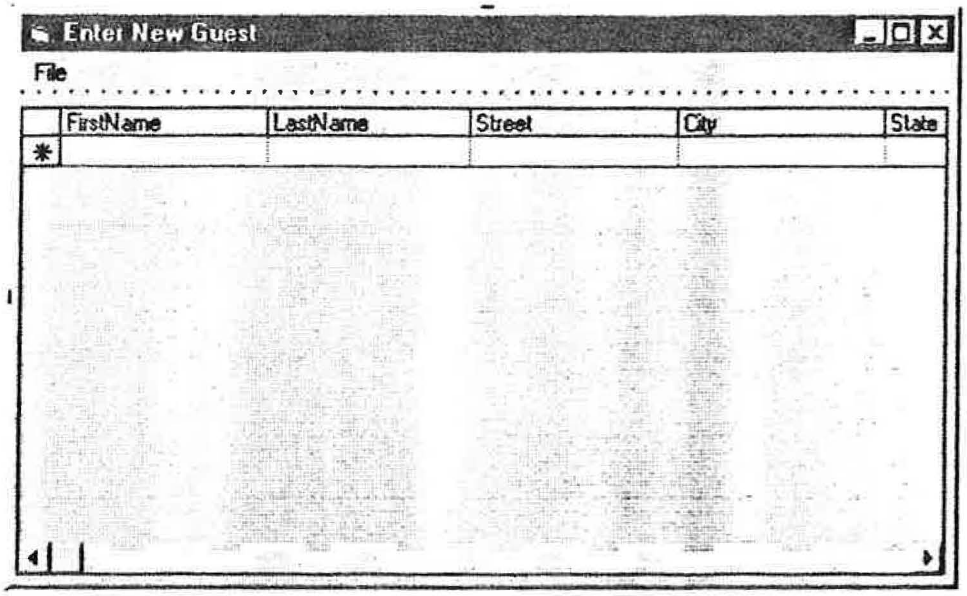
Project Score: 98

116





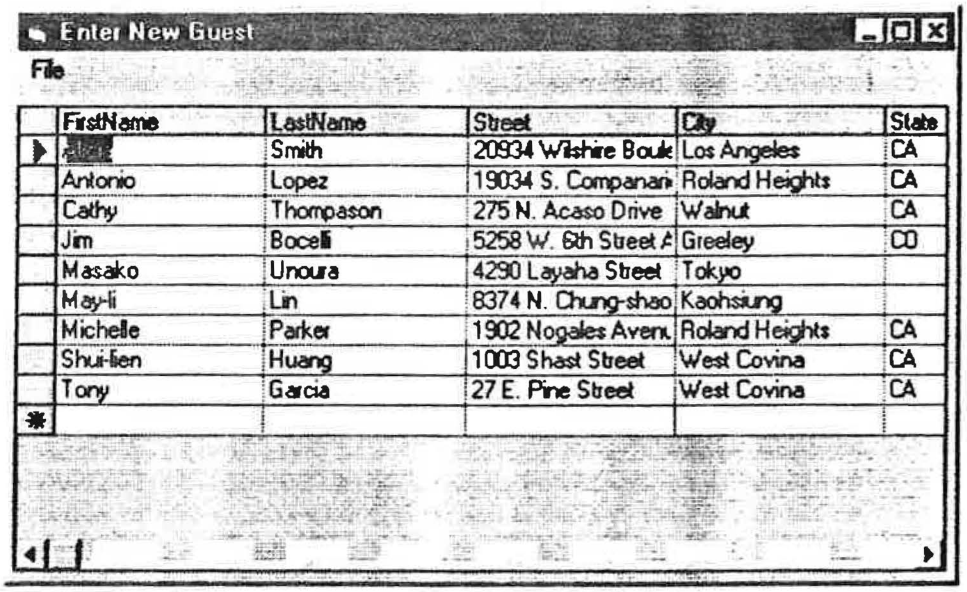
FirstName	LastNam	ArrivalDa	DepartDat	Room	CreditCardNo	Expiratio	ConfirmationN
Shui-lien	Huang	1/1/99	1/2/99	Forest	1111 2222 3333 4444	12/30/99	34
May-li	Lin	1/3/99	1/4/99	Ocean	1111 3333 5555 7777	10/30/99	68
Tony	Garcia	1/1/99	1/4/99	Garden	2222 2222 2222 2222	12/30/99	112
							(AutoNumber)



