Title: PAYROLL PROCESSING SYSTEM

PROJECT

2019- 2020

Submitted to

SKE Society's

RANI PARVATI DEVI COLLEGE

OF ARTS & COMMERCE

Tilakwadi, Belagavi.

By

Name:_____

Class: _____ Roll no: _____

Date of Submission: _____

Under the guidance of the department of COMPUTER APPLICATION

Student's signature

Staff Signature

S.K.E.SOCIETY'S

RANI PARVATI DEVI COLLEGE OF ARTS AND COMMERCE
BELAGAVI

DEPARTMENT OF COMPUTER APPLICATION

Certificate

Roll No. _____ Date: _____

Examination Seat No. _____

This is to certify that Mr./Ms._____ has successfully carried out the course of exercises in practical Computer Application prescribed by the Rani Channamma University, Belagavi for B.A. ____ semester examination in the Computer laboratory of his college during the year 2018-19.

Staff member in-charge **Head**
Dept. of Computer

External Examiner: _____

Internal Examiner: _____

Student Records Management System in VB 6.0 with Access Database

In this project, we will create an application for Student Record Management System in Visual Basic 6 with Microsoft Access Database.

Contents

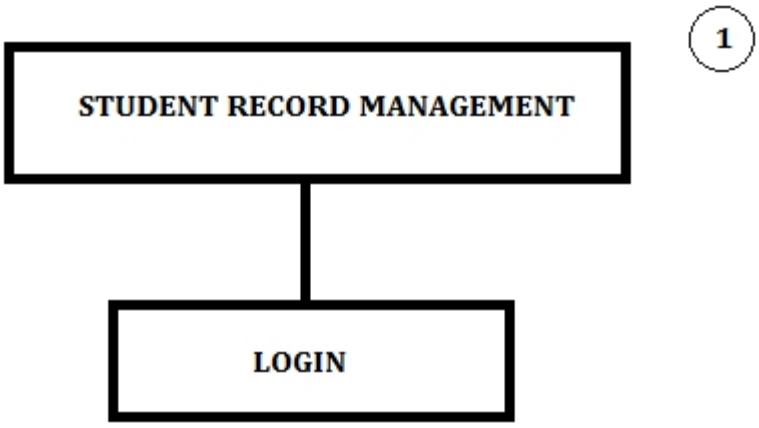
1_System Design
2_Data Model
3_Tables
4_Components of frmStudentManagent System
5_Code for frmStudentManagement System
6_Components of frmLogin
7_Code for frmLogin
8_Components of frmCourse
9_Code for frmCourse
10_Components of frmStudentDetails
11_Code for frmStudentDetails
12_Code for Module1.Bas

The list of Modules and Forms used in the program as follows.

```
(A) frmStudentManagement
Caption: STUDENT RECORD MANAGEMENT SYSTEM
(B) frmLogin
Caption: LOGIN
(C) frmCourse
Caption: COURSE DETAILS
(D) frmStudentDetails
Caption: STUDENT DETAILS
(E) Module1.bas
```

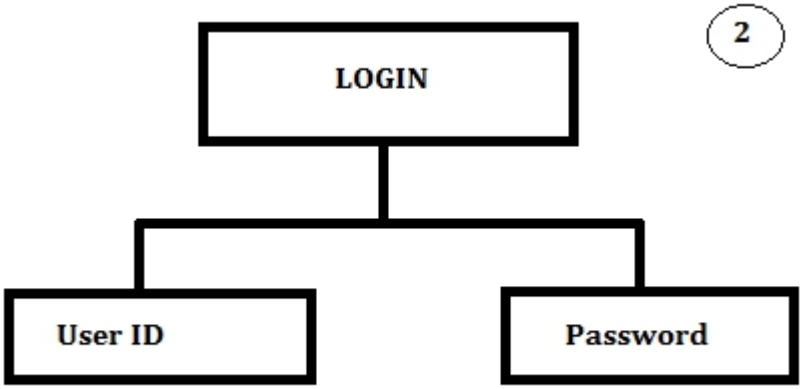
System Design

First, we will design the System Model and simple high-level diagram of each module will be enough. There are many kind of the diagram to model a system especially when you are using Object-Oriented design. These models are specified in Unified Modelling Language (UML) which not scope of this post.



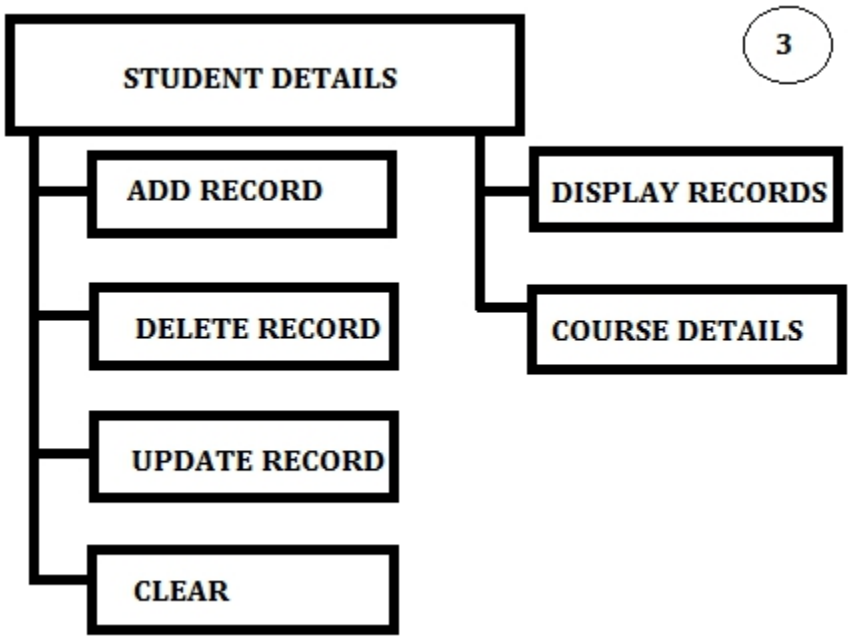
A student must Log In to access the System

The Administrator has to log in to the system with username and password provided during registration process. No one is allowed to log in without a proper User ID and Password.



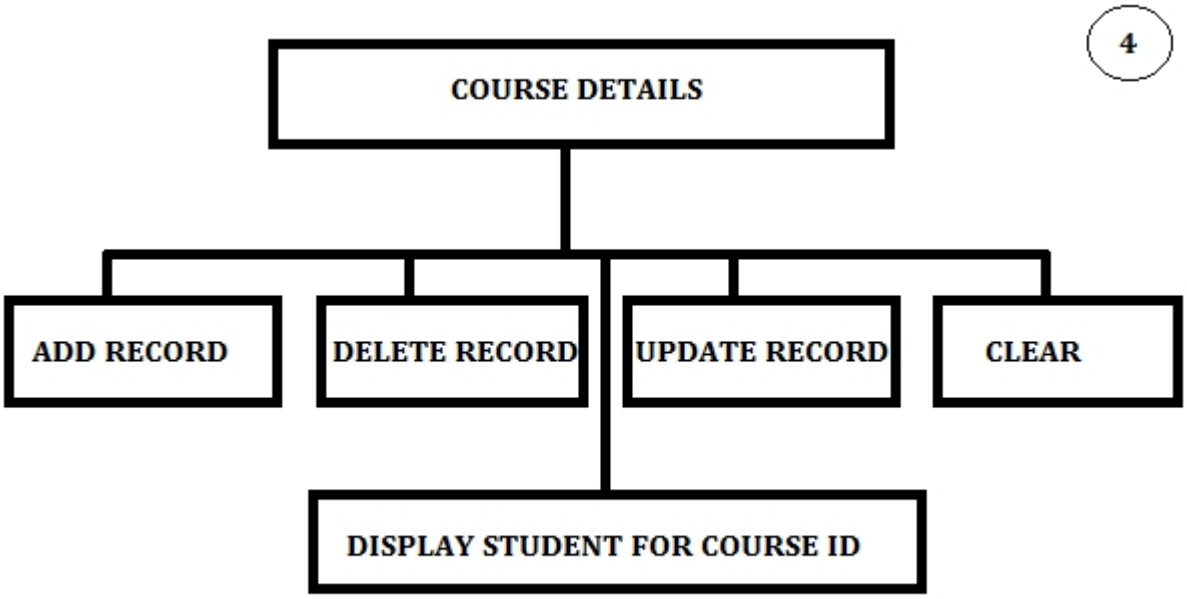
Administrators must provide valid Username and Password

Once the Administrators logged in they can see form for Student Details where they can Insert, Delete, Update and Clear the Student Records. They also have the option to go to First Record, Last Record, Next Record and Previous Record.



Maintain Student Record using Student Details Form

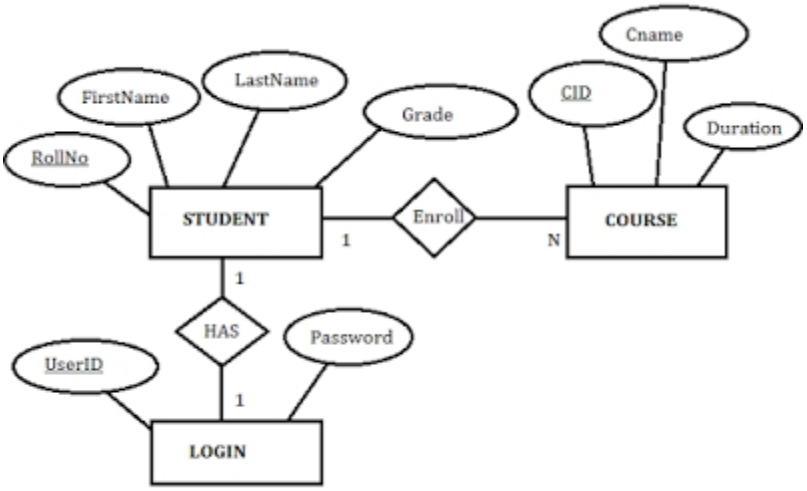
Next, there is an option to maintain the Course Details, when you click a Specific button. We will discuss that later.



