**BANGALOREUNIVERSITY**

**A**

**PROJECT REPORT ON**

**“AIRLINE RESERVATIONS”**

A 6TH SEM project report Submitted in partial fulfillment of the requirements of Bangalore University for the award of the degree of

**BACHELOR OF SCIENCE (B.SC)**

**Submitted by:**

**GUNASHREE.V.N** **SINDHU SHREE.N**

(15WXS85049) (15WXS85067)



SMT.DANAMMA CHANNABASAVAIAH COLLEGE OF ARTS, COMMERCE, SCIENCE & MANAGEMENT STUDIES

NH-75 KODIRAMASANDRAKOLAR – 563101

(NAAC Accredited with ‘B’ Grade & Affiliated to Bangalore University)

**CERTIFICATE**

SMT.DANAMMA CHANNABASAVAIAH COLLEGE OF ARTS COMMERCE , SCIENCE & MANAGEMENT STUDIES

KOLAR – 563101



**Certified that the project work entitled**

**“AIRLINE RESERVATIONS”**

Project Work jointly carried out by Gunashree.V.N & Sindhu shree.N of 6th semester BSc. This Report is Submitted in Partial fulfillment of requirements for the award of the Degree in Bachelor of Science by Bangalore University during the Academic Year 2017-2018

Signature of the Guide Signature of the H.O.D

**Mr.Satyajit Ray** **Mr.Satyajit Ray**

Name of the Examiner Signature with date:

1)………………………..

2)…………………………

**DECLARATION**

The project titled “AIRLINE RESERVATIONS” developed by us in partial fulfillment of B.Sc sixth semester, Bangalore university is an authentic work carried out by us under the guidance of “MR.SATYAJIT RAY” ,Head of the department of computer science .

I declared that the project has not been submitted to any degree or diploma to this university or any university.

Place: Kolar

Date:…………

**GUNASHREE.V.N** (15WXS85049)

**SINDHU SHREE.N** (15WXS85067)

Yes I have gone through this project and verified, and checked as well

…………………………..

(Project guide)

***ACKNOWLEDGEMENT***

**We thank almighty God for giving us the patience to satisfactorily complete our project in the provided time.**

We take this opportunity to express our deep sense of gratitude to our Founder Chairman Honurable **SMT.USHA GANGADHAR madam, SMT. PUSHPALATHA madam**, Principal of her valuable guidance, keen interest and helpful During the course of Study.

We would like to express our sincere thanks to **Mr.SATYAJIT** **RAY**, head of the department of computer science for his guidance, encouragement and support.

We are grateful to our parents who are supported and provided us very facility and a good working environment at our home and finally our special thanks to our friends and throughout our endeavors.

**GUNASHREE.V.N**

**SINDHU SHREE.N**

**PREFACE**

* **The main goal of the project**, **”AIRLINE RESERVATIONS” is to acquaint user with partial fulfillment of requirements and data collection and presentation.**
* **The project is designed according to the data collection of Apartment Management System can be managed with computerization system. The project needs some more upgradation to fulfill the requirements.**

**With Regards,**

**GUNASHREE.V.N**

**SINDHU SHREE.N**

**INDEX**

|  |  |  |
| --- | --- | --- |
| **SL.NO** | **CONTENTS** | **PAGE.NO** |
| **1.** | **PROJECT SYNOPSIS** | **1** |
| **2.** | **SOFTWARE DEVELOPMENT LIFE CYCLE(SDLC)** | **2** |
| **3.** | **SOFTWARE REQUIREMENT SPECIFICATION** | **8** |
| **4.** | **SOFTWARE TOOLS AND CONNECTIVITY** | **10** |
| **5.** | **DATAFLOW DIAGRAM** | **31** |
| **6.** | **ER DIAGRAMS** | **33** |
| **7.** | **CODING** | **36** |
| **8.** | **SNAPSHOTS** | **44** |
| **9.** | **DATABASE TABLES** | **47** |
| **10.** | **TESTING** | **48** |
| **11.** | **CONCLUSION** | **52** |
| **12.** | **BIBILOGRAPHY** | **52** |

**SYNOPSIS**

**PROJECT TITLE : AIRLINE RESERVATIONS**

**SUBMITTING BY: Gunashree.V.N, Sindhu shree.N**

**SOFTWARE REQUIREMENTS:**

**OPERATING SYSTEM** :Windows 7 Home Premium

**FRONT END** : Visual Basic 6.0

**BACK END** : Microsoft Office Access 2007

**HARDWARE REQUIREMENTS:**

**PROCESSOR**: INTEL I3

**MONITOR** : Any display unit

**HARD DISK** : 200GB and above

**RAM** : 2.00 GB

DESCRIPTION:

The software has been developed using visual basic 6.0 as front and Microsoft office access 2007 as back end. The Operating system used is windows 7. One can work with different files at the same time which saves spaces and handling time during the process of searching and processing of data.

The project “AIRLINE RESERVATIONS” contains 8 modules.

They are as follows:

1. Login

2. Flight

3. Search

4. Fare

5. Reservation

This project is helpful for whoever with any basic knowledge of computers can navigate through this project.

**Software development life cycle (SDLC)**

**Modeling:**

During the evaluation and solution synthesis activity, the analyst creates models of the system in an effort to better understand data and control flow. The model serves as a foundation for the software design and the basis for the creation of specification for the software. For the better understanding of data and control flow we use Data Flow Diagram.

**THE PRINCIPLE STAGES OF SDLC ARE:**

* **Requirement Analysis and Definition:**

After feasibility study has been performed, the requirements analysis which includes the software system’s services, constraints and goals is established after consulting the system users. Then the exact requirements are defined in detail.

* **System and software Design:**

Once the requirements for a system have been documented the software system must be designed to meet them. The system design process divides the requirements into either hardware or software systems along with the overall system architecture. Software design involves designing abstraction and their relationships.

* **Implementation and Unit Testing:**

This process produces the actual code as a set of programs. Testing is an integral and important phase of software development process. Unit testing ensures that defects are recognized quickly and that each unit meets its specification.

* **Integration and system testing:**

All the modules or individual programs that have been developed and tested individually are integrated and tested as a whole system to ensure the software requirements have been meet. After system testing the software is delivered to the customer.

* **Operation and maintenance:**

This phase involves installation, customization, testing and evaluation.

Maintenance involves correcting errors which were missed in the earlier

**WATER FALL MODEL**

Requirements Definition

System and Software Design

Implementation & Unit Testing

Integration and System Testing

Operation & Maintenance

The simplest model is the water fall model that states that the phases are organized in a linear order. The software life cycle of this model consists of the following phases

1. **Requirements Analysis and Specification:**

After feasibility study has been performed to define the costs and benefits of a software system. In cases where requirements not clear e.g. for a system that has never been developed before much interaction is required between the user and developer.

1. **Design:**

Once the requirements for a system have been documented, phase is split into two sub phases:

* High level design
* Detailed design

High level design deals with the overall modules structure and organizations. The high level design is then refined by designing each module in detail.

The general principle involves making clear distinctions between what the problem is and how to solve the problem. There are many ways that the requirements may be met. Some solutions may not involve the use of computers at all.

1. **Coding and Unit Testing:**

This phase produces the actual code that will be delivered to the customer. The other phases of life cycle may also develop code, such as prototypes, tests, drives, but these are for use by the developer.

1. **Integration and System Testing:**

All the modules that have developed before and after tested individually are put together – integrated in this phase as a whole system.

1. **Delivery and Maintenance:**

Once the system phases all test, it is delivered to the customer and enters maintenance phase. Software will undergo change after it is delivered to the customer. Change will occur because errors have been encountered because the software must be adapted to accommodate changes in its new environment.

**Advantages of Waterfall model**

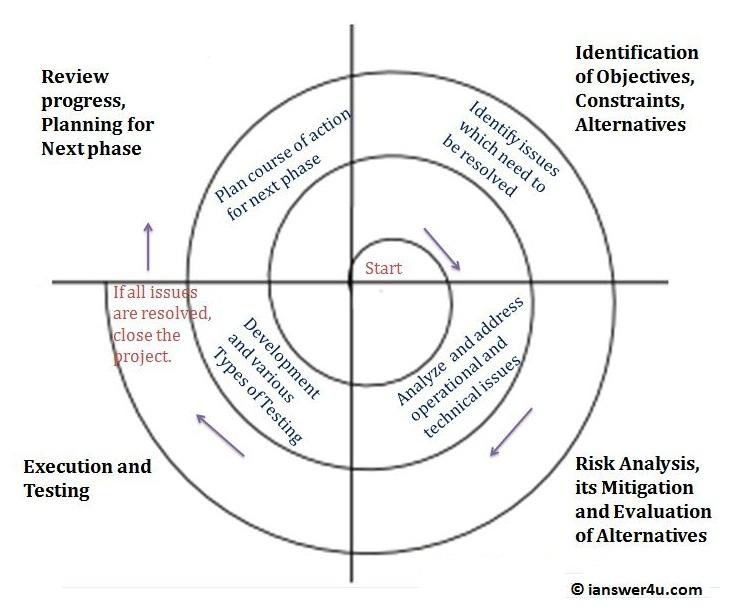
* It is easy to develop s/w through this methodthis method in a short period of time.
* The simplicity of model makes it easy to explain the development process to a client who may not familiar with s/w development.