Enter New Guest

File

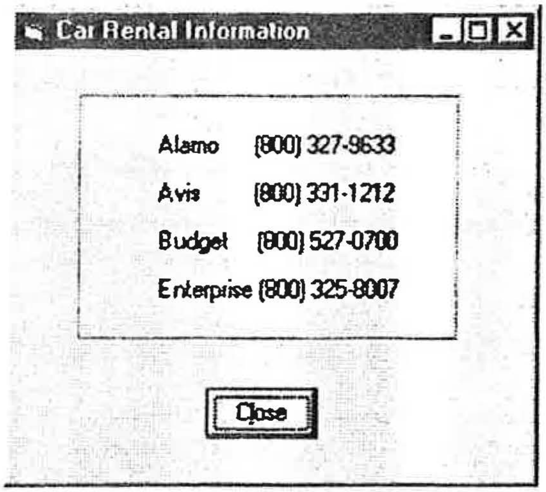
	FirstName	LastName	Street	City	State
*					



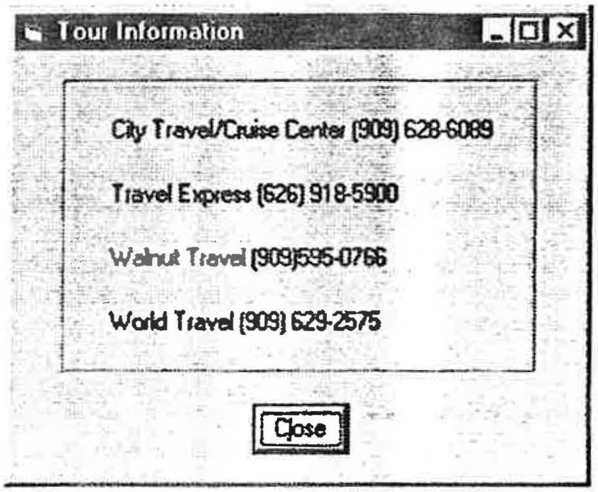
Enter New Guest

File

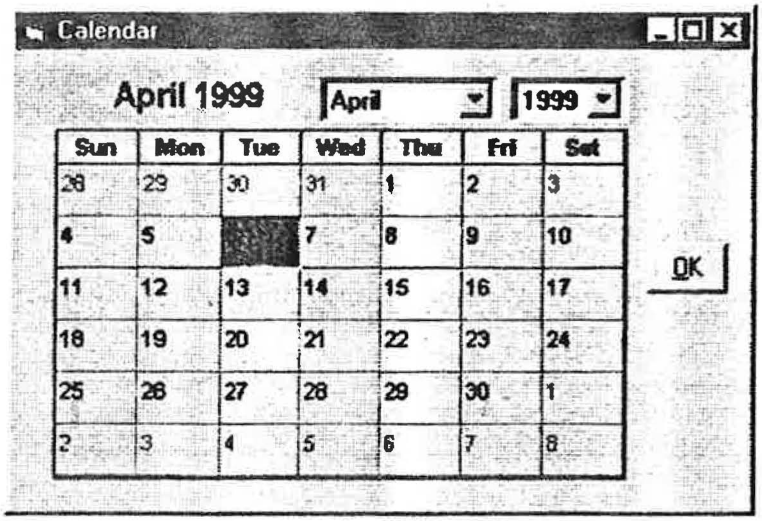
	FirstName	LastName	Street	City	State
▶	[illegible]	Smith	20934 Wilshire Boule	Los Angeles	CA
	Antonio	Lopez	19034 S. Companari	Roland Heights	CA
	Cathy	Thompason	275 N. Acaso Drive	Walnut	CA
	Jim	Bocelli	5258 W. 6th Street A	Greeley	CO
	Masako	Unoura	4290 Layaha Street	Tokyo	
	May-li	Lin	8374 N. Chung-shao	Kaohsiung	
	Michelle	Parker	1902 Nogales Avenu	Roland Heights	CA
	Shui-lien	Huang	1003 Shast Street	West Covina	CA
	Tony	Garcia	27 E. Pine Street	West Covina	CA
*					



Alamo	(800) 327-9633
Avis	(800) 331-1212
Budget	(800) 527-0700
Enterprise	(800) 325-8007



City Travel/Cruise Center	(909) 628-6089
Travel Express	(626) 918-5900
Walnut Travel	(909)595-0766
World Travel	(909) 629-2575