Course Details

Data Model

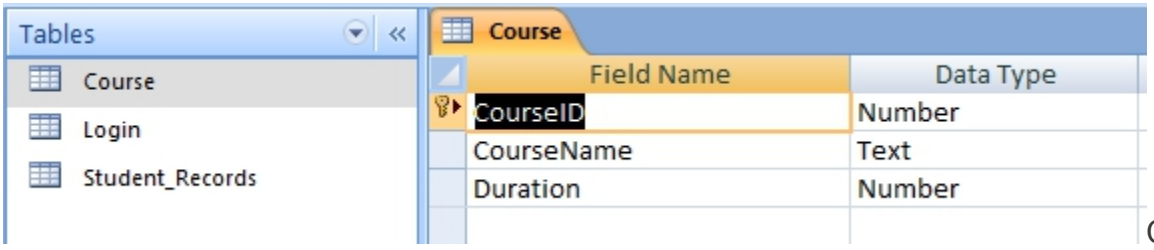
ER Model



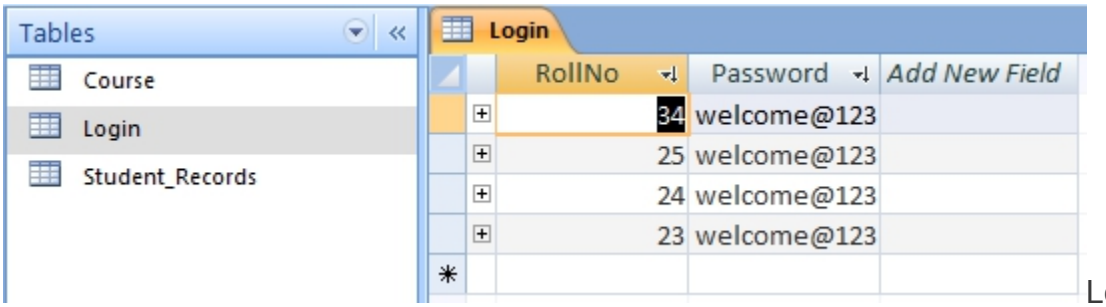
ER Diagram for Student Management

Tables

First steps after creating the data model are to make sure that you create an MS Access Database called **StudentDB.mdb**. If you have MS Access 2007 or higher version, save your database with the file extension (**.mdb**).
Create the following table with correct data types as shown in the following figures.

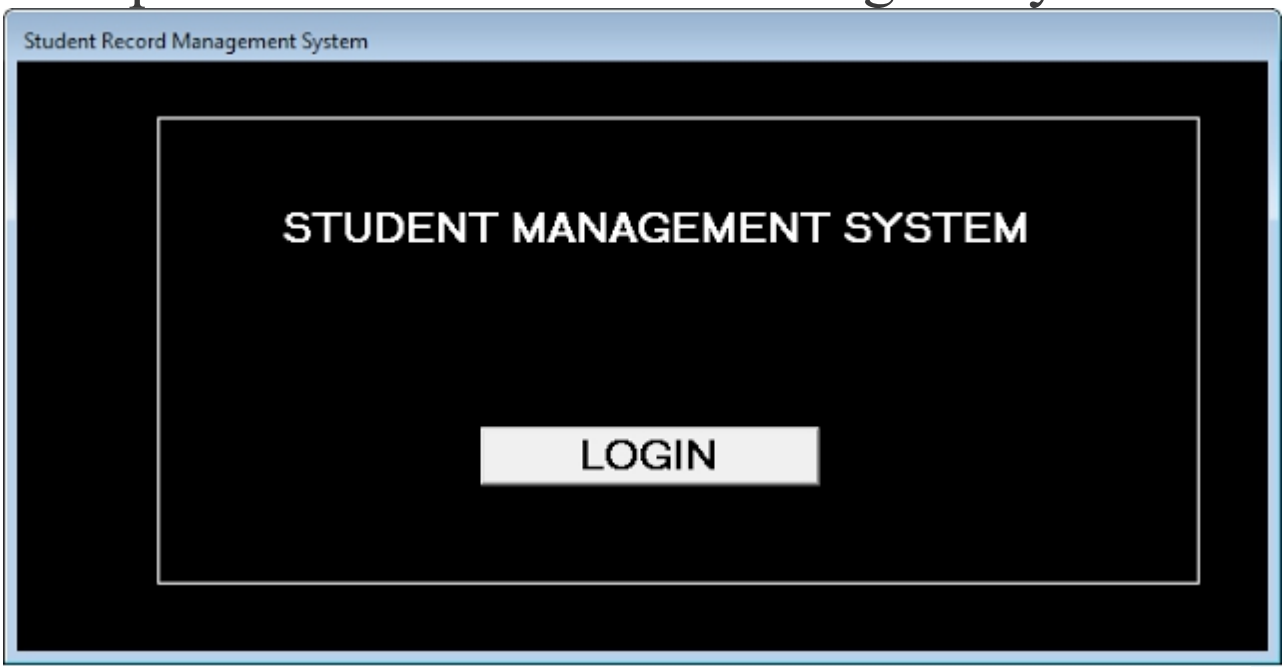


Course Table in Design View



Login Table in Design View

Components of frmStudentManagent System



Form Student Management System

Labels
Name :lblStudentManagement

Caption : Student Management System

Font: MS Sans Serif

Font-Size: 24 PX

Font-Weight : Bold

Buttons
Name : btnLogin

Caption: Log in

Font: MS-Sans-Serif

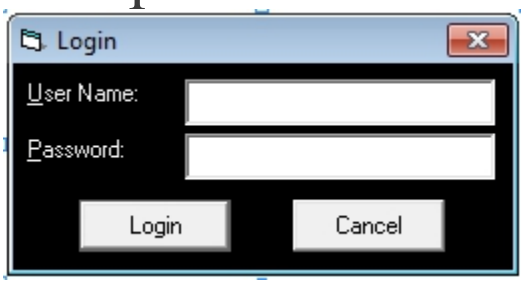
Font-weight:Bold

Font-Size: 10px

Code for frmStudentManagement System

```
Login Button

Private Sub Login_Click()

    frmLogin.Show

End Sub
```

Components of frmLogin



Login Form – Student Management

Labels
Name: lblUserID

Caption : User Name:

Name : lblPassword

Caption: Password

Text-Boxes
Name : txtUserID

Text : ‘leave blank’

Name : txtPassword

Text : ‘leave blank’

Buttons
Name : cmdOK

Caption : Log in

Name : cmdCancel

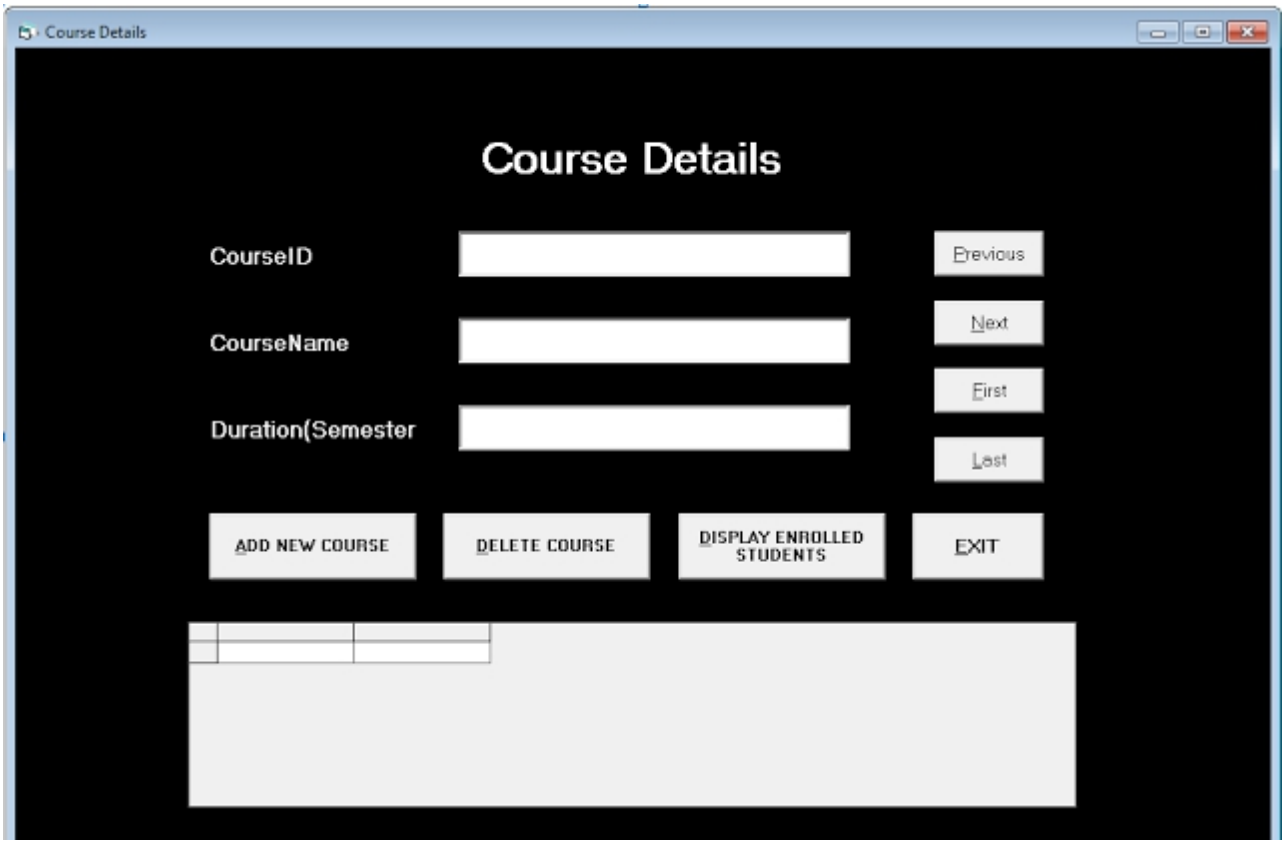
Caption : Cancel

Code for frmLogin

```
Form Login

Option Explicit
Dim rs As New ADODB.Recordset
Dim con As New ADODB.Connection
Dim constr As String

Public LoginSucceeded As Boolean

Private Sub cmdCancel_Click()
    'set the global var to false '
    'to denote a failed login '

    LoginSucceeded = False
    Me.Hide

End Sub

Private Sub cmdOK_Click()

    Dim userfound As Boolean
```

```
    With rs .Open "SELECT * FROM LOGIN",
    con, adOpenDynamic, adLockOptimistic

    'check the username '

    .MoveFirst While Not .EOF
    If txtUserName = !RollNo And
    txtPassword = !Password Then

    frmSTUDREC.Show
    LoginSucceeded = True
    Exit Sub
    Me.Hide

    ElseIf

    txtUserName &lt;> !RollNo Then
    .MoveNext
    Else
    MsgBox "Invalid Password, try again!", ,
    "Login" txtPassword.SetFocus
    SendKeys "{Home}+{End}"

    Exit Sub
    End If
    Wend
    .Close
    End With
End Sub

Private Sub Form_Load()
    Call loadcon
    MsgBox ("connected")
End Sub
```