**Disadvantages of Waterfall model**

* The Waterfall model assumes that the requirements of a system can be frozen before designs. But for new system, determining the requirements is difficult, as user does not know the requirements.
* A large project may take few years to complete. It takes a lot of effort & time, so only is not suitable for new projects.

**SPIRAL MODEL**

The spiral is similar to the increment model, with more emphasis placed on risk analysis. The spiral model has four phases planning, risk analysis, engineering and evaluation. A software project repeatedly passes through these phases in iterations. The baseline spirals, starting in the planning phase, requirements are gathered and risk is assessed. Each subsequent spiral builds on the baseline spiral. Requirements are gathered during the planning phase. In the risk analysis phase, a process is undertaken to identify risk and alternate solution. A prototype is produced at the end of the risk analysis phase.

****

The spiral model is similar to the increment model, with more emphasis placed on risk analysis.

Iterative enhancement model:

Evolutionary models are iterative. They are characterized in a manner that enables software engineers to develop increasingly more complete versions of the software.

**Advantages:**

* Reduces the risk in a system development.
* The risk is analyzed and eliminated accordingly.
* Prototype gives the option of evolution in the system.
* It reduces the chances of system failure.
* The spiral model is compact and iterative.
* The spiral model has planning strategy for easy spiral

**Disadvantages:**

* It is expensive compare to waterfall model.
* It requires more skilled persons.
* The risk assessment and analysis sometime is not accepted by the clients.

**When to use spiral model?**

* When cost and risk evaluation is important.
* For medium to high risk projects.
* Long term project commitment unwise because of potential changes to economic priorities users are ensure of their needs.
* Requirements are complex new product line.
* Significant changes are expected (research and exploration) it.
* The user interface should reinforce.

Iterative enhancement model:

Evolutionary models are iterative. They are characterized in a manner that enables software engineers to develop increasingly more complete versions of the software.

SOFTWARE REQUIREMENT SPECIFICATIONS

ABSTRACT

This document fully and formally describes the requirements of the proposed said project system. It sets out the functional requirements and includes a description of the user interface and documentation and training requirements.

A SRS is basically an organization’s understanding of a customer or potential client’s requirements and dependencies at a particular point in time (usually) prior to any actual design or development work. It’s a two way insurance policy that assures that the both client and the organization the others requirements from the perspective at a given point in time.

The SRS document itself states in precise and explicit language those functions and capabilities of a software system must provide, as well as states any requires constrains by which the system must abide. The SRS also function as a blueprint for completing a project with as little cost growth as possible. The SRS is often referred to as the “parent” document because all subsequent project management documents, such as design specification, statements of work, software architecture, testing and validation plans, documentation plans, are related to it.

It’s important to note that an SRS contains functional and nonfunctional requirements only: it doesn’t offer design suggestion, possible solutions to technology or business issues, or any other information other that what the development team understands the customer’s system requirements to be.

**A well designed, well written SRS accomplishes four major goals:**

* It provides feedback to the officer. An SRS is the customer’s assurance that the development organization understands the issues or problems to be solved and the software behavior necessary to address those problems. Therefore, the SRS should be written natural language, in a unambiguous manner that may also include charts, tables, dataflow diagrams, decision tables, and so on.
* It decomposes the problems into component parts. The simple act writing down software requirements in a well design format organize information, places breakdown the problem, solidifies ideas, helps breakdown the problem into its component part in an orderly fashion.
* It serves as an input to design specification. As mentioned previously, the SRS serves as the parent document to subsequent document, such as the software design specification and statements of work. Therefore, the SRS must contain sufficient details the functional system requirements so that a design solution can be devised.
* It serves as a product validation check. The SRS also serves as the parent document for testing and validation strategies that will be applied to the requirements for verification.
* SRS are typically develop during a first stage of “Requirement developments”, which is the initial product requirements are needed and not. This information is gathered about what requirements are needed and not.
* This information gathering stage can include onsite visits questionnaires, surveys, interviews, and perhaps a written on investment (ROI) analysis or needs analysis of the customer or clients current business environment. The actual specification, then, is written after the requirements have been gathered and analyzed.

**FUNCTIONAL REQUIREMENTS:**

General description of inputs and outputs:

The system has basically a menu driven input format. The officer has to choose from the menu, the appropriate options.

**EXTERNAL INTERFACE REQUIREMENTS**:

User interface:

User interface is of the most important parts of the effective software. A menu driven system is to be developed using which the user it. The interface should reinforce.

**SOFTWARE TOOLS USED & CONNECTION**

INTRODUCTION TO VISUAL BASIC AND MICROSOFT OFFICE ACCESS

**INTRODUCTION TO VISUAL BASIC**

Visual basic is a powerful Application Development tool. It is Microsoft Windows programming language. It is evolve from “BASIC LANGUAGE” which was developed by prof’s John Kemeny and Thomas Kurtz of Dartmouth college in the mid of 1960’s



**Visual Basic:**

It is a tool that allows you to develop windows (Graphical User Interface-GUI) applications. The application has familiar appearance to the user. As you develop as a visual basic programmer, you will begin to look at windows applications in a different light, you will organize and understand how various elements of Word, Excel, Access and other application work.

Visual Basic is event driven meaning code remains idle until called upon to respond to some event.

Visual Basic is governed by an event processor. Nothing happens until an event is detected. Once the event is detected, the code corresponding at the event is executed. Program control is then returned to event processor.

Event procedure

Basic Code

Basic Code

Basic Code

Event procedure

All windows applications are driven. For example nothing is happen in the world until we click on to a button, select a menu option, of type some text. Each of these is an event.

* + The event driven nature of visual basic makes it very easy to work with. As you develop a visual basic application event procedures can be building and tested individually saving development time. And, often event procedures are similar in their coding, allowing re-use( and lots of copy and paste).

**Some features of Visual Basic:**

* + Full set of controls you draw the application.
  + Lots of icons and pictures for our use.
  + Response to mouse and keyboard.
  + Clipboard and printer access.
  + Full array of mathematical, string handling and graphic functions.
  + Can handle fixed and dynamic variable and control arrays.
  + Sequential and random access files.
  + Useful debugger and error handling facilities.
  + Powerful database access tools.
  + ActiveX supports.
  + Package & development wizard makes disturbing your applications simple.

**Visual Basic 6 versus Other Versions of Visual Basic:**

* The original visual basic for DOS and visual basic for windows were introduced in the year 1991.
* Visual basic 3 was released in 1993.
* Visual basic 4 was released in 1995.
* Visual basic 5 was released in 1996.
* And now some visual basic 6 some identifies some features of visual basic 6.
  + Faster compiler.
  + New ActiveX data control object.
  + Allows database integration.
  + New data report register.
  + New package and development wizard.
  + Additional internet capabilities.
* Applications built using visual basic 6 will run with Windows 95, 98, 2000 or Windows NT

**STRUCTURE OF VISUAL BASIC**

**APPLICATION**

Control 1

Control 1

Control 1

Control 1

Control 1

Control 1

Control 1

Control 1

Control 1

**Application (project) is made up of**

* **Forms**: Windows that you create for user interface.
* **Controls**: Graphics features draw on forms to allow user interaction.
* **Properties**: Every characteristics of a form or control is specified by a property. Example, property includes names, captions, size, color, position and contents. Visual basic applies default properties. You can change properties at design time or run time.
* **Methods**: built-in procedure that can be invoked to impart some action to a particular object.
* **Event procedure**: code related to some object. This code is the code that is the code that is executed when a certain event occurs.
* **General procedure**: code not related to some objects. This code must be invoked by the application.
* **Modules**: collection of general procedures, variable declarations and constant definitions used by the application.

**Steps in developing Application:**

* The visual basic development environment makes building an application a straight forward process. These are three primary steps involved in building a visual basic application.

1. **Draw** the **user interface** by placing controls on the form.
2. **Assign properties** to the controls.
3. **Attach code** to control events (and perhaps with other procedures).