Sun	Mon	Tue	Wed	Thu	Fri	Sat
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	1
2	3	4	5	6	7	8

```
'Name:          Shui-lien Huang
'Date:          April 5, 1999
'Folder:        Project5
'File Name:     BnBReservation
'Description:   The form lets the user to make room reservation.
'               If this is a new guest, information must be first
'               entered in the guest table.  This module checks room
'               availability based on the arrival date.  Then the
'               information is saved to the reservation table.

Option Explicit
Dim mblnSameDate As Boolean

Private Sub Form_Load()
    'Center the current form

    Me.Top = (Screen.Height - Me.Height) / 2
    Me.Left = (Screen.Width - Me.Width) / 2
End Sub

Private Sub mnuFileClose_Click()
    'Close the current form

    Unload Me
End Sub

Private Sub mnuInfoCalendar_Click()
    'Display calendar for reference

    frmBnBCalendar.Show
End Sub

Private Sub mnuInfoCarRental_Click()
    'Show the CarRental form for reference

    frmBnBCarRental.Show
End Sub

Private Sub mnuInfoTour_Click()
    'Show the TourInfo form for reference

    frmTourInfo.Show
End Sub

Private Sub mnuRecordSave_Click()
    'Save the current record to the reservation table
    Dim strSQL As String

    strSQL = "Select * From Reservation"    'Get the table, otherwise
    datReservation.RecordSource = strSQL    'the subset is used        ✓
    datReservation.Recordset.AddNew
    TransferData                            'Transfer data from text boxes
    On Error GoTo HandleSaveErrors          'to the reservation table
    datReservation.Recordset.Update

    Exit Sub
```

```
HandleSaveErrors:
    Dim strMsg As String

    Select Case Err.Number
        Case 3022
            strMsg = "Duplicate name not saved"
            MsgBox strMsg, vbExclamation, "Database Error"
            SetupAddRecord                    'Give another chance to add
            On Error GoTo 0
        Case 3058, 3315
            strMsg = "Name must be completed before saved"
            MsgBox strMsg, vbExclamation, "Database Error"
            SetupAddRecord
            On Error GoTo 0
        Case Else
            strMsg = "Record could not be saved" & vbCr & Err.Description
            MsgBox strMsg, vbExclamation, "Database Error"
            datReservation.Recordset.CancelUpdate
            Resume Next
    End Select
End Sub

Private Sub TransferData()
    'Transfer data from controls to table fields

    With datReservation
        .Recordset!FirstName = txtFirstName
        .Recordset!LastName = txtLastName
        .Recordset!ArrivalDate = txtArrivalDate
        .Recordset!DepartDate = txtDepartDate
        .Recordset!Room = txtRoom
        .Recordset!CreditCardNo = txtCreditCardNo
        .Recordset!ExpirationDate = txtExpirationDate
    End With
End Sub

Private Sub SetupAddRecord()
    'Set up a new Add to allow the user another try
    Dim strLastName As String
    Dim strFirstName As String
    Dim strArrivalDate As String
    Dim strDepartDate As String
    Dim strRoom As String
    Dim strCreditCardNo As String
    Dim strExpirationDate As String

    On Error Resume Next

    'Save the contents of the form controls
    strLastName = txtLastName
    strFirstName = txtFirstName
    strArrivalDate = txtArrivalDate
    strDepartDate = txtDepartDate
    strRoom = txtRoom
    strCreditCardNo = txtCreditCardNo
    strExpirationDate = txtExpirationDate
    'Start a new Add
    datReservation.Recordset.AddNew

    'Place saved data back in form controls
    With txtLastName
        .Text = strLastName
        .SelStart = 0
        .SelLength = Len(.Text)
        .SetFocus
    End With
    txtFirstName = strFirstName
    txtArrivalDate = strArrivalDate
    txtDepartDate = strDepartDate
    txtRoom = strRoom
    txtCreditCardNo = strCreditCardNo
    txtExpirationDate = strExpirationDate
End Sub

Private Sub txtArrivalDate_Validate(Cancel As Boolean)
    'Check to see if the date is available
    Dim dtmDate As Date
    Dim strSQL  As String

    dtmDate = CDate(txtArrivalDate)
    strSQL = "Select * From Reservation " & _
             "where ArrivalDate = #" & dtmDate & "#"        'search string

    With datReservation
        .RecordSource = strSQL
        .Refresh                                            'Reopen the database
        If Not (.Recordset.BOF And .Recordset.EOF) Then     'Match
            mblnSameDate = True
        End If
    End With
End Sub

Private Sub txtLastName_Validate(Cancel As Boolean)
    'check to see if it is a new guest
    Dim strName As String
    Dim strSQL As String

    strName = txtLastName & txtFirstName
    strSQL = "Select LastName, FirstName From Guests " & _
             "where LastName+FirstName = '" & strName & "'"

    With datGuests
        .RecordSource = strSQL
        .Refresh                                    'Reopen the database
        If .Recordset.BOF And .Recordset.EOF Then   'No match
            MsgBox "Enter the new guest in Guest table", , _
                    "Guest not Found"
            Me.Hide
            frmBnBGuest.Show
        End If
    End With
End Sub
```

```
Private Sub txtRoom_Validate(Cancel As Boolean)
    'Check to see if the room is available
    Dim strSQL As String

    If mblnSameDate Then

        With datReservation
            strSQL = "Select * From Reservation Where Room = '" & txtRoom & "'"
            .RecordSource = strSQL
            .Refresh
            If Not (.Recordset.BOF And .Recordset.EOF) Then
                MsgBox "This room is not available", , "Room Reservation"
                Cancel = True
                With txtRoom
                    .SelStart = 0
                    .SelLength = Len(txtRoom)
                End With
            End If
        End With
    End If
End Sub
```

✓

```
'Name:          Shui-lien Huang
'Date:          April 5, 1999
'Folder:        Project5
'File Name:     BnBGuest
'Description:   The form lets the user to enter a new guest.

Option Explicit

Private Sub Form_Load()
    'Center the form

    Me.Top = (Screen.Height - Me.Height) / 2
    Me.Left = (Screen.Width - Me.Width) / 2
End Sub

Private Sub mnuFileClose_Click()
    'Return to the reservation form

    Me.Hide
    frmBnBReservation.Show
End Sub
```