Components of frmCourse



Form Course Details – Student Management

Labels
Name : lblCID

Caption: Course ID

Name : lblCname

Caption : Course Namev

Name : lblDuration

Caption : Duration(Semester):

Text-Boxes
Name : txtCID

Text : ‘leave blank’

Name : txtCname

Text : ‘leave blank’

Name : txtDurationv

Text : ‘leave blank’

Buttons

Name : cmdAdd

Caption : &ADD NEW COURSE

Name : cmdDel

Caption : &DELETE COURSE

Name : cmdDisplay

Caption : &DISPLAY ENROLLED STUDENTS

Name : cmdExit

Caption : &EXIT

Name : cmdFirst

Caption : First Record

Name : cmdLast

Caption : :Last Record

Name : cmdNext

Caption : Next Record

Name ; cmdPrev

Caption : Previous Record

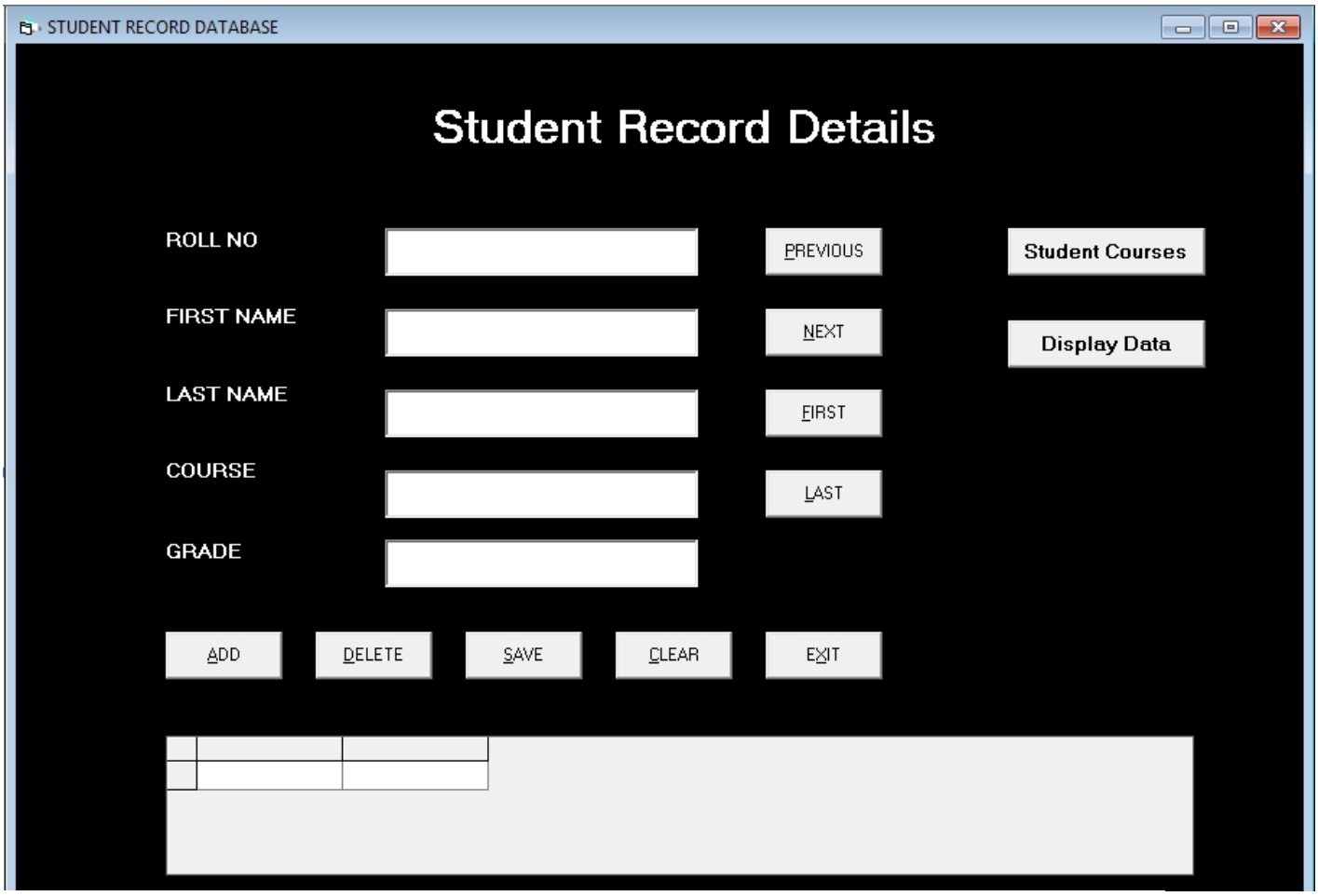
Code for frmCourse

```
Form Course

Dim con As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim constr As String

Private Sub cmdAddCourse_Click()

    con.Execute "INSERT INTO
    Course(CourseID, CourseName, Duration)
    VALUES ('" + txtCID.Text + "','" +
    txtCNAME.Text + "','" + txtDURATION.Text + "')"
    txtCID.Text = "" txtCNAME.Text = "" txtDURATION.Text = ""

    con.Close

End Sub

Private Sub cmdDelCourse_Click()

    con.Execute "DELETE * FROM COURSE
    WHERE CourseID = " &
    txtCID & ""

End Sub

Private Sub cmdDisplay_Click()

    Dim rsGrid As New ADODB.Recordset
    con.CursorLocation = adUseClient rsGrid.CursorLocation
    = adUseClient rsGrid.CursorType = adOpenStatic

    rsGrid.Open "SELECT * FROM Student_Records
    WHERE CourseID = " &
    txtCID & "",
    con, adOpenForwardOnly, adLockPessimistic

    Set DataGrid1.DataSource = rsGrid
```

```
End Sub

Private Sub cmdFirst_Click()

    With rs .Open "SELECT * FROM Course",
    con, adOpenDynamic, adLockOptimistic

    While Not .EOF
    txtCID.Text = rs!CourseID
    txtCNAME.Text = rs!CourseName
    txtDURATION.Text = rs!Duration

    .MoveFirst
    Wend
    End With

End Sub

Private Sub cmdLast_Click()

    With rs .Open "SELECT * FROM Course",
    con, adOpenDynamic, adLockOptimistic

    While Not .EOF
    txtCID.Text = rs!CourseID
    txtCNAME.Text = rs!CourseName
    txtDURATION.Text = rs!Duration
    .MoveLast

    Wend
    End With

End Sub

Private Sub cmdNext_Click()

    With rs .Open "SELECT * FROM Course",
    con, adOpenDynamic, adLockOptimistic

    While Not .EOF

    txtCID.Text = rs!CourseID
    txtCNAME.Text = rs!CourseName
    txtDURATION.Text = rs!Duration
```

```
    .MoveNext
    Wend
    End With
End Sub

Private Sub cmdPrev_Click()

    With rs .Open "SELECT * FROM Course",
    con, adOpenDynamic, adLockOptimistic

    While Not .EOF
    txtCID.Text = rs!CourseID
    txtCNAME.Text = rs!CourseName
    txtDURATION.Text = rs!Duration

    .MovePrevious
    Wend
    End With

End Sub

Private Sub Form_Load()

    Call loadcon
    MsgBox ("Connected")

End Sub
```

Components of frmStudentDetails



Form Student Record Details – Student Management

Labels

Name : lblStudentID

Caption : STUDENT ID

Name : lblFirstName

Caption : FIRST NAME:

Name : lblLastName

Caption : LAST NAME:

Name : lblCourseID

Caption : COURSE ID

Name : lblGrade

Caption : GRADE

Text-Boxes
Name : txtStudentID

Text : ‘leave blank’

Name : txtFirstName

Text : ‘leave blank’

Name : txtLastName

Text : ‘leave blank’

Name : txtCourseID

Text : ‘leave blank’

Name : txtGrade

Text : ‘leave blank’