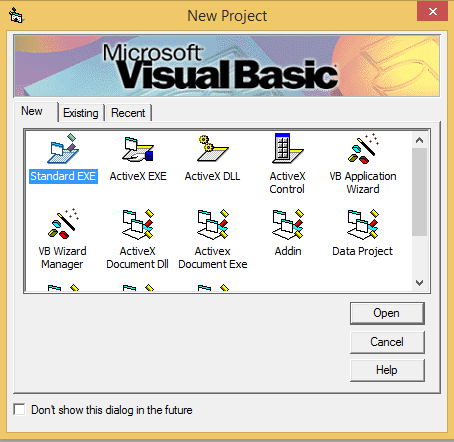
**Starting Visual Basic:**

* We assume you have Visual Basic 6 installed and operational on your computer.
* Click on the **start** button on the windows task bar.
* Select **programs**, then **Microsoft Visual Basic.**
* Click on **Visual Basic 6.0**

Some of the headings given here may differ slightly on your computer, but you should have no trouble finding the correct ones.



Visual Basic will start and this dialog box appears:



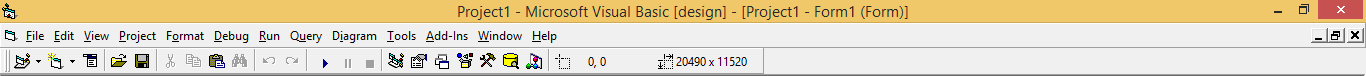
For now, just click on open- we are starting a new project. Later, once you have saved some projects, they can be opened using the existing and recent tabs, the visual basic development environment will start.

**Drawing the user interface and setting properties:**

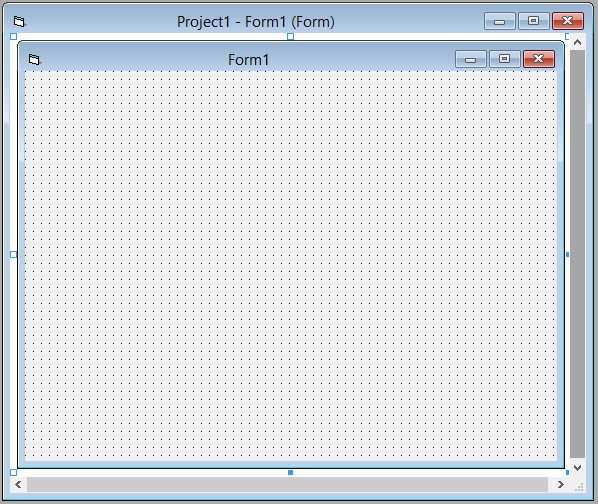
* Visual Basic operates in three mode:
* Design mode- used to build an application.
* Run mode- used to run the application.
* Break mode- application halted and debugger is available.

We focus on hear a **Design** mode.

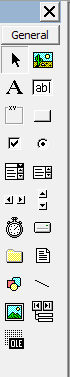
* Six Window appear when you start visual basic. Each window can be viewed by selecting menu options, depressing function keys or using the toolbar. Use the method you feel most comfortable with.
* The **main window** consists of the title bar and toolbar. The title bar indicates the project name, current visual basic operating mode, and the current form. The menu bar has dropdown menus from which you control the operation of the environment. The toolbar has button that provides shortcuts to sum of the menu options. The main window also shows the location of the current from relative to the upper left corner of the screen and the width and length of the current form. Usually just pressing <f1> can get you help you need.



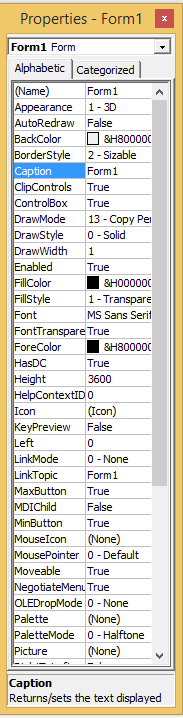
* The **form window** is central to developing Visual Basic application. It is where you draw your application.



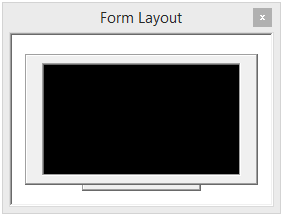
* The **toolbar** is the selection menu for controls used in your computer application help with any control is available by clicking the control and pressing <f1>

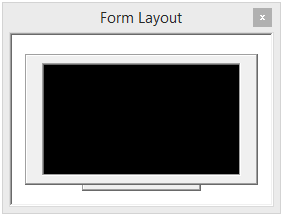


* The **properties window** is used to establish initial property values for objects. The drop-down box at the top of the window lists all objects in the current form. Two views are available: alphabetic properties for the currently selected object.

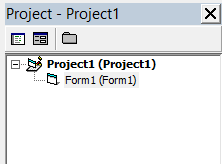


* The **form layout window** shows where your form will be displayed relative to your monitor’s screen.





* The **project window** displays a list of all forms and modules making up your application. You can also obtain a view of the **form** or **code** windows form the project windows.

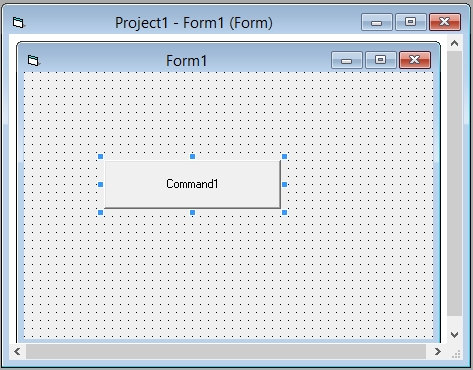


* As mentioned, the user interface is drawn in the form **window.**

There are two ways to place controls on a form.

1. Double click the tool in the toolbox and it is created with a default size on the form. You can then move it or resize.
2. Click the tool in the toolbox then move the mouse pointer to the crosshair at the upper left corner of where you want the control to be, press the left mouse button and hold it down while dragging the cursor toward the lower right corner.

* To move a control you have drawn, click the object in the form window and drag it to the new location. Release the mouse button.
* To resize a control, click the objects so that it is selected and sizing handles appear. Use these handles to resize the objects.



Setting properties of Objects at Design Time:

* Each form and control has properties assigned to it by default when you start up a new project. There are two ways to display the properties of an object. The first way is to click on the object in the form window. Then, click on the properties window or the properties window button in the toolbar. The second way is to first click on the window or the properties window. Then, select the object from the object box in the properties window. Shown in the properties window for the stopwatch application:

Database File

Table

Field

Data

Value

* **Database File**: This is your main file that encompasses the entire database and that is saved to your hard-drive of floppy disk.
* **Example**: StudentDatabase.mdb
* **Table**: A table is a collection of data about a specific topic. There can be multiple tables in a database.
* **Field**: Fields are the different categories within a table. Tables usually contain multiple fields.
* **Data types**: Data types are the properties of each field. A field only has 1 data type.
* Fieldname] Student Last Name
* Data type]Text

The drop-down box at the top of the properties window is the object box. It displays the name of each object in the application as well as its type. This display shows the form object. The properties list is directly below this box. In this list, you can scroll through the list of properties for selected.