BnBCalendar.frm

```
Option Explicit

Private Sub cmdOK_Click()
        'Hide the Calendar form

    Me.Hide
End Sub

Private Sub Form_Load()
    'Center the current form

    Me.Top = (Screen.Height - Me.Height) / 2
    Me.Left = (Screen.Width - Me.Width) / 2
End Sub
```

BnBCarRental.frm

```
Option Explicit

Private Sub cmdClose_Click()
    'Hide the CarRental form

    Me.Hide
End Sub

Private Sub Form_Load()
    'Center the current form

    Me.Top = (Screen.Height - Me.Height) / 2
    Me.Left = (Screen.Width - Me.Width) / 2
End Sub
```

BnBTourInfo.frm

```
Option Explicit

Private Sub cmdClose_Click()
    'Hide the Tour form

    Me.Hide
End Sub

Private Sub Form_Load()
    'Center the current form

    Me.Top = (Screen.Height - Me.Height) / 2
    Me.Left = (Screen.Width - Me.Width) / 2
End Sub
```

Visual Basic

Project Evaluation Sheet

Programmer Name: Shui-lien Huang Project Number: 6

Disk Folder Name: project 6 Project (.vbp) file: vbg Guest

Date Due: 4/20/99 Date Turned In: 4/19/99

Above to be completed by student

Points (100 Possible)

Project Performance:

Project works according to specifications ____

Output is accurate ____

Logic is efficient ____

Meets all requirements ____

Disk Storage:

Folder is present and contains all necessary project files (no extra files) ____

Folder name and file names match those listed above ____

Project runs as submitted ____

User Interface:

Professional appearance ____

Layout, placement, spelling ____

Follows Windows standards ____

Keyboard access keys (& in Caption, Default and Cancel Set)

Tab order is correct

External documentation:

Professional presentation ____

Folder: outside label, evaluation sheet, divider tabs, and disk securely attached ____

Printout of all forms, form text, and code ____

Standards and conventions:

Object names ____

Identifiers for Variables and Named Constants ____

Option Explicit / All variables declared ____

Remarks in General Declarations section ____

Programmer name

Project number

Date

Project description

Descriptive remarks in every procedure ____

Proper indentation ____

Proper blank lines ____

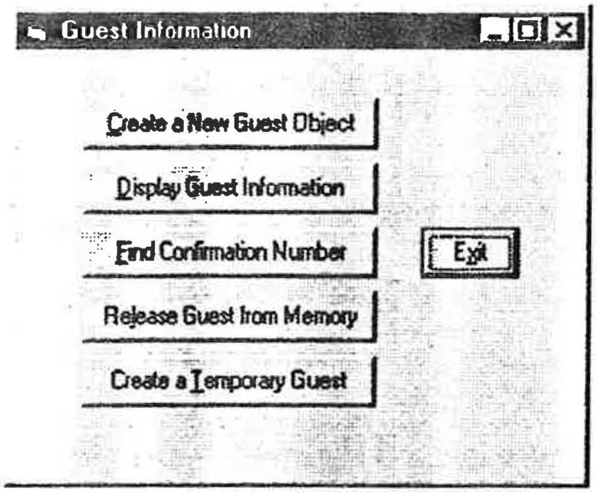
No unused procedures included ____

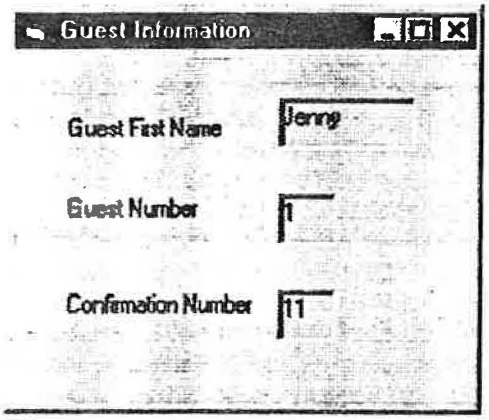
Timeliness: ____

Extra credit: ____

Project Score: 100

126


Guest Information
Create a New Guest Object
Display Guest Information
Find Confirmation Number
Release Guest from Memory
Create a Temporary Guest
Exit