Buttons
Name : cmdAdd

Caption : &ADD

Name : cmdDel

Caption: &DELETE

Name : cmdUpdatev

Caption : &UPDATE

Name : cmdClear

Caption : &CLEAR

Name : cmdExit

Caption : &EXIT

Name : cmdDisplay

Caption : &DISPLAY

Name : cmdStudentCourses

Name: cmdNext

Caption : &Next Record

Name : cmdPrev

Caption : &Previous Record

Name : cmdFirst

Caption : &First Record

Name : cmdLast

Caption : &Last Record

Data Grid Control
Name : DataGrid1

Code for frmStudentDetails

```
Form Student Details

Dim con As New ADODB.Connection
Dim rs As New ADODB.Recordset

Dim constr As String

Private Sub btnCourse_Click()
    Unload Me
    frmCourse.Show
End Sub

Private Sub cmdData_Click()
    Dim rsGrid As New ADODB.Recordset
    con.CursorLocation = adUseClient
    rsGrid.CursorLocation = adUseClient
    rsGrid.CursorType = adOpenStatic
    rsGrid.Open "SELECT * FROM Student_Records",
    con, adOpenForwardOnly, adLockPessimistic
    Set DataGrid1.DataSource = rsGrid
End Sub

Private Sub cmdDel_Click()

    con.Execute "DELETE * FROM
    STUDENT_RECORDS WHERE RollNo = " &
    (Text1.Text) & ""
    MsgBox ("Record Deleted Successfully")

End Sub

Private Sub cmdExit_Click()
    Unload Me
End Sub

Private Sub cmdFirst_Click()

    With rs
    .Open "SELECT * FROM STUDENT_RECORDS",
    con, adOpenDynamic, adLockOptimistic

    Text1.Text = rs!RollNo
    Text2.Text = rs!FirstName
```

```
    Text3.Text = rs!LastName
    Text4.Text = rs!CourseID
    Text5.Text = rs!Grade
    .MoveFirst
    .Close

    End With

End Sub

Private Sub cmdLast_Click()
    With rs
    .Open "SELECT * FROM STUDENT_RECORDS",
    con, adOpenDynamic, adLockOptimistic
    Text1.Text = rs!RollNo
    Text2.Text = rs!FirstName
    Text3.Text = rs!LastName
    Text4.Text = rs!CourseID
    Text5.Text = rs!Grade

    .MoveLast
    End With

End Sub

Private Sub cmdNext_Click()

    With rs
    .Open "SELECT * FROM STUDENT_RECORDS",
    con, adOpenDynamic, adLockOptimistic
    While Not .EOF

    Text1.Text = rs!RollNo
    Text2.Text = rs!FirstName
    Text3.Text = rs!LastName
    Text4.Text = rs!CourseID
    Text5.Text = rs!Grade
    .MoveNext
    Wend

    .Close

    End With

End Sub
```

```
Private Sub cmdPrev_Click()
    With rs
   .Open "SELECT * FROM STUDENT_RECORDS",
    con, adOpenDynamic, adLockOptimistic

    While Not .EOF

    Text1.Text = rs!RollNo
    Text2.Text = rs!FirstName
    Text3.Text = rs!LastName
    Text4.Text = rs!CourseID
    Text5.Text = rs!Grade

    .MovePrevious
    Wend

    End With

End Sub

Private Sub cmdSave_Click()

    con.Execute "UPDATE STUDENT_RECORDS
    SET FirstName = '" & Text2 & "',
    LastName = '" & Text3 & "',
    CourseID =" & Text4 & ",
    Grade = '" & Text5 & "'
    WHERE RollNo = " & Text1 & ""

  MsgBox ("Record Updated")

End Sub

Private Sub Form_Load()

    constr = "Provider=Microsoft.Jet.OLEDB.4.0;Data
    Source=C:UsersGirishDocumentsStudentDB.mdb;
    Persist Security Info=False"
    con.Open constr

End Sub

Private Sub cmdAdd_Click()

    con.Execute "INSERT INTO
    STUDENT_RECORDS(RollNo,FirstName,LastName,CourseID,Grade)
```

```
    VALUES ('" + Text1.Text + "','" +
    Text2.Text + "','" +
    Text3.Text + "','" +
    Text4.Text + "','" +
    Text5.Text + "')"

    Text1.Text = ""
    Text2.Text = ""
    Text3.Text = ""
    Text4.Text = ""
    Text5.Text = ""

    con.Close

End Sub
```

Code for Module1.Bas

The **Module1.Bas** is the code for connection to Microsoft Access Database. Since you may not want to define this connection on each form that is used in the application.

```
Module Connection

Public Sub loadcon()
constr = "Provider=Microsoft.Jet.OLEDB.4.0;
Data Source=C:UsersGirishDocumentsStudentDB.mdb;
Persist Security Info=False"
con.Open constr
End Sub
```

Note : *The `C:\Users\Peter\Documents\StudentDB.mdb;`* is path to the Access database , you need to change is according to path of your Student Database.

Title: PAYROLL PROCESSING SYSTEM

PROJECT

2019- 2020

Submitted to

SKE Society's

RANI PARVATI DEVI COLLEGE

OF ARTS & COMMERCE

Tilakwadi, Belagavi.

By

Name:_____

Class: _____ Roll no: _____

Date of Submission: _____

Under the guidance of the department of COMPUTER APPLICATION

Student's signature Staff Signature

S.K.E.SOCIETY'S

RANI PARVATI DEVI COLLEGE OF ARTS AND COMMERCE
BELAGAVI

DEPARTMENT OF COMPUTER APPLICATION

Certificate

Roll No. _____ Date: _____

Examination Seat No. _____

This is to certify that Mr./Ms._____ has successfully carried out the course of exercises in practical Computer Application prescribed by the Rani Channamma University, Belagavi for B.A. ____ semester examination in the Computer laboratory of his college during the year 2018-19.

Staff member in-charge **Head**
Dept. of Computer

External Examiner: _____

Internal Examiner: _____

Employees Record Management using VB 6.0 and MS Access Database

In this project, we will learn to create an application in Visual Basic 6.0 for Employee Record Management in an organization.

Contents hide

1 System Model
2 E-R Diagram for Employee Record Management
3 Working with MS Access Database
4 Accessing Access Database from Visual Basic 6
5 Creating Module1 for Database Connectivity
6 Code for frmEMPMANAGE
7 Components of frmLogin
8 Code for frmLogin
9 Components for frmEmpDetails
10 Code for frmEmpDetails
11 Component for frmSalary
12 Code for frmSalary
13 Components for frmDependent
14 Code for frmDependent
15 frmMenu [MDI Form]
16 Create a New MDI Form
17 Code for frmMENU

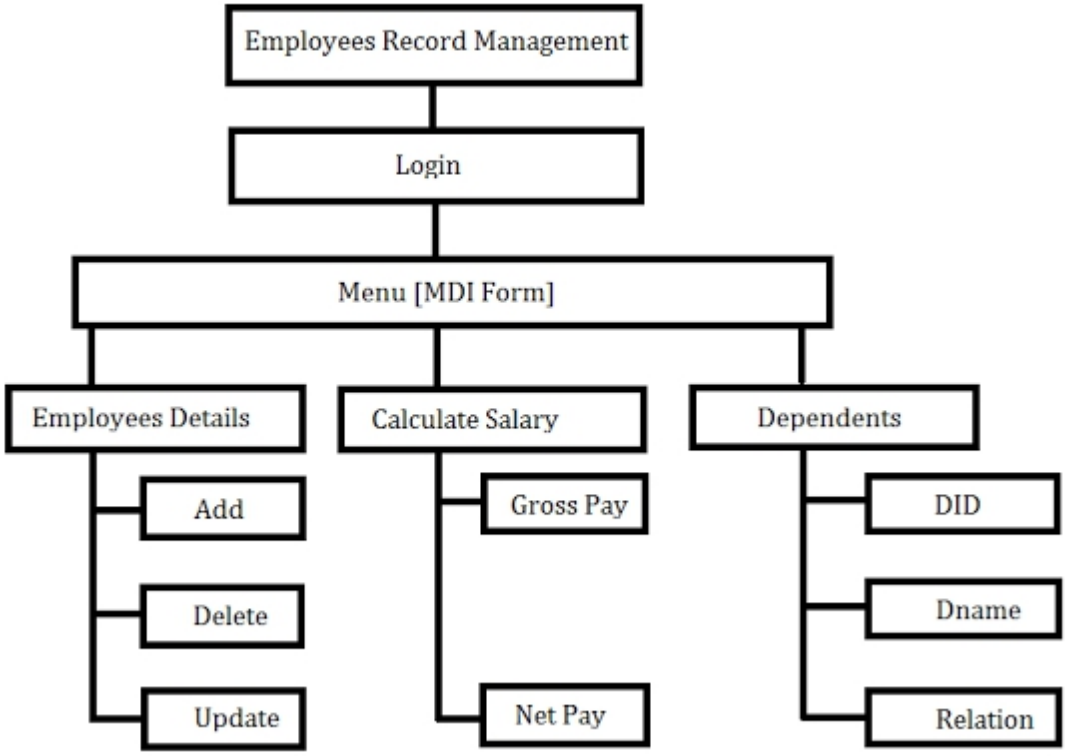
The application does following tasks

1. Insert, Delete, Update employee records.
2. Compute the Gross Pay and Net Pay of each employee.
3. Maintain records of departments for each employee.
4. Each employee should be able to login to the system with own password.

The development of the application – Employees Record Management is done in following ways.