**INTRODUCTION TO MICROSOFT OFFICE ACCESS 2007**

**INTRODUCTION:**

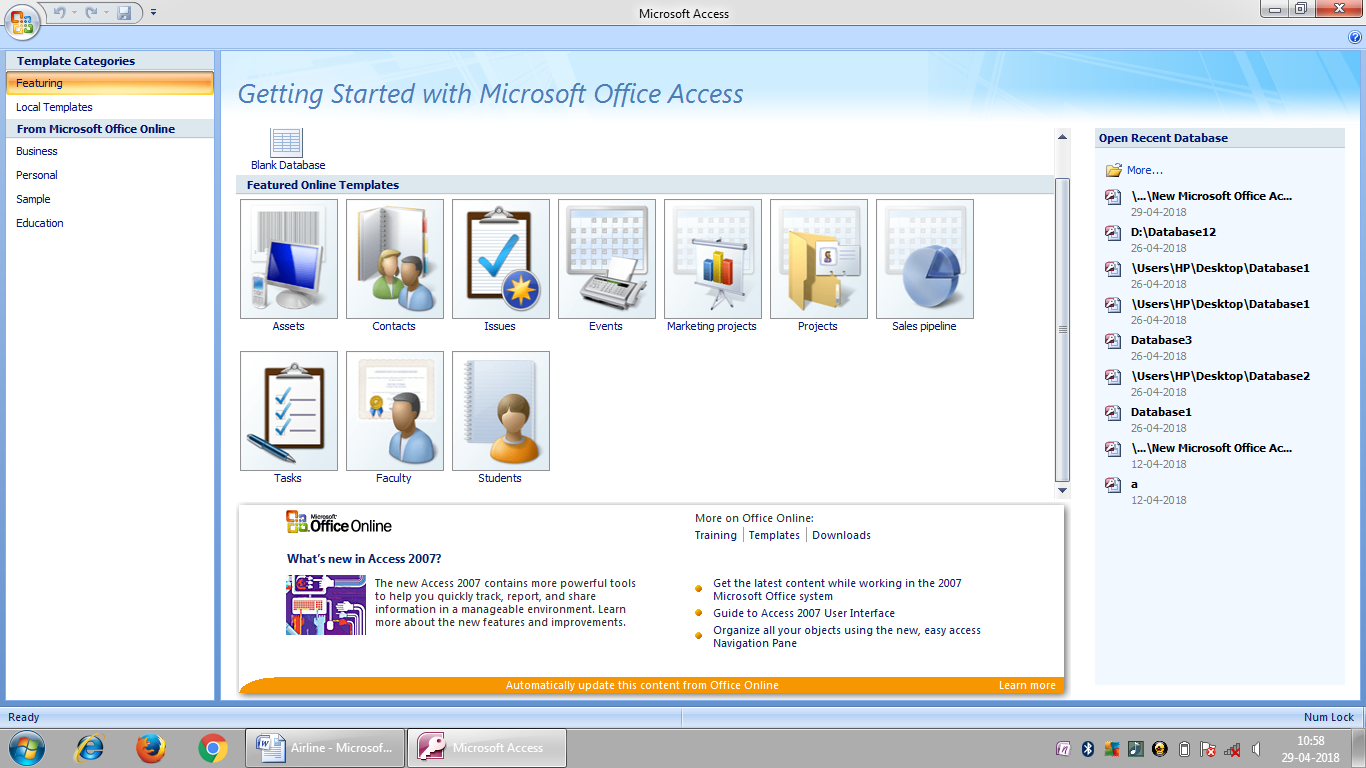
Microsoft Access is a computer application used to creat and manage computer-based databases on desktop computers and/or on connected computers (a network).Microsoft Access can be used for personal information management(PIM), in a small business to organize and manage data, or in an enterprise to communicate with servers.

The Work Area

The main area of the Microsoft Access interface is divided in three sections. The left side displays a column of various links with the top labeled Online Templates.

The middle section itself is made of two sided. The top part allows you to create a database. The bottom section displays some promotional information from Microsoft.

The right side displays a column with the top title labeled Open Recent Database with a More button under it.

**

The bottom section of the Microsoft Access interface displays a status bar.

The Database as a File

Microsoft Access database is primarily a windows file. It must have a location, also called a path, which indicates how the file can be retrieved and made available. Although you can create a database on the root directory such as the C: drive, it is usually a good idea to create your files, including your databases, in an easily recognizable folder.

When you installed the computer (or when it was installed), it (the operating system) might have created a folder called My Documents that provides a convenient place for you to create your files. If various people use the same computer, there is a different My Documents folder for each one. When you log in, the computer(the operating system) locates your corresponding My Documents folder and makes it available.

Creating a Database

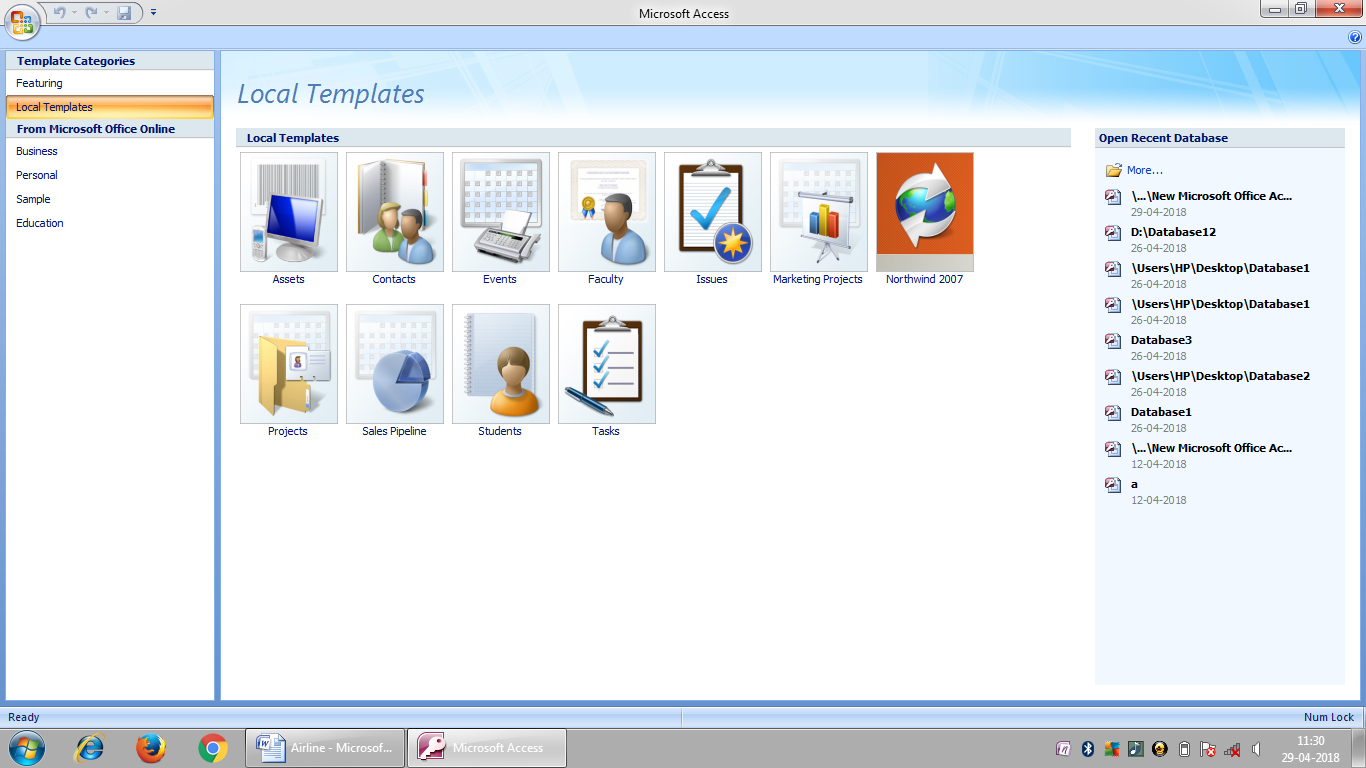
In our lessons, we will learn different techniques of creating a database. For now, a database is first of all a Windows file. It is mainly created from Microsoft Access. If you have just started Microsoft Access, to create a database, you can use one of the links in the main(middle section of the interface).

You can proceed from one of these options. Like every file in the computer, a database must have a name that identifies it. This name must be specified when creating the database.

In our description of the Microsoft Access interface, we saw that the right section displayed an empty area. If you start creating a database as we will see in the next sections and lessons, the right side gets filled with some options, such as prompting you to name your database.

The Database Wizard

Many techniques allow you to create a database, the fastest of which consists of using one of the provided templates. To create a database using one of the samples, in the left section, first click Local Templates, then, in the main section, select one of the samples under Local Templates:

**

The templates are organized in categories. To access a sample by category, in the left section, you can click Business, Personal, or Education. When you click a category, its templates display under Local Templates and you can choose one. After selecting a template, in the right section, you must give a name to your new databse in the File Name textbox. By default, Microsoft Access suggests a name you can use. If you do not like it, you can provide your own. Also, by default, Microsoft Access suggests that the database be created in the My Documents folder. If you want it located in another folder, you can click the Browse button. This would open the File New Database dialog box where you can select an existing folder or create a new one using the Create New Folder button. Display the folder in the save in combo box and click OK. Once you have specified the name of the database and its location, you can click create.

**DATA FLOW DIAGRAM**

DATABASE FLOW DIAGRAM

A DFD or data flow diagram is a graphical representation that depicts the information flow and the information that are applied to data it moved from input to output. A zero level flow diagram is also called a context model. This represents entire project as a bubble with input and output data indicate by incoming and outgoing arrows. The diagram is elaborate is successive levels from the existing level. The symbols used in the DFD are

FUNCTION

FILE/DATABASE

INPUT/OUTPUT

FLOW

CONTEXT-LEVEL DFD

DESTINATION

SOURCE

FIRST-LEVEL DFD

INVALID

VALID

**MDI**

Database

SECOND-LEVEL DFD

DATABASE

ENTITY RELATIONSHIP DIAGRAM

E-R DIAGRAM:

An entity relationship diagram is a graphical depiction of organizational system elements and the association among the elements. E-R diagrams can be help to define the system boundaries. The elements that make up a system are referred to as entities.