Guest First Name Jenny

Guest Number 1

Confirmation Number 11

```
'Name:          Shui-lien Huang
'Date:          April 14, 1999
'Folder:        Project6
'File Name:     CGuest.cls
'Description:   Declare property for the CGuess class

Option Explicit
Public FirstName           As String
Private mintGuestNumber    As Integer

Public Property Get GuestNumber() As Long
    'Read the Guest number

    GuestNumber = mintGuestNumber
End Property

Private Sub Class_Initialize()
    'Called when object is instantiated

    mintGuestNumber = getGuestNumber

    'Show what's happening
    Debug.Print "Confirm # " & GuestNumber & " Instantiated"
End Sub

Private Sub Class_Terminate()
    'Called when object is destroyed

     Debug.Print "Confirm # " & GuestNumber & " Guest: " _
        & FirstName & " Destroyed"
End Sub

Public Function ConfirmNumber() As Long
    'Generate a confirmation number using hashing algorithm
    Dim intASCIIsum As Integer
    Dim intIndex    As Integer

    For intIndex = 1 To Len(FirstName)
        intASCIIsum = intASCIIsum + Asc(Mid(FirstName, intIndex, 1))
    Next intIndex

    ConfirmNumber = intASCIIsum Mod 101
End Function
```

```
'Name:         Shui-lien Huang
'Date:         April 14, 1999
'Folder:       Project6
'File Name:    CGuestDialog.cls
'Description:  Add a dialog box to the DLL component

Option Explicit
Private WithEvents mdlgGuest As frmShowGuest
Attribute mdlgGuest.VB_VarHelpID = -1
Event ShowGuestInfo(ByVal strFirstName As String, _
                    ByVal intGuestNumber As Integer, _
                    ByVal lngConfirmNumber As Long)

Private Sub Class_Initialize()
    'Create a dialog object

    Set mdlgGuest = New frmShowGuest
End Sub

Private Sub Class_Terminate()
    'Unload the dialog box

    Unload mdlgGuest
    Set mdlgGuest = Nothing
End Sub

Private Sub mdlgGuest_ShowGuestInfo(ByVal strFirstName As String, _
                    ByVal intGuestNumber As Integer, _
                    ByVal lngConfirmNumber As Long)
    'Call the event

    RaiseEvent ShowGuestInfo
End Sub

Public Function ShowGuestInfo(ByVal strFirstName As String, _
                    ByVal intGuestNumber As Integer, _
                    ByVal lngConfirmNumber As Long)
    'Display the Dialog box

    With mdlgGuest
        .lblFirstName = strFirstName
        .lblGuestNumber = intGuestNumber
        .lblConfirmNumber = lngConfirmNumber
        .Show
    End With
End Function
```

```
'Name:          Shui-lien Huang
'Date:          April 14, 1999
'Folder:        Project6
'File Name:     Guest.bas
'Description:   Confirm Standard Code Module

Option Explicit
Public gdtmServerStarted As Date

Sub Main()
    'Start component execution

    gdtmServerStarted = Now
    Debug.Print "Component executing"
End Sub

Public Function getGuestNumber() As Integer
    'Generate a Guest number
    Static intGuestNumber As Integer

    intGuestNumber = intGuestNumber + 1
    getGuestNumber = intGuestNumber         'return value
End Function
```

```
'Name:          Shui-lien Huang
'Date:          April 14, 1999
'Folder:        Project6
'File Name:     Guest.frm
'Description:   The project main user interface that allows the
'               user to create, display, release object.

Option Explicit
Private mgueGuest As CGuest
Private WithEvents mdlgGuest  As CguestDialog
Attribute mdlgGuest.VB_VarHelpID = -1

Private Sub cmdCreate_Click()
    'Instantiate a new guest object

    Set mgueGuest = New CGuest
    mgueGuest.FirstName = InputBox("Enter First Name")
    With mgueGuest
        Set mdlgGuest = New CguestDialog
        mdlgGuest.ShowGuestInfo .FirstName, .GuestNumber, .ConfirmNumber
    End With
End Sub

Private Sub cmdExit_Click()
    'Exit the project

    End
End Sub

Private Sub cmdConfirmNumber_Click()
    'Call the GuestNumber method

    MsgBox "Confirmation Number: " & mgueGuest.ConfirmNumber, _
        vbOKOnly, "Guest"
End Sub

Private Sub cmdDisplay_Click()
    'Display the Guest in a message box

    MsgBox ("First Name: " & mgueGuest.FirstName & vbCrLf & _
        "Guest Number: " & mgueGuest.GuestNumber)
End Sub

Private Sub cmdRelease_Click()
    'Remove the object

    Set mgueGuest = Nothing
End Sub

Private Sub cmdTemp_Click()
    'Create a local object
    Dim gueTemporary As New CGuest

    gueTemporary.FirstName = InputBox("Enter First Name ", "Local Object")
End Sub
```

```
Private Sub Form_Load()
    'Center the form

    Me.Top = (Screen.Height - Me.Height) / 2
    Me.Left = (Screen.Width - Me.Width) / 2
End Sub
```

```
'Name:         Shui-lien Huang
'Date:         April 14, 1999
'Folder:       Project6
'File Name:    ShowGuest.frm
'Description:  A dialog box contains an event for the Guest component

Option Explicit
Event ShowGuestInfo(ByVal strFirstName As String, _
                    ByVal intGuestNumber As Integer, _
                    ByVal lngConfirmNumber As Long)

Private Sub Form_QueryUnload(Cancel As Integer, UnloadMode As Integer)
    'Prevent the Close button from unloading the form

    If UnloadMode = vbFormControlMenu Then
        Cancel = True                       'Don't close
        Me.Visible = False                  'Hide dialog
    End If
End Sub

Private Sub Form_Load()
    'Center the form

    Me.Top = (Screen.Height - Me.Height) / 2
    Me.Left = (Screen.Width - Me.Width) / 2
End Sub
```

Visual Basic

Project Evaluation Sheet

Programmer Name: Shui-lien Huang Project Number: 7

Disk Folder Name: Project 7 Project (.vbp) file: VeryBeads

Date Due: 4/27/99 Date Turned In: 4/19/99

Above to be completed by student

Points (100 Possible)