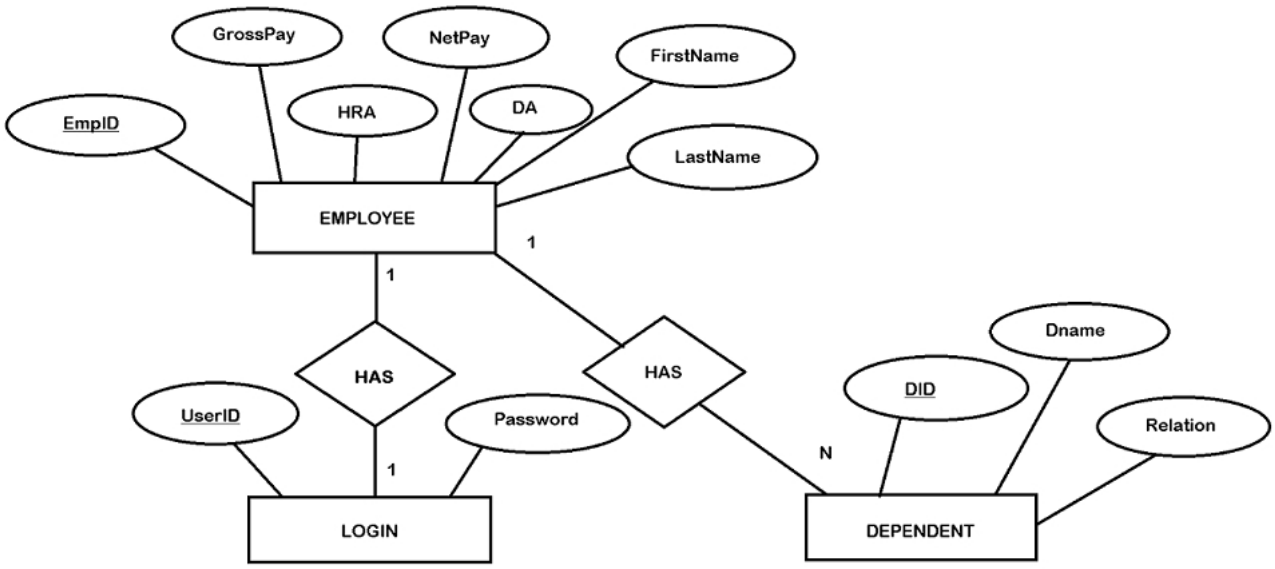
- Develop System Model.
- Define the Data Model (E-R Diagram)
- Design the User Interface in VB 6.
- Rename the components.
- Write codes for components.
- Test and verify Output.

System Model



System Model – Employee Record Management

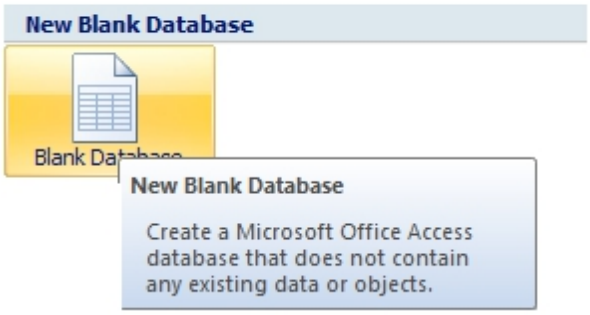
E-R Diagram for Employee Record Management



E-R Diagram – Employee Record Management

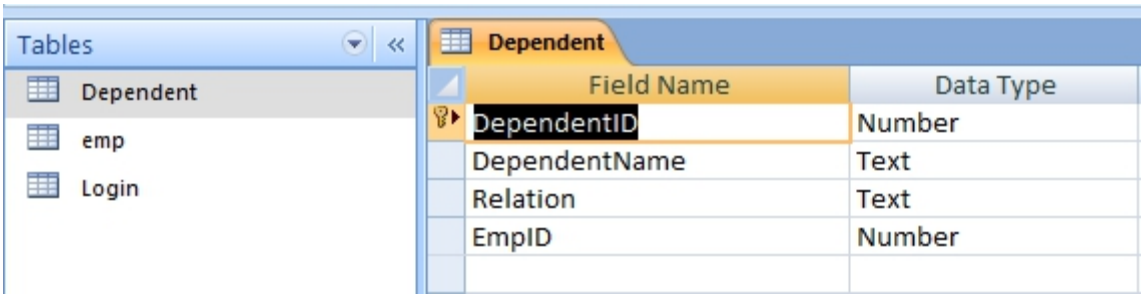
Working with MS Access Database

Based on the E-R Model you can create a New Access Database and save it as the **emp.mdb** file. To create a blank database
Open Microsoft Office Access 2007 (Check MS office documentation for other versions) > Click **"Blank Database".**



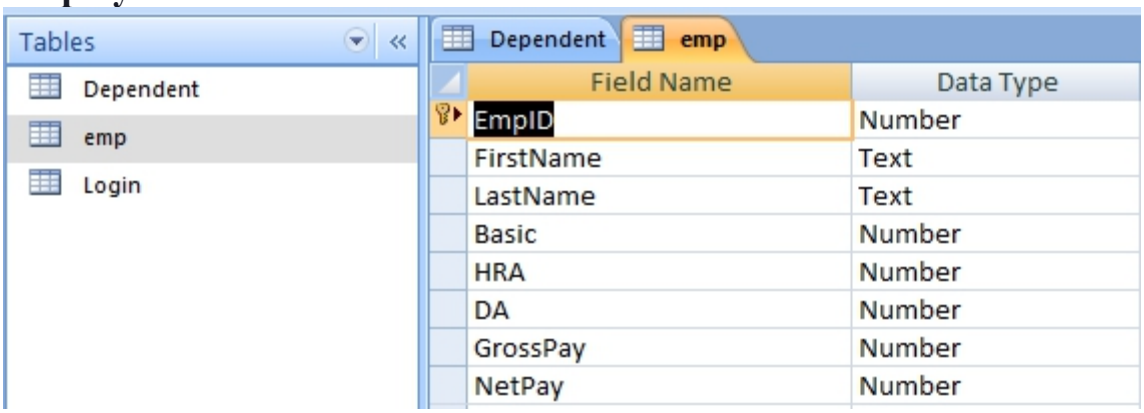
Click Blank Database

Create three table for the **emp.mdb** database.
Dependent Table Schema



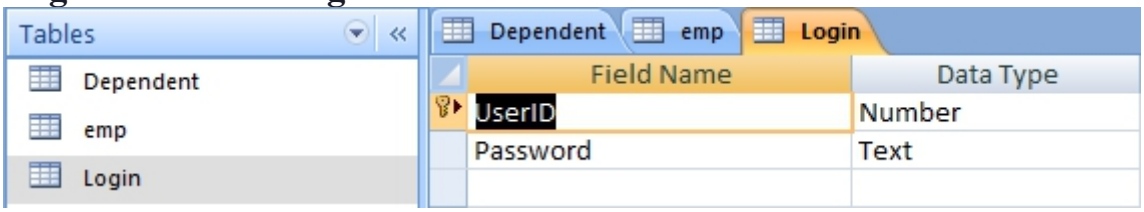
Dependent Table in Design View

Employee Table Schema



Employee Table in Design View

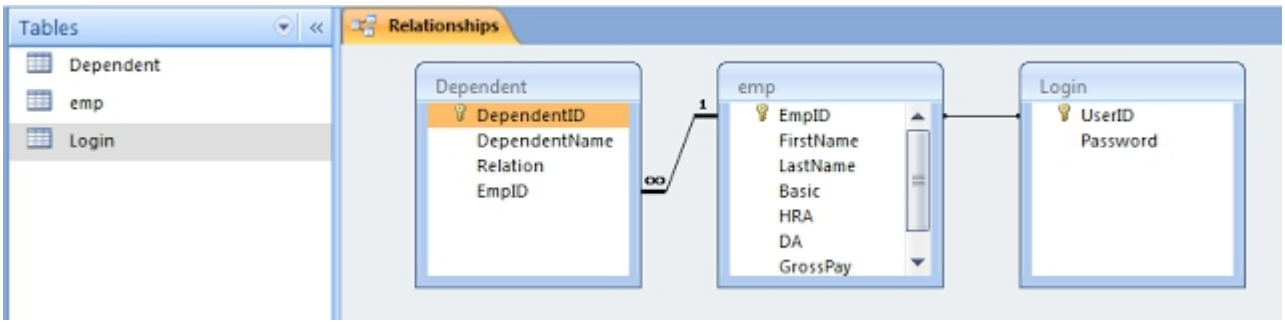
Login Table in Design View



Login Table in Design View

Note: Each Table should have at least one field as Primary Key.

Relationship between Tables

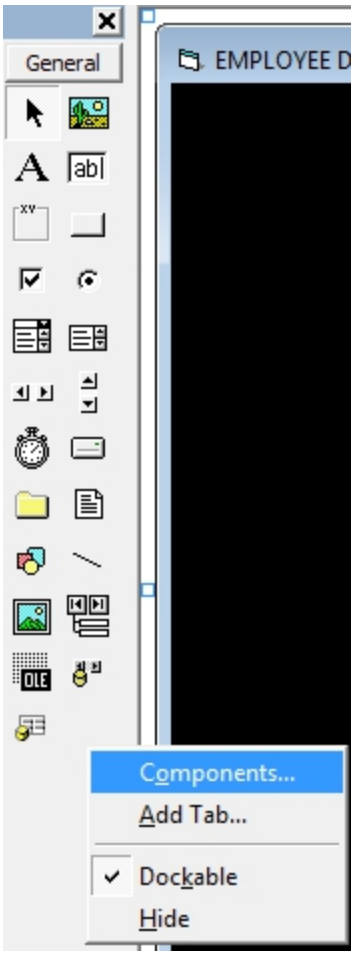


Relationship Between Tables

Accessing Access Database from Visual Basic 6

You can use a common module from visual basic to connect to the access database. Functions from this module will be called from all Forms in the visual basic project, so that we do not need to rewrite the same code for database connectivity.

Add the ADODB components in the VB Toolbox.