An E-R diagram may also indicate the cardinality of a relationship. Cardinality is the number of instances of one entity that can, or must, be associated with each instance of another entity. In general we may speak of one-to-one, one-to-many, or many-to-many relationships. It also distinguishing different types of entities.

An entity is an object or concept about which some information is stored.

THE FOLLOWING ARE USED TO DRAW AN E-R DIAGRAM

RECTANGLES: This represents an entity.

ELLIPSES: This represents attributes.

DIAMONDS: This relationship sets.

LINES: Which link attributes to entity sets and entity sets to relationship sets.

DOUBLE ELLIPSE: Which represented multivalued attributes.

DASHED ELLIPSES: Which represents derived attributes.

THE VARIOUS SYMBOLS USED ARE:-

ENTITY

WEAK ENTITY

RELATIONSHIP

ATTRIBUTE

MULTIVALUE ATTRIBUTE

DERIVED ATTRIBUTE

Connects

Booking

LOGIN

checks

makes

FARE

Flight

confirms

Search

CODING

LOGIN FORM:

Private Sub Command1\_Click()

Dim s1 As String

Dim s2 As String

s1 = Text1.Text

s2 = Text2.Text

If Text1.Text = "" Or Text2.Text = "" Then

MsgBox "enter please"

Exit Sub

End If

Adodc1.RecordSource = "select \* from table1 where Username='" + Text1.Text + "'and Password='" + Text2.Text + "'"

Adodc1.Refresh

If Adodc1.Recordset.RecordCount = 0 Then

MsgBox "invalid login"

Else

MsgBox "valid login"

Form1.Show

End If

End Sub

FORM-1

Private Sub Command1\_Click()

Dim s1 As String

Dim s2 As String

s1 = Text1.Text

s2 = Text2.Text

If Text1.Text = "" Or Text2.Text = "" Then

MsgBox "enter please"

Exit Sub

End If

Adodc1.RecordSource = "select \* from table1 where Username='" + Text1.Text + "'and Password='" + Text2.Text + "'"

Adodc1.Refresh

If Adodc1.Recordset.RecordCount = 0 Then

MsgBox "invalid login"

Else

MsgBox "valid login"

Form1.Show

End If

End Sub

FORM-2

Private Sub Command1\_Click()

Adodc1.Recordset.AddNew

End Sub

Private Sub Command2\_Click()

End

End Sub

Private Sub Command3\_Click()

Adodc1.Recordset("Flightnumber") = Text1.Text

Adodc1.Recordset("Source") = Text2.Text

Adodc1.Recordset("Destination") = Text3.Text

Adodc1.Recordset("Airline") = Combo1.Text

Adodc1.Recordset("Type") = Combo2.Text

Adodc1.Recordset("Fare") = Val(Text4.Text)

Adodc1.Recordset("Class") = Combo3.Text

Adodc1.Recordset.Update

MsgBox "Flight saved"

End Sub

FORM-3

Private Sub Command1\_Click()

Dim Fno As String

Fno = InputBox("Enter Flight number")

Adodc1.RecordSource = "select \*from table1 where Flightnumber ='" + Fno + "'"

Adodc1.Refresh

End Sub

FORM-4

Private Sub Command1\_Click()

If Text1.Text = "Bangalore" And Text2.Text = "Delhi" And Combo1.Text = "Non-sleeper" And Combo2.Text = "Domestic" Then

Text3.Text = 7000

Else If Text1.Text = "Delhi" And Text2.Text = "Los Angeles" And Combo1.Text = "sleeper" And Combo2.Text = "International" Then

Text3.Text = 20000

Else If Text1.Text = "Mangalore" And Text2.Text = "O'Hare" And Combo1.Text = "sleeper" And Combo2.Text = "International" Then

Text3.Text = 30000

End If

End Sub

FORM-5

Private Sub Command1\_Click()

If Text1.Text = "Bangalore" And Text2.Text = "Delhi" And Combo1.Text = "Non-sleeper" And Combo2.Text = "Domestic" Then

Text3.Text = 7000

Else If Text1.Text = "Delhi" And Text2.Text = "Los Angeles" And Combo1.Text = "sleeper" And Combo2.Text = "International" Then

Text3.Text = 20000

Else If Text1.Text = "Mangalore" And Text2.Text = "O'Hare" And Combo1.Text = "sleeper" And Combo2.Text = "International" Then

Text3.Text = 30000

End If

End Sub

FORM-6

Private Sub Command1\_Click()

If Text4.Text = 1 Then

Adodc1.Recordset("Source") = Text1.Text

Adodc1.Recordset("Destination") = Text2.Text

Adodc1.Recordset("date1") = Text3.Text

Adodc1.Recordset("Class") = Combo1.Text

Adodc1.Recordset("Type") = Combo2.Text

Adodc1.Recordset("Firstname") = Text5.Text

Adodc1.Recordset("Lastname") = Text6.Text

Adodc1.Recordset("Age") = Val(Text7.Text)

Adodc1.Recordset("dob") = Text8.Text

Adodc1.Recordset("contactnumber") = Val(Text9.Text)

Adodc1.Recordset("EmailId") = Text10.Text

Adodc1.Recordset.Update

MsgBox "Booking done"