Project Performance:

Project works according to specifications ______

Output is accurate ______

Logic is efficient ______

Meets all requirements ______

Disk Storage:

Folder is present and contains all necessary project files (no extra files) ______

Folder name and file names match those listed above ______

Project runs as submitted ______

User Interface:

Professional appearance ______

Layout, placement, spelling ______

Follows Windows standards ______

Keyboard access keys (& in Caption, Default and Cancel Set)

Tab order is correct

External documentation:

Professional presentation ______

Folder: outside label, evaluation sheet, divider tabs, and disk securely attached ______

Printout of all forms, form text, and code ______

Standards and conventions:

Object names ______

Identifiers for Variables and Named Constants ______

Option Explicit / All variables declared ______

Remarks in General Declarations section ______

Programmer name

Project number

Date

Project description

Descriptive remarks in every procedure ______

Proper indentation ______

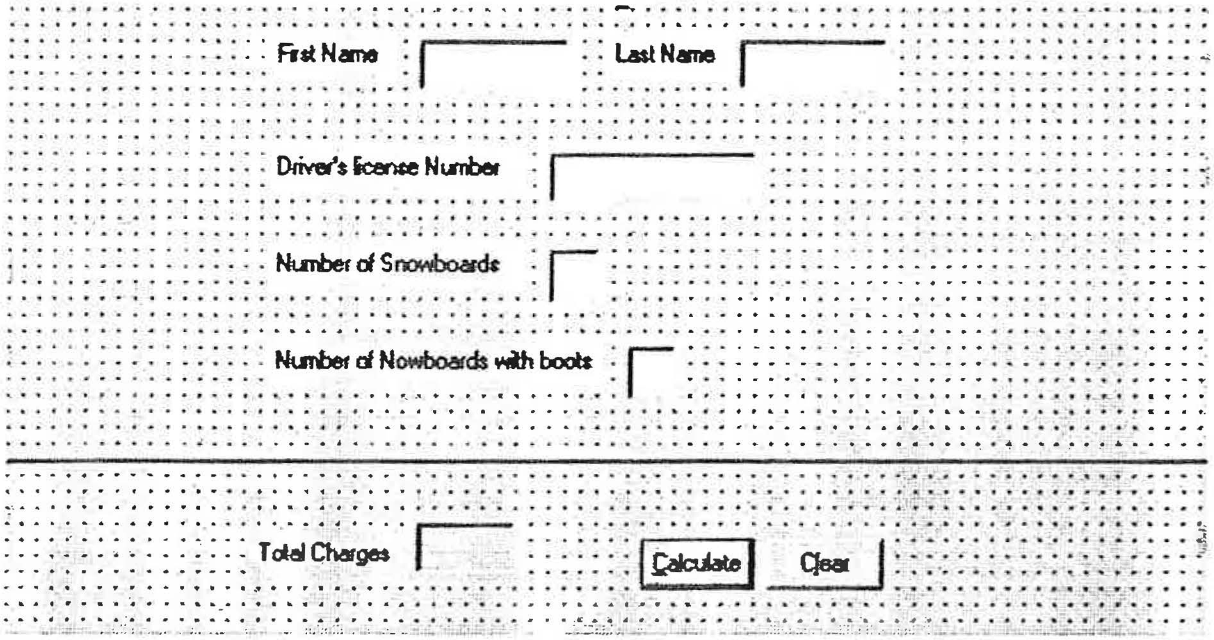
Proper blank lines ______

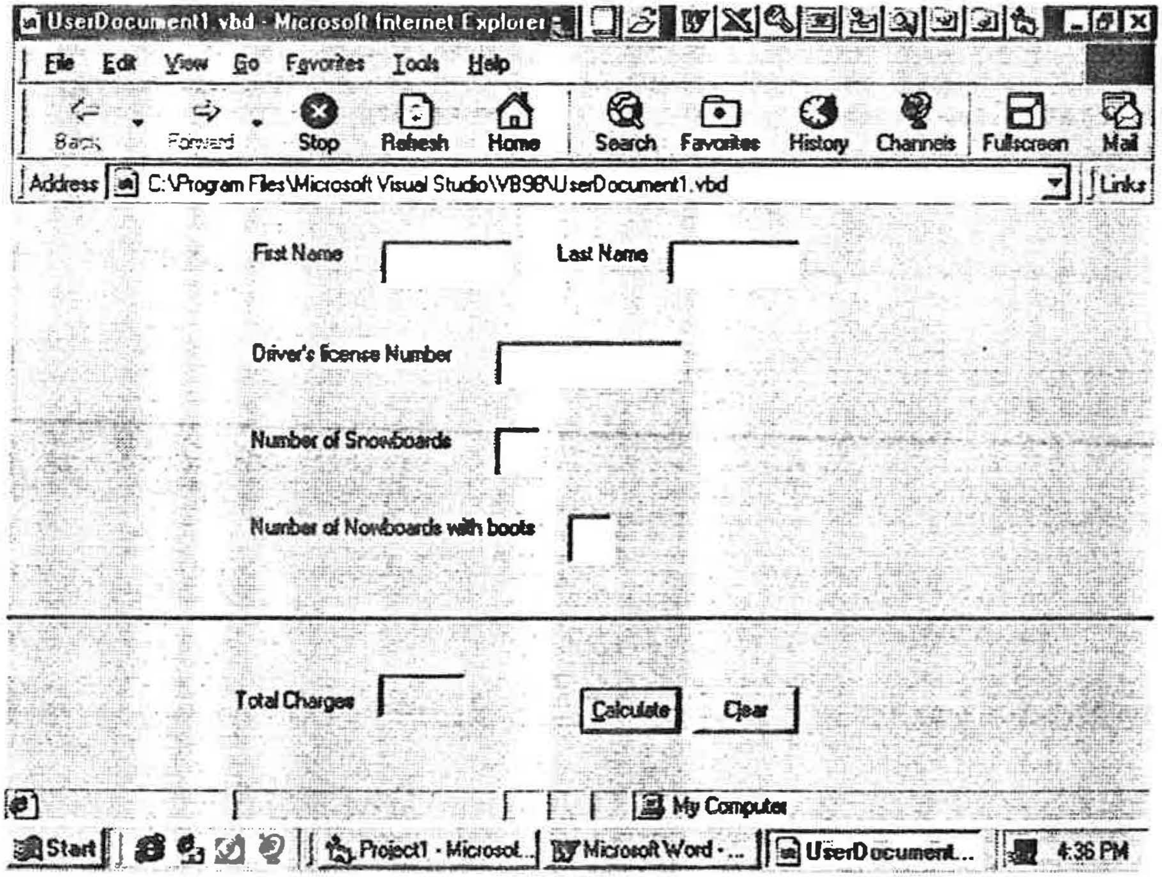
No unused procedures included ______

Timeliness: ______

Extra credit: ______

Project Score: 110





Address: C:\Program Files\Microsoft Visual Studio\VB98\UserDocument1.vbd

```
'Name:          Shui-lien Huang
'Date:          April 17, 1999
'Folder:        Project7
'File Name:     VeryBoards.dob
'Description:   The project creates an ActiveX document that will
'               contains one UserDocument to calculate snowboard
'               rental charges in Internet Explorer browser.

Option Explicit

Private Sub cmdCalc_Click()
    'Calculate the charges
    Const curNO_BOOTS   As Currency = 20
    Const curWITH_BOOTS As Currency = 30


    lblCharges = (txtNoOfBoards * curNO_BOOTS) + _
                  (txtNoOfBoardsBoots * curWITH_BOOTS)
    lblCharges = Format$(lblCharges, "currency")
End Sub

Private Sub cmdClear_Click()
    'Clear the text boxes

    txtFirstName = ""
    txtLastName = ""
    txtLicenseNo = ""
    txtNoOfBoards = ""
    txtNoOfBoardsBoots = ""
    txtFirstName.SetFocus
End Sub
```

✓

Visual Basic

Project Evaluation Sheet

Programmer Name: Shui-lien Huang Project Number: 8

Disk Folder Name: Project 8 Project (.vbp) file: Club

Date Due: 5/11/99 Date Turned In: 4/27/99

Above to be completed by student

Points (100 Possible)

Project Performance:

Project works according to specifications

Output is accurate

Logic is efficient

Meets all requirements

Disk Storage:

Folder is present and contains all necessary project files (no extra files)

Folder name and file names match those listed above

Project runs as submitted

User Interface:

Professional appearance

Layout, placement, spelling

Follows Windows standards

Keyboard access keys (& in Caption, Default and Cancel Set)

Tab order is correct

External documentation:

Professional presentation

Folder: outside label, evaluation sheet, divider tabs, and disk securely attached

Printout of all forms, form text, and code

Standards and conventions:

Object names

Identifiers for Variables and Named Constants

Option Explicit / All variables declared

Remarks in General Declarations section

Programmer name

Project number

Date

Project description

Descriptive remarks in every procedure

Proper indentation

Proper blank lines

No unused procedures included

Timeliness:

Extra credit: +10

Nice help file

Project Score: 110

```
'Name:          Shui-lien Huang
'Date:          April 27, 1999
'Folder:        Project8
'File Name:     Club.frm
'Description:   The project displays club data from a remote AS400
'               COBOL data file using ODBC connection.  The project
'               includes a help menu item that displays the user-defined
'               help file.

Option Explicit

Private Sub Form_Load()
    'Center the form

    Me.Top = (Screen.Height - Me.Height) / 2
    Me.Left = (Screen.Width - Me.Width) / 2
End Sub

Private Sub lblClassLevel_Change()
    'Display Freshman for code F and Sophomore for code S

    If lblClassLevel = "F" Then
        lblClassLevel = "FRESHMAN"
    ElseIf lblClassLevel = "S" Then
        lblClassLevel = "SOPHOMORE"
    End If
End Sub

Private Sub mnuFileExit_Click()
    'End the project

    End
End Sub

Private Sub mnuHelpHow_Click()
    'Display the help file

    dlgCommon.HelpFile = App.Path & "\ClubHelp.hlp"
    dlgCommon.HelpCommand = cdlHelpContents
    dlgCommon.ShowHelp
End Sub
```

Project 9

MSFlexGrid Control

The Microsoft FlexGrid control displays and operates on tabular data. It allows complete flexibility to sort, merge, and format tables containing strings and pictures. The MSFlexGrid control permits you to create highly effective decision-support applications. It displays data in a grid that makes it easy for user to group information in order to view and summarize data more easily.