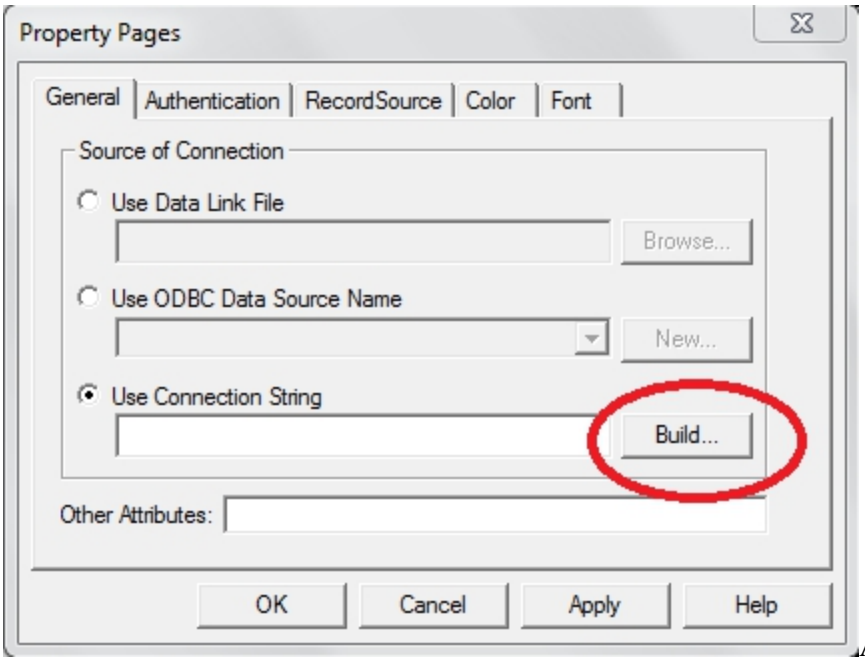


Add ADOBD component

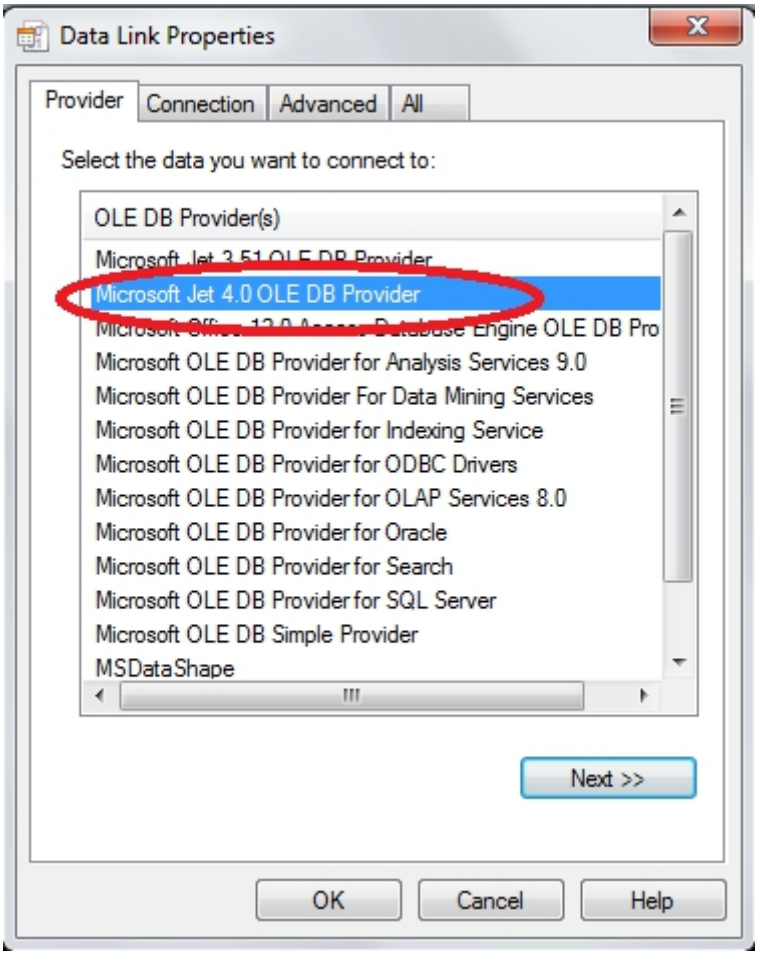
Add an **Adodb component**, [Right Click] on the **Control Object** > Click **Properties**.

ADODB Component

Under the **Adodc properties** windows > Click **Build**.

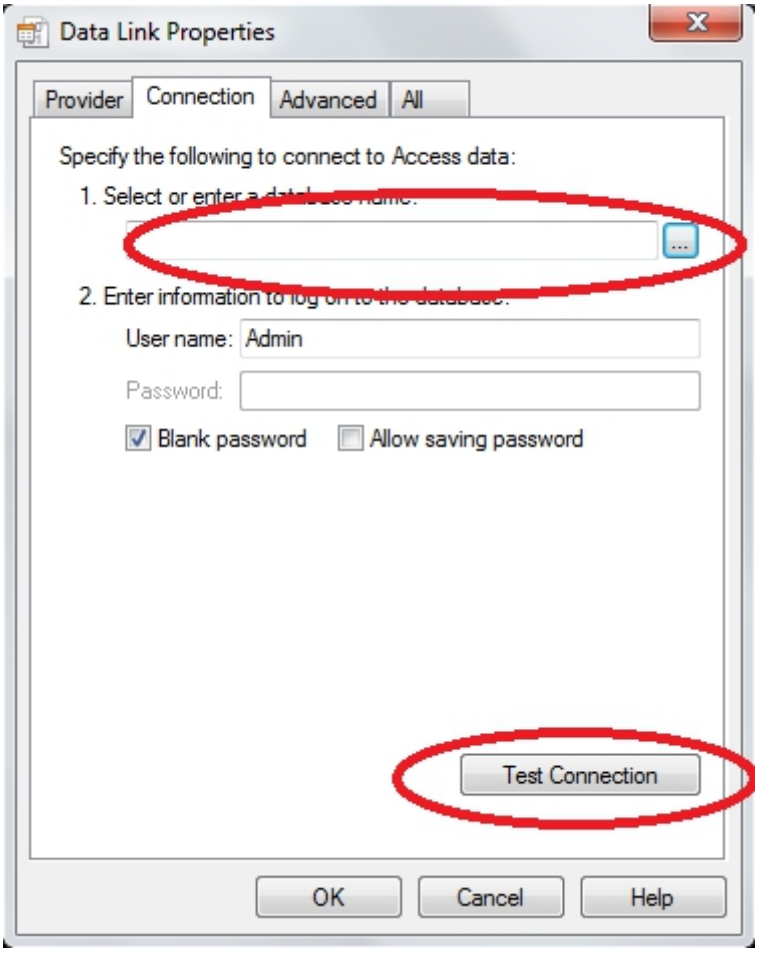


ADODC Properties



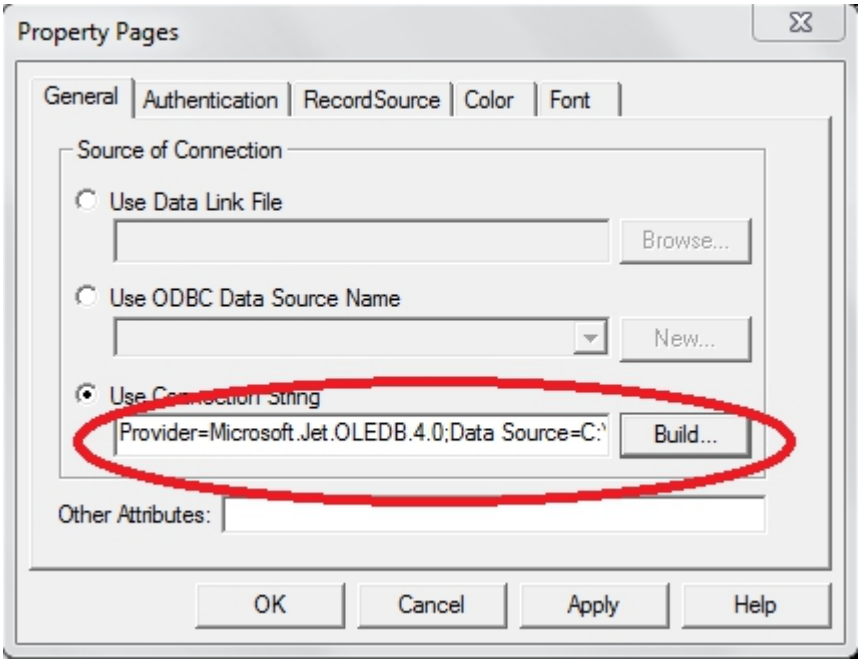
Select the Provider

Under the **Connection** tab > Select the **Database** you created previously.



Select the Database

Now Click **"Test Connection"** to check if the database connection is successful. Click on the "**Provider**" tab again and we now have our *connection string*.

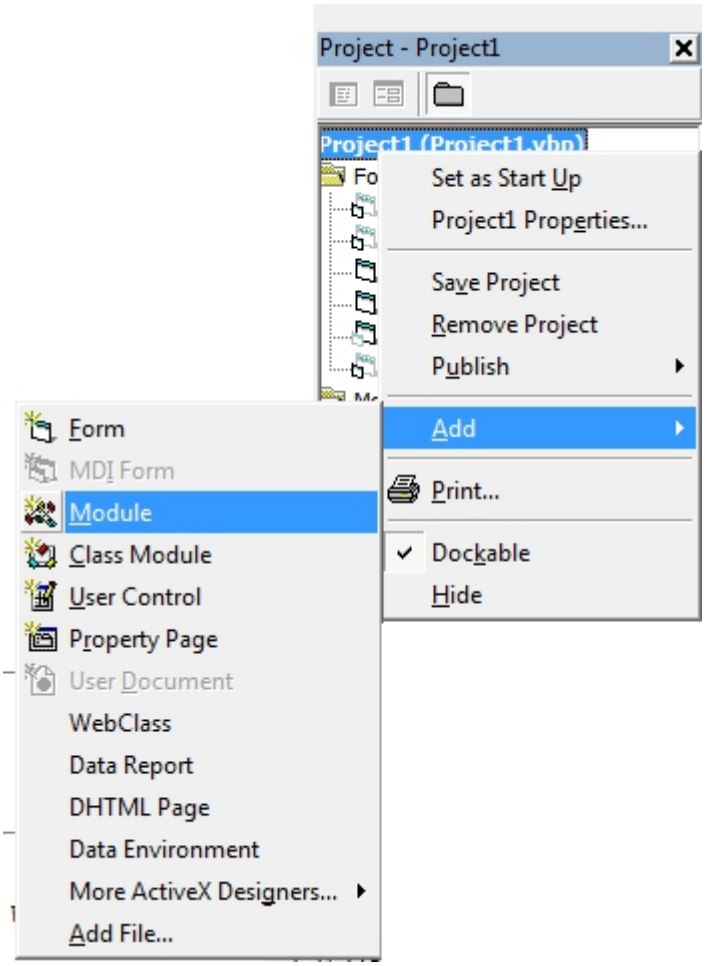


Connection String

This new Connection String will be used in Module.Bas file for accessing the database at run-time. The existing ADODC control is no longer required and you can delete it.