End If

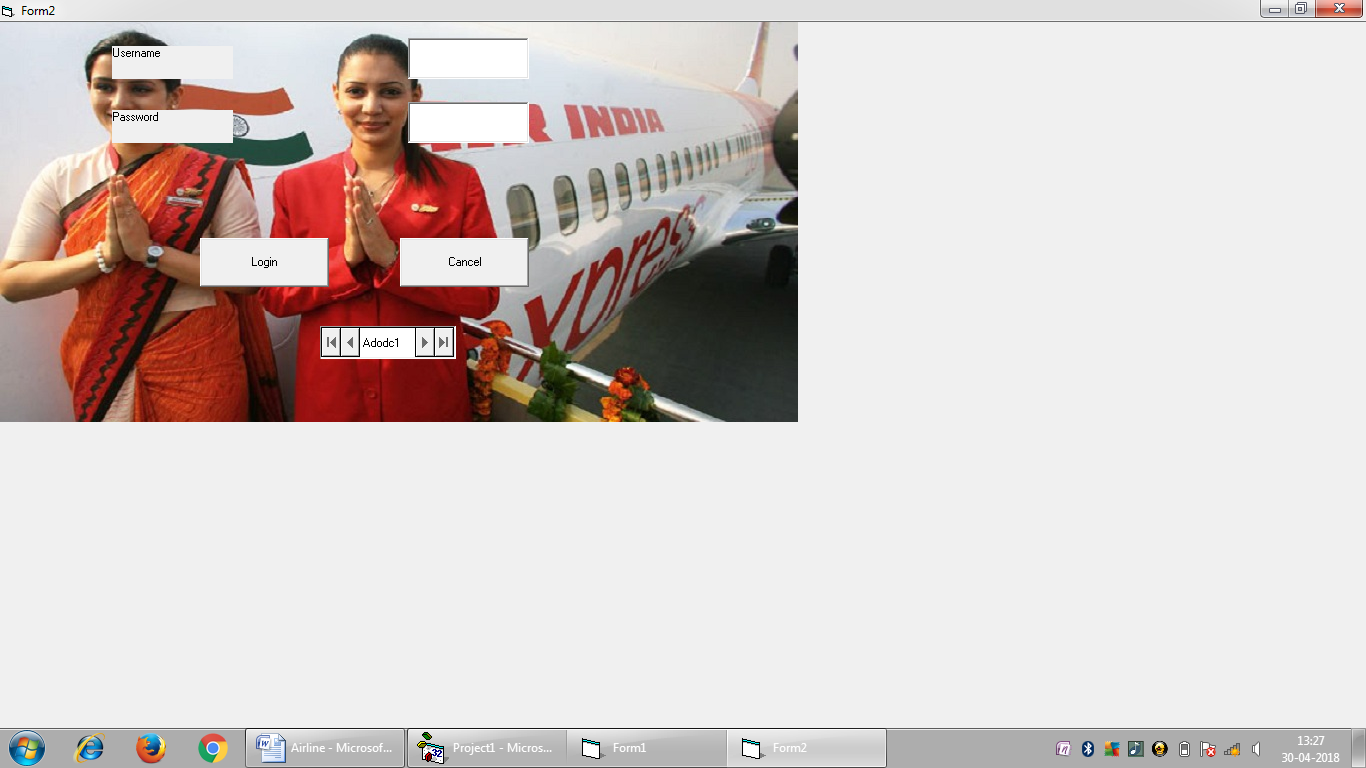
End Sub

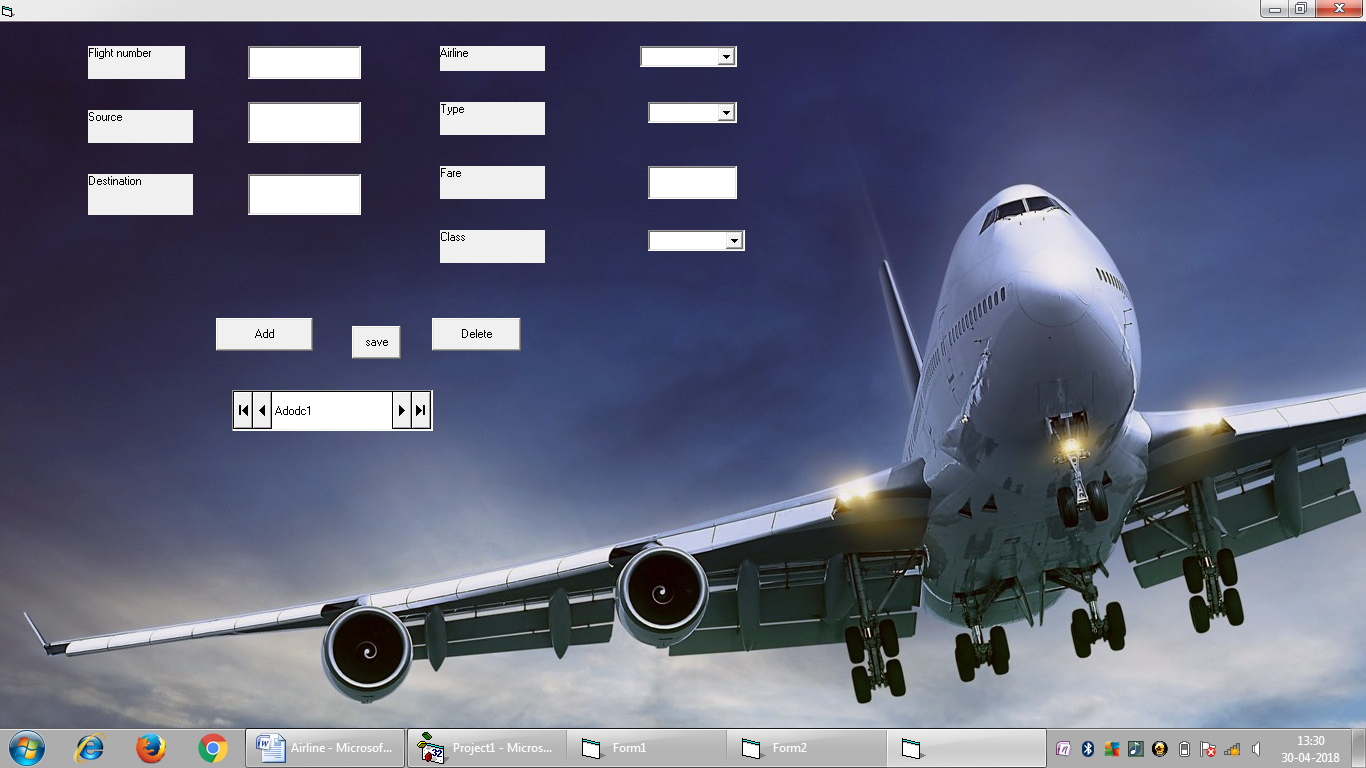
Private Sub Command2\_Click()