A. Unbound mode

When the MSFlexGrid is not bound to a data control, you can add data to the control using its properties and methods. You start populating an unbound FlexGrid by setting the grid's Rows and Columns properties.

B. Bound mode

You can connect the FlexGrid control to a database table through a Data control. When you do this, the data displayed by the grid is determined by the Data control. Data displayed in the FlexGrid is Read Only.

Merge Example

1. Add a Data control to the form.
   Properties: Name - datPurchase
   DatabaseName - Purchase.mdb
   RecordSource - Purchase
   Visible - False

2. Add a FlexGrid control to the form.
   Properties: DataSource - datPurchase
   FixedCols - 0

3. Write code.

```
Private Sub Form_Load()
   'Enable merging for the first two columns

   MSFlexGrid1.MergeCol(0) = True
   MSFlexGrid1.MergeCol(1) = True
End Sub

Private Sub MSFlexGrid1_Click()
   'Allow the user to determine which column is merged

   MSFlexGrid1.ColPosition(MSFlexGrid1.MouseCol) = 0
   MSFlexGrid1.Col = 1
   MSFlexGrid1.ColSel = 0
   MSFlexGrid1.Sort = flexSortStringAscending
End Sub
```

C. MSFlexGrid Properties

1. MergeCol Properties

Returns or sets a value that determines which columns can have their contents merged.

object.MergeCol(*number*) [=*Boolean*]

2. ColPosition Properties

Sets the position of a MSFlexGrid column, allowing you to move columns to specific positions.

object.ColPosition(*number*) [= *value*]

3. MouseCol Properties

Returns the current mouse position, in column coordinates.

object.MouseCol [=*value*]

4. Cols Properties

Returns or sets the total number of columns.

object.Cols [= *value*]

5. ColSel Properties

Returns or sets the start column for a range of cells.

object.ColSel [= *value*]

To select a block of cells from code, you must first set the Col property, and then set ColSel.

6. Sort Property

Sets a value that sorts selected rows according to selected criteria.

object.Sort [=*value*]

Constant	Description
flexSortStringAscending	String Ascending. An ascending sort using case-sensitive string comparison is performed.

Appendix D

Transcripts

MT. SAN ANTONIO COLLEGE

WALNUT, CALIFORNIA

ACADEMIC TRANSCRIPT

HUANG SHUI-LIEN 459-47-5834 02/08/1951 ENTERED: SPRING 1991

COURSE	COURSE TITLE	UN.A.	UN.C.	GRADE	GPTS	CSU	LEVEL
	* * * * SUMMER 1994 * * *						
PE-A 8A	SWIMMING - BEGINNING		1.00	CR			BL
	SEMESTER TOTAL		1.00			GPA=	
	* * * * SUMMER 1998 * * *						
ART 30A	CERAMICS--BEGINNING CAN:ART 06	3.00	3.00	A	12.00	C1	BL
	SEMESTER TOTAL	3.00	3.00		12.00	GPA=	4.00
	* * * * * FALL 1998 * * *						
ART 30B	CERAMICS--BEGINNING	3.00	3.00	A	12.00		BL
BUSO 5	BUSINESS ENGLISH	3.00	3.00	A	12.00		AA/AS
BUSR 50	REAL ESTATE PRINCIPLES			W			BL
	SEMESTER TOTAL	6.00	6.00		24.00	GPA=	4.00
	* * * * SPRING 1999 * * *						
ART 31A	CERAMICS--INTERMEDIATE	3.00	3.00	A	12.00		BL
CIS 38	ADV BASIC PROGRAMMING	4.00	4.00	A	16.00		BL
CIS 33	ADV DATABASE MGMT-MICROS	4.00	4.00	A	16.00		AA/AS
	SEMESTER TOTAL	11.00	11.00		44.00	GPA=	4.00

ACADEMIC SUMMARY	UN.A.	UN.C.	GPTS	
MSAC NON-DEGREE TOTAL		1.50		GPA=
GRADE POINT BALANCE				
MSAC DEGREE TOTAL	56.00	57.00	224.00	GPA=4.00
GRADE POINT BALANCE			112.00	
MSAC CUMULATIVE TOTAL	56.00	58.50	224.00	GPA=4.00
GRADE POINT BALANCE			112.00	
ALL COLLEGE TOTAL	56.00	58.50	224.00	GPA=4.00
GRADE POINT BALANCE			112.00	

CONTINUED ON PAGE 3

Lynn 142 Hanks

DIRECTOR. ADMIS