Creating Module1 for Database Connectivity

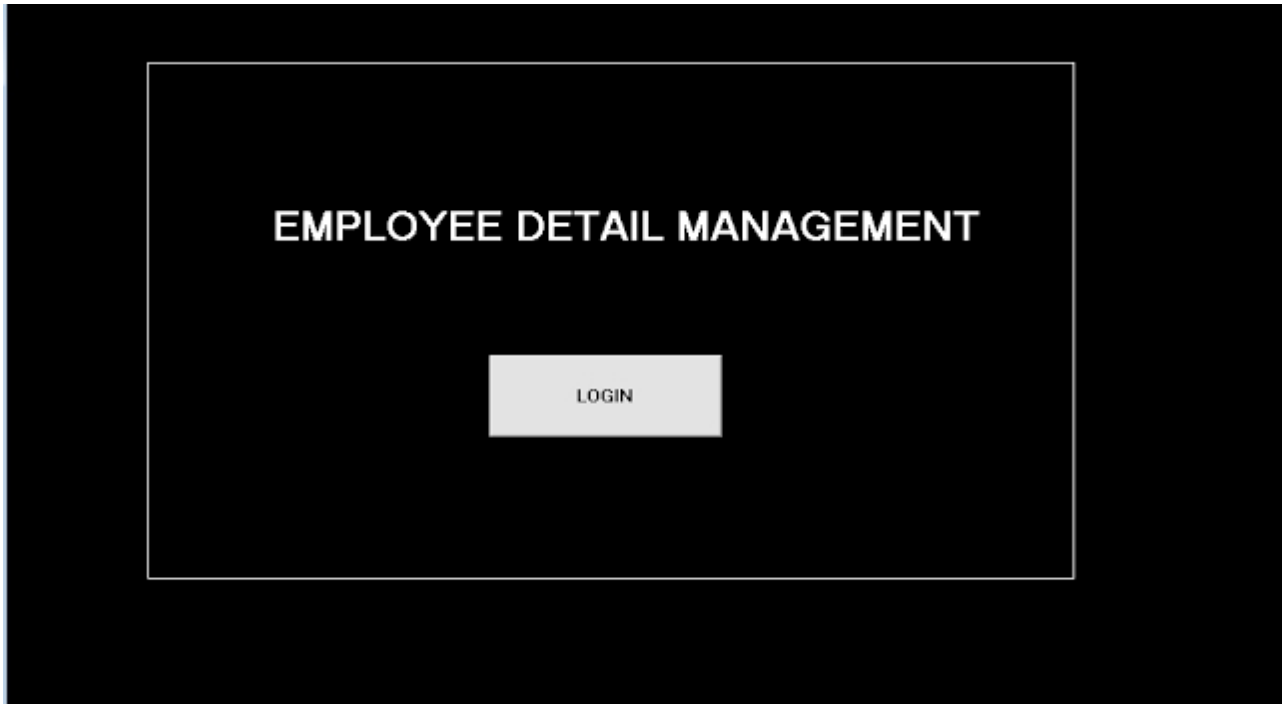
To create a new database connection at run-time, do the following.
Under **Project.**vbp windows > **Project1** [Right-
Click] > **Add** [Select] > **Module** [Click].
A new module will open as shown in the following figure.



Create a New Module

Type the following code in the Module Code Editor.

```
Module Connection

Public con As New adodb.Connection
Public rs As New adodb.Recordset
Public constr As String

Public Sub loadcon()
    constr = "Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=C:\Users\Peter\Documents\emp.mdb;Persist Security Info=False"

    con.Open constr
    MsgBox ("connected")

End Sub
```



FORM EMPLOYEE DETAILS MANAGEMENT

Labels
Name: lblEMPMANAGE

Caption: EMPLOYEE RECORD MANAGEMENT

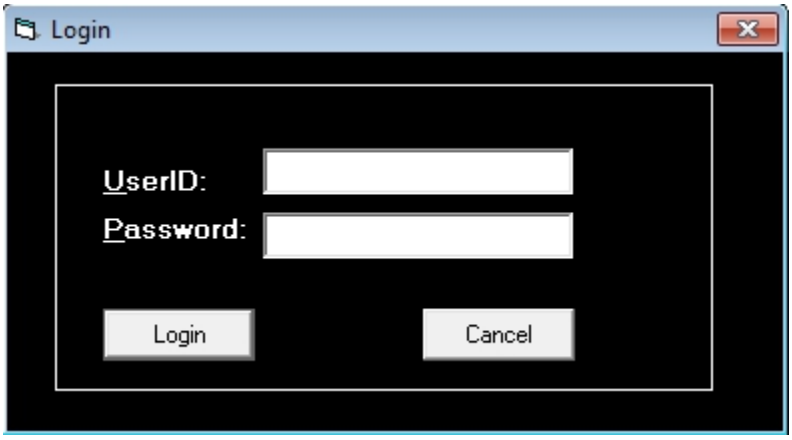
Buttons
Name: cmdLogin

Caption: Login

Code for frmEMPMANAGE

```
Login Button

Private Sub cmdLogin_Click()

    Unload Me
    frmLogin.Show

End Sub
```

Components of frmLogin



Form Login

Labels
Name: lblUserID

Caption:UserID

Name: lblPassword

Caption: Password

Textboxes
Name: txtUserID

Text: ‘leave blank’

Name: txtPassword

Text; ‘leave blank’

Buttons
Name; vbOK

Caption: Login

Name: vbCancel

Caption: Cancel

Code for frmLogin

```
Form Login

Option Explicit
Public LoginSucceeded As Boolean

Private Sub cmdCancel_Click()
    'set the global var to false
    'to denote a failed login
    LoginSucceeded = False
    Me.Hide
End Sub

Private Sub cmdOK_Click()
    With
        rs .Open "SELECT * FROM LOGIN", con, adOpenDynamic,
adLockOptimistic

        'check the username

        .MoveFirst While Not .EOF

    If txtUserID = !UserID And txtPassword = !Password Then
        frmMenu.Show
        LoginSucceeded = True
        con.Close
        MsgBox ("Connection Closed")
        Exit Sub
        Unload Me
    ElseIf
        txtUserID <> !UserID Then
        .MoveNext
    Else
        MsgBox "Invalid Password, try again!", , "Login"
        txtPassword.SetFocus SendKeys "{Home}+{End}"
    Exit Sub
    End If
    Wend
    .Close
    End With

End Sub
```

```
Private Sub Form_Load()
    Call loadcon
    Unload frmEMPMANAGE
End Sub
```

Components for frmEmpDetails

Form Employee Details

Labels

Name: lblEmpID

Caption: EMPID

Name: lblFirstname

Caption: FirstName

Name: lblLastname

Caption: LastName

Name: lblHRA

Caption: HRA

Name: lblDA

Caption: DA

Name: lblBasic

Caption: Basic

TextBoxes
Name: txtEmpID

Text: ‘leave blank’

Name: txtFirstname

Text: ‘leave blank’

Name: txtLastname

Text: ‘leave blank’

Name: txtHRA

Text: ‘leave blank’

Name: txtDA

Text: ‘leave blank’

Name: txtBasic

Text: ‘leave blank’