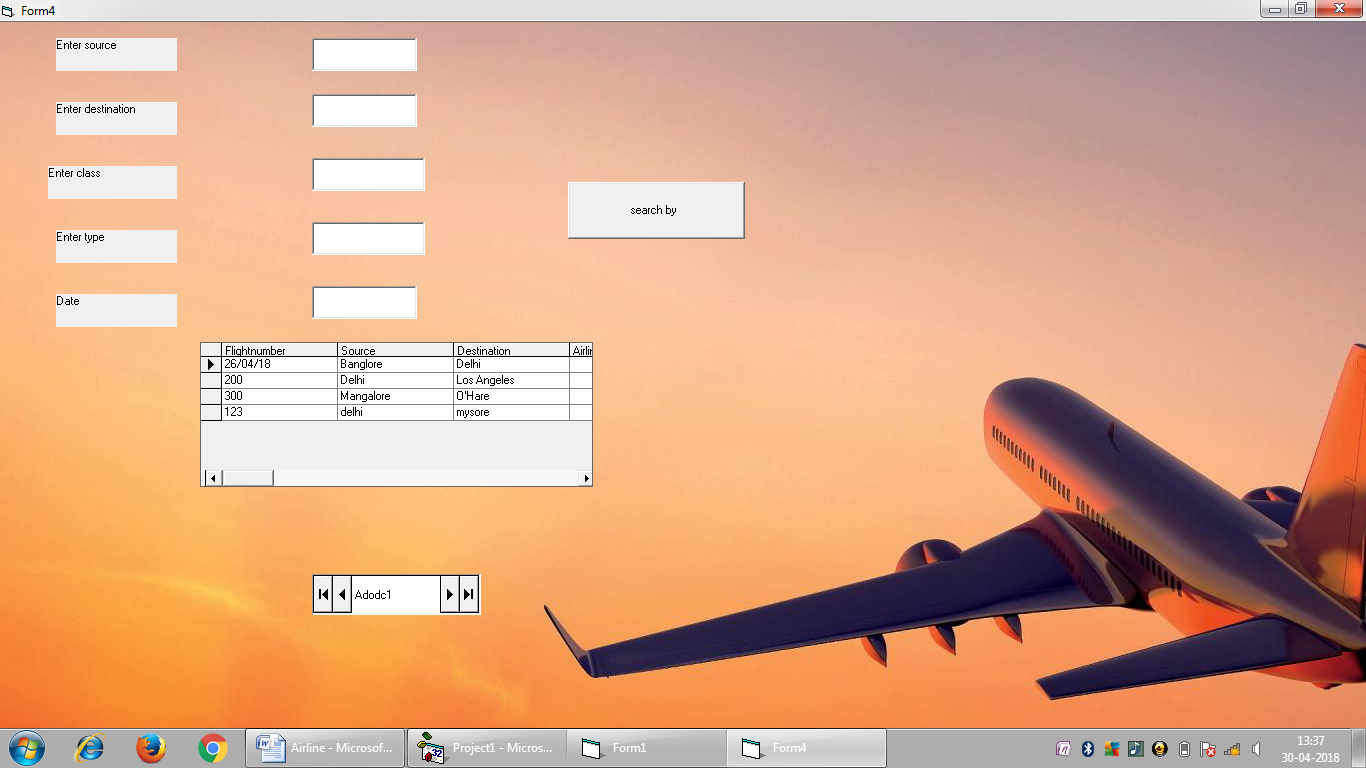
Adodc1.Recordset.AddNew

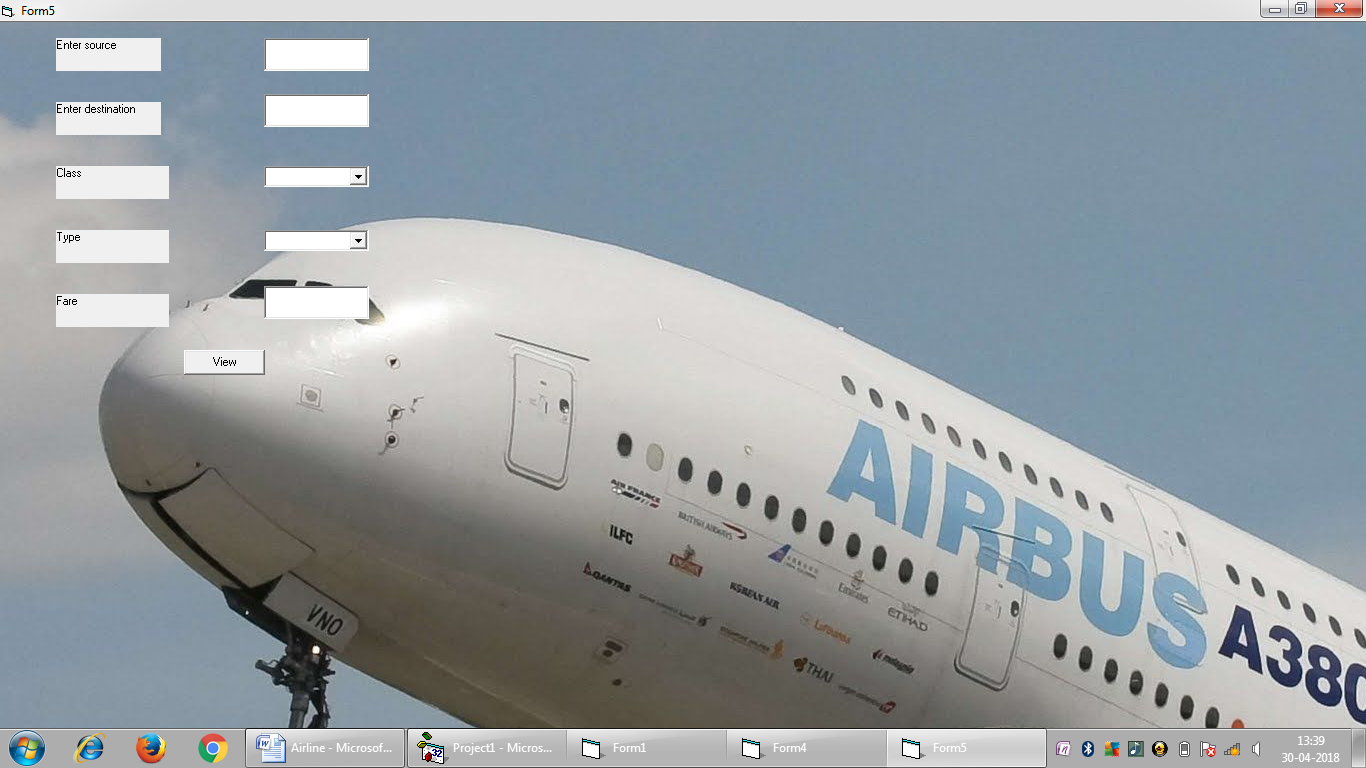
End Sub

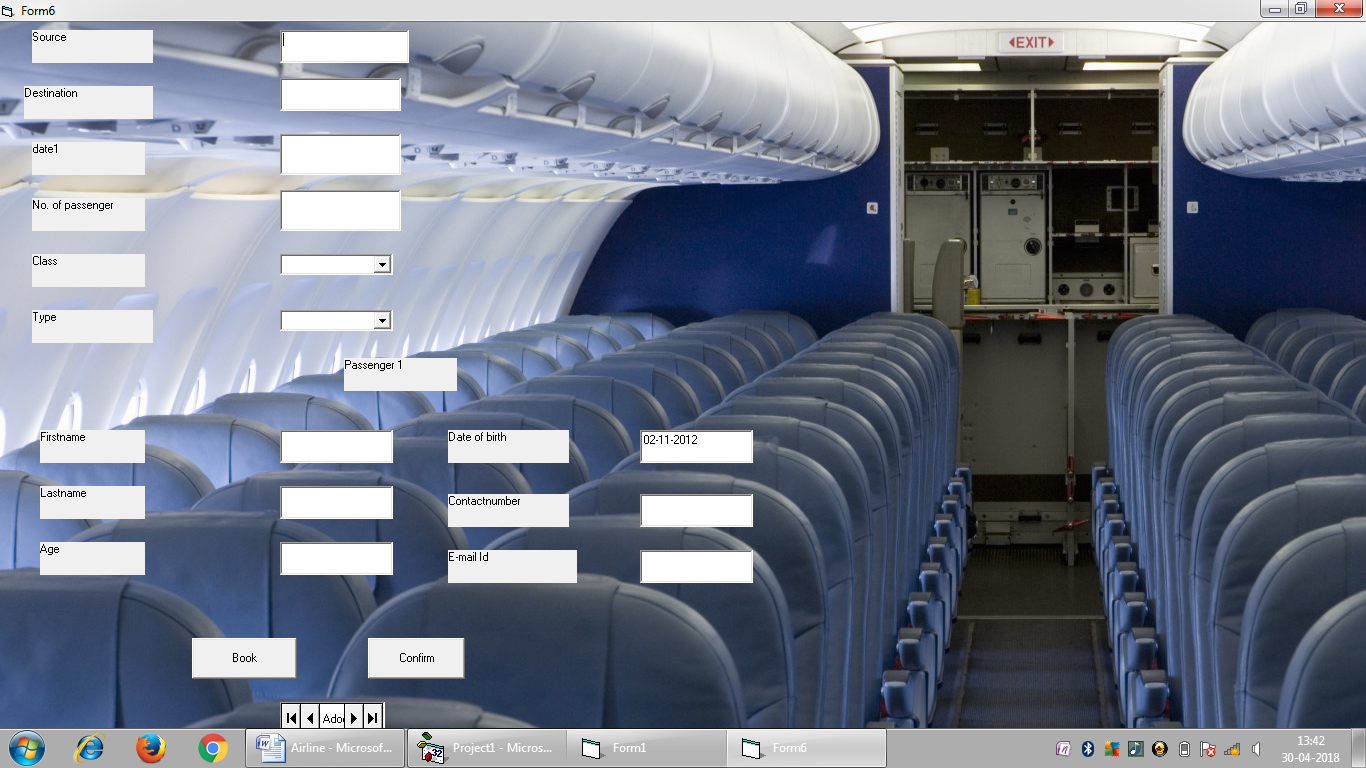
SNAPSHOTS

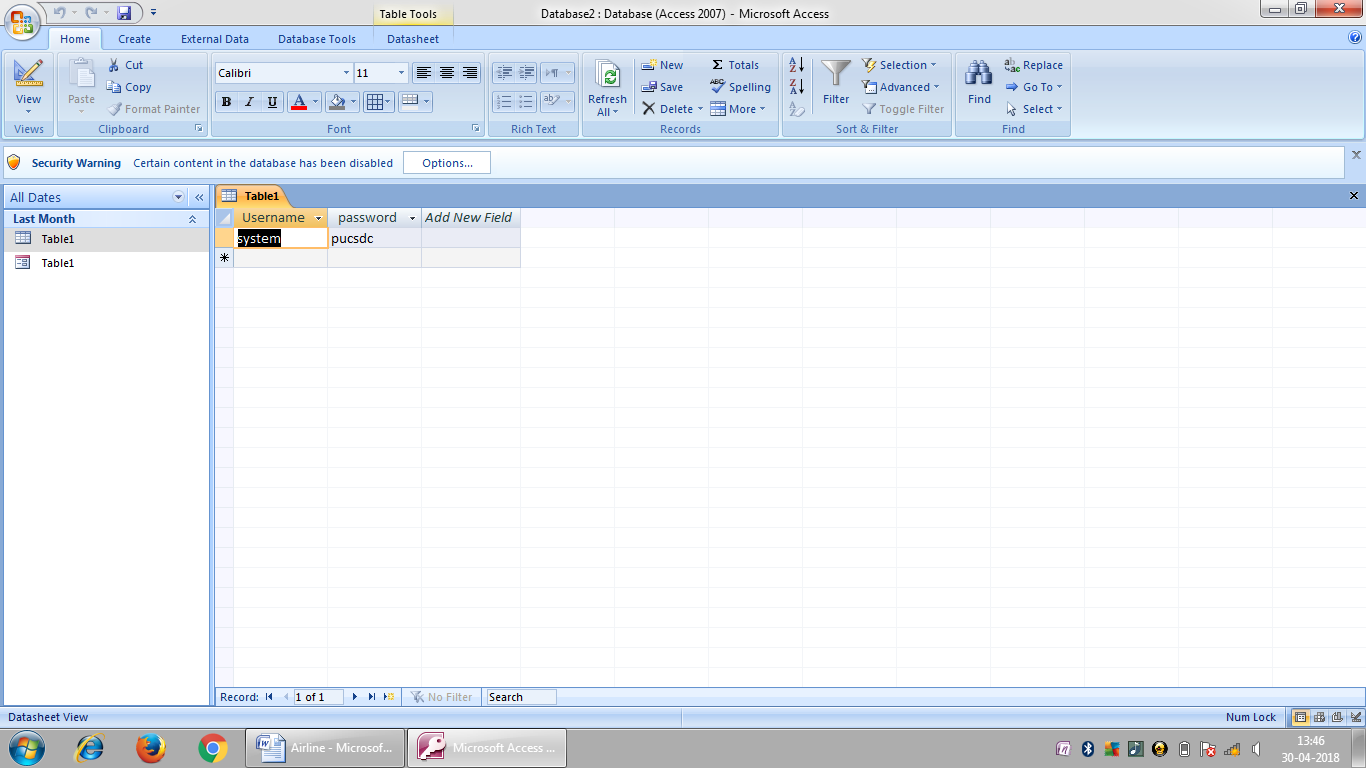


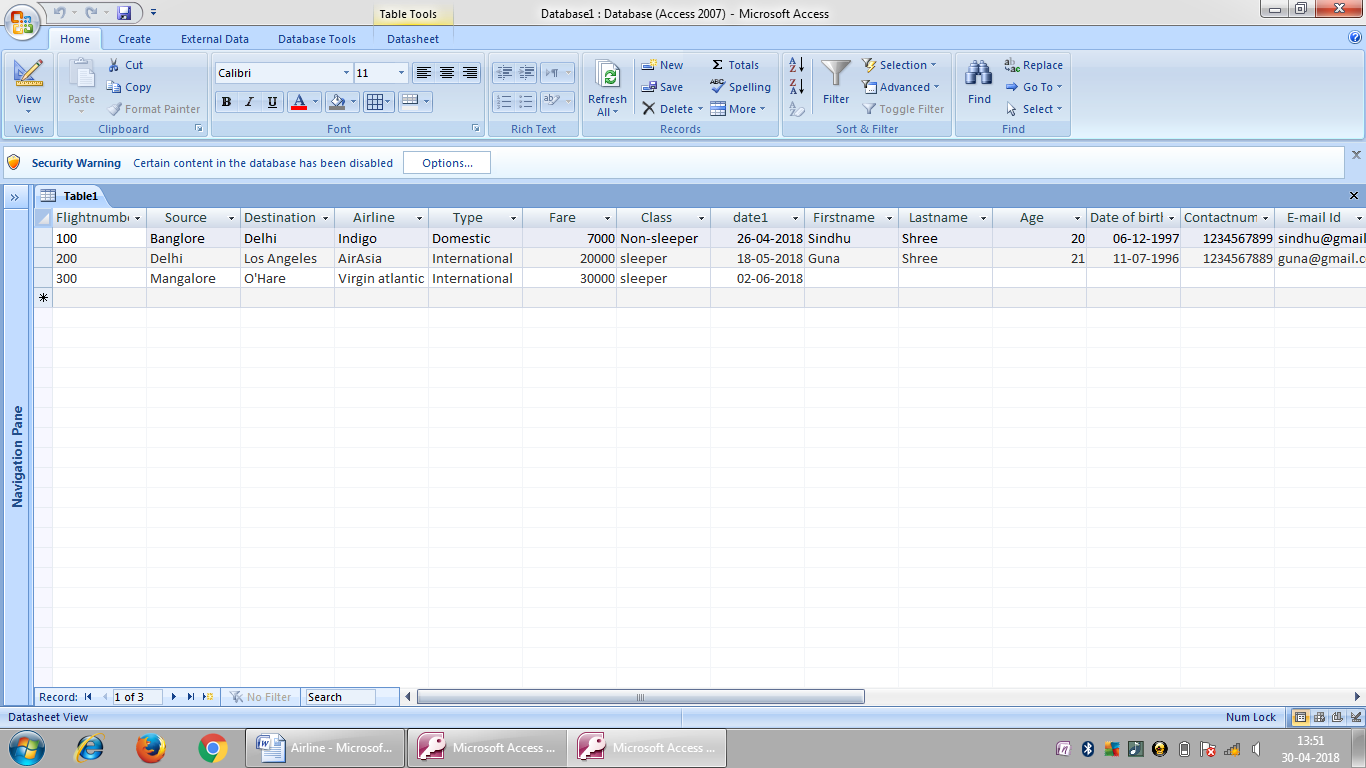


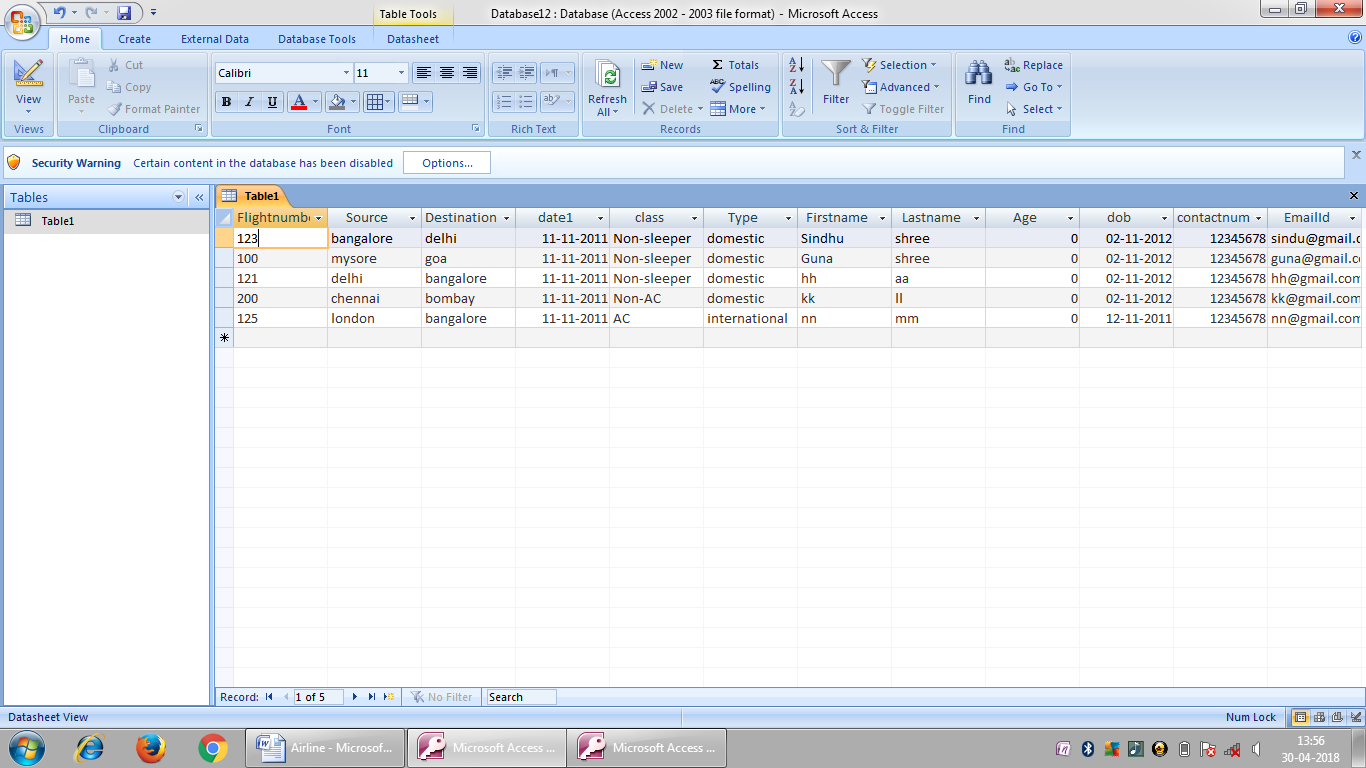












**TESTING:**

Testing is a process of executing a program with the intent of finding an error. The purpose of testing is to identify and correct the errors in the developed program. Nothing is incomplete without testing. It is the vital to the success of the system or program.

Software Testing is a review of specification design and coding. It is a process of detection of errors and enhances software quality. It only show the errors present but cannot show the absent of defects.

Software Testing is an element of software in quality and includes and represents its ultimate view in specification, design and coding. It also includes verification of the basic programs, logics help in running the program. Test case also has some overall testing objectives.

Testing Objectives:

* For finding error in program testing is being used.
* A good test case is one that has high probability of finding an error it was not find.
* Its main intention is to correct the error which is not corrected earlier.

**UNIT TESTING**

Unit testing means that its test’s the smallest unit of software design, the module it is simplified only when a high cohesion is designed when the address of module has only one function. The number of test cases is reduced and errors can be more easily detected.

**INTEGRATION TESTING**

Integration testing is a Systematic Technique for constructing the program structure and at the same time constructing tests to uncover the errors associated with interfacing.

**SYSTEM TESTING**

System Testing is a set of different tests its main function is to intend the compute based system. It is a type of designed to verify the requirements that system meets. It is performed on the running application.

**TEST PLAN**

It is a systematic approach to test a system i.e. software. The plan typically contains a detailed understanding of what the eventual testing workflow will be.

**TEST CASE**

It is a specific purpose of testing a particular requirement.

* Identification of specific requirement tested.
* Test case success.
* Specific steps to execute test.
* Test Data.

**Verification and validation**

* **Verification**: The software should confirm to its specification (Are we building the project right?)
* **Validation**: The software should do what the user really requires (Are we building the right product?)

**Testing Methodologies**

* **Block Box Testing**
* **White Box Testing**

**Future Enhancement:** This is the beginning of our software and our software will be upgraded as comfort of our users and their feedback.

**CONCLUSION**

This project is all about the working of “AIRLINE RESERVATION”.

This project was developed by Visual Basics and Microsoft Access.

We could know how real world constraints has to be dealt with and how any problems that arose and could be solved.

Working on this project made us to learn the language Visual Basic and Microsoft Access.

**BIBILOGRAPHY**

Books referred are:

|  |  |
| --- | --- |
| David Weller | Visual Basic 2005 |
| David Weller | Visual Basic 2005 |