Buttons
Name: cmdAdd

Caption: &ADD

Name: cmdDel

Caption: &DELETE

Name: cmdUpdate

Caption: &Update

Name: cmdClr

Caption: &Clear

Name: cmdExit

Caption: &Exit

Name: cmdFirstname

Caption: &FirstName

Name: cmdLastname

Capttion: &LastName

Name: cmdNext

Caption: Next Record

Name: cmdPrev

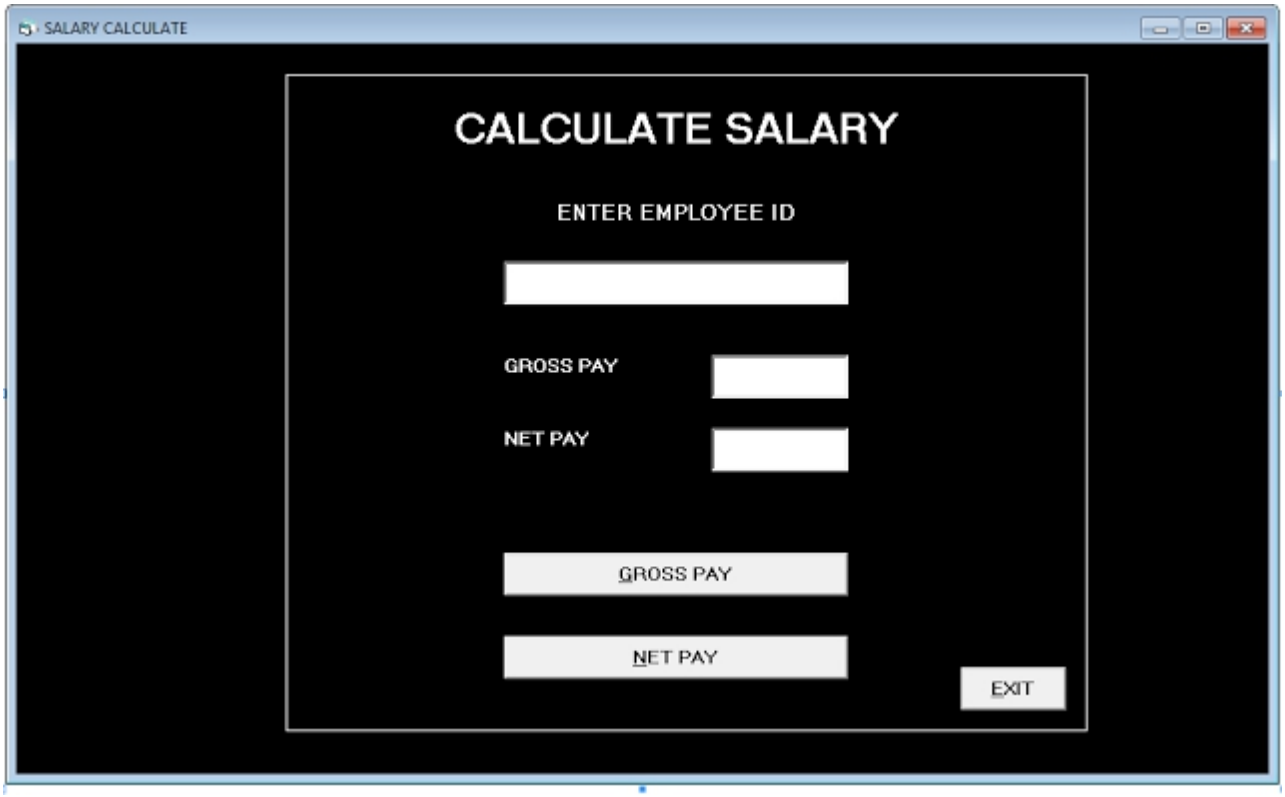
Caption: Previous Record

Code for frmEmpDetails

```
Form Employee Details

Private Sub cmdAdd_Click()
    con.Execute "INSERT INTO emp(EmpID,FirstName,
    LastName,Basic, HRA, DA)
    VALUES (" &   txtEmpID & ",'" &
    txtFirstname & "','" &
    txtLastname & "'," &
    txtBasic & "," &
    txtHRA & "," &
    txtDA & " );"

    MsgBox ("Record Inserted")
End Sub

Private Sub cmdCLR_Click()

    txtEmpID = ""
    txtFirstname = ""
    txtLastname = ""
    txtBasic = ""
    txtHRA = ""
    txtDA = ""

End Sub

Private Sub cmdDel_Click()

    con.Execute "DELETE * FROM emp
    WHERE EmpID = " &
    txtEmpID & ""

    MsgBox ("Record Deleted")
    txtEmpID = ""

End Sub

Private Sub cmdExit_Click()
    Unload Me
    con.Close
End Sub
```

```
Private Sub cmdUpdate_Click()
    con.Execute "UPDATE emp SET FirstName = '" &
    txtFirstname & "', LastName = ' " &
    txtLastname & " ' , Basic = " &
    txtBasic & " , HRA = " &
    txtHRA & " , DA = " &
    txtDA & " WHERE EmpID = " &
    txtEmpID.Text & ""
End Sub

Private Sub Form_Load()
    Call loadcon
End Sub
```

Component **for** frmSalary



Form Salary – Employee Record Management

Labels

Name: lblCalculateS

Option: CALCULATE SALARY

Name: lblEmpID

Caption: ENTER EMPLOYEE ID

Name: lblGrossP

Caption: GROSS PAY

Name: lblNetP

Caption: NET PAY

Textboxes

Name: txtEmpID

Text: ‘leave blank’

Name: txtGross

Text: ‘leave blank’

Name: txtNet

Text: ‘leave blank’

Buttons
Name : cmdGross

Caption: GROSS PAY

Name: cmdNet

Caption: NET PAY

Name: cmdExit

Caption: EXIT

Code for frmSalary

```
Form Salary

Private Sub cmdExit_Click()
    Unload Me
    con.Close
End Sub

Private Sub cmdGross_Click()
Dim gross As Double
Dim net As Double
Dim tax As Double
    With
```

```
        rs rs.Open "SELECT * FROM emp WHERE EmpID = " &
        txtEmpID & "", con, adOpenDynamic, adLockPessimistic
        gross = !Basic + !HRA + !DA
    End With
        txtGross.Text = gross

    con.Execute "UPDATE emp SET GrossPay =  ' " &
    gross & " ' WHERE EmpID = " &
    txtEmpID & ""

    rs.Close
End Sub

Private Sub cmdNet_Click()
Dim gross As Double
Dim net As Double
Dim tax As Double
    With rs
        rs.Open "SELECT * FROM emp
        WHERE EmpID = " &
        txtEmpID & "",
        con, adOpenDynamic, adLockPessimistic

        gross = !Basic + !HRA + !DA
        tax = gross * 10 / 100
        net = gross - tax
    End With
    rs.Close
    txtNet.Text = net
    con.Execute "UPDATE emp
    SET NetPay =  ' " & net & " '
    WHERE EmpID = " & txtEmpID & ""

End Sub

Private Sub Form_Load()
    Call loadcon
End Sub
```

Components for frmDependent

Form Dependent – Employee Record Management System

Labels

Name: lblDID

Caption: Department ID

Name: lblDname

Caption: Department Name

Name: lblRelation

Caption: Relation

Name: lblDependent

Caption: Employee Dependent

Textboxes
Name: txtDID

Text: ‘leave blank’

Name: txtDname

Text: ‘leave blank’

Name: txtRel

Text: ‘leave blank’

Name: txtEmpID

Text: ‘leave blank’

Buttons
Name; cmdAdd

Caption: &ADD

Name: cmdDel

Caption: &DELETE

Name: cmdUpdate

Caption: &UPDATE

Name: cmdClr

Caption: &CLEAR

Name: cmdExit

Caption: &EXIT

Code for frmDependent

```
Form Dependent

Private Sub cmdAdd_Click()
    con.Execute "INSERT INTO
    Dependent(DependentID, DependentName,Relation,EmpID )
    VALUES (" & txtDID & ", '" &
    txtDName & "','" &
    txtRel & "'," &
    txtEmpID & ");"

    txtDID.Text = ""
    txtDName.Text = ""
    txtRel.Text = ""
    txtEmpID = ""

    con.Execute "DELETE * FROM Dependent
    WHERE DependentID = " & txtDID & ""
End Sub

Private Sub cmdClr_Click()
    txtDID.Text = ""
    txtDName.Text = ""
    txtRel.Text = ""
    txtEmpID = ""
End Sub

Private Sub cmdExit_Click()
    Unload Me
End Sub

Private Sub cmdUpdate_Click()

    con.Execute "UPDATE Dependent
    SET  DependentName = ' " &
    txtDName & " ' , Relation = ' " &
    txtRel & " ' WHERE DependentID = " &
    txtDID & " "
    MsgBox ("Record Updated")

End Sub
```

```
Private Sub Form_Load()
    Call loadcon
End Sub
```

frmMenu [MDI Form]

MDI Form are Multiple-Document Interface in which you can open multiple forms. All our previously created will be child form of MDI.

This form is called **frmMENU.**
This form has menu editor to add different menu items and each menu item is linked to one of the forms created earlier. Before we get started with MDI form, you must set the **MDI Child** property of following Forms to 'True'.
frmEmpDetails
frmSalary
frmDependent

Create a New MDI Form

To create a new MDI From, go to **Project1.vbp > [**Right Click] and then [Select] **Add** > [Click] **MDI Form.**
Right Click the form and you will see the **Menu Editor,** using the editor create following menu items.
Caption : Employee Details

Name : employee

Index : 0

Caption : Salary Calculation

Name : salary

Index : 1

Caption : Dependents

Name : dependents

Index : 2

Caption : Exit

Name : exit

Index : 3

Code for frmMENU

The code for frmMENU is simple, you click on one of the menu item and a new form opens up, however the new form does not open in a separate windows, but it will open within frmMENU as MDI Child form.

```
Form Menu

Private Sub depend_Click(Index As Integer)
    frmDependent.Show
End Sub

Private Sub empdetails_Click(Index As Integer)
    frmEmpDetails.Show
End Sub

Private Sub exit_Click(Index As Integer)
    Unload Me
    Unload frmLogin
End Sub

Private Sub salary_Click(Index As Integer)
    frmSalary.Show
End Sub
```

Title: PAYROLL PROCESSING SYSTEM

PROJECT

2019- 2020

Submitted to

SKE Society's

RANI PARVATI DEVI COLLEGE

OF ARTS & COMMERCE

Tilakwadi, Belagavi.

By

Name:_____

Class: _____ Roll no: _____

Date of Submission: _____

Under the guidance of the department of COMPUTER APPLICATION

Student's signature Staff Signature

S.K.E.SOCIETY'S

RANI PARVATI DEVI COLLEGE OF ARTS AND COMMERCE
BELAGAVI

DEPARTMENT OF COMPUTER APPLICATION

Certificate

Roll No. _____ Date: _____

Examination Seat No. _____

This is to certify that Mr./Ms._____ has successfully carried out the course of exercises in practical Computer Application prescribed by the Rani Channamma University, Belagavi for B.A. ____ semester examination in the Computer laboratory of his college during the year 2018-19.

Staff member in-charge **Head**
Dept. of Computer

External Examiner: _____

Internal Examiner: _____

Employees Record Management using VB 6.0 and MS Access Database

In this project, we will learn to create an application in Visual Basic 6.0 for Employee Record Management in an organization.

Contents hide

1 System Model
2 E-R Diagram for Employee Record Management
3 Working with MS Access Database
4 Accessing Access Database from Visual Basic 6
5 Creating Module1 for Database Connectivity
6 Code for frmEMPMANAGE
7 Components of frmLogin
8 Code for frmLogin
9 Components for frmEmpDetails
10 Code for frmEmpDetails
11 Component for frmSalary
12 Code for frmSalary
13 Components for frmDependent
14 Code for frmDependent
15 frmMenu [MDI Form]
16 Create a New MDI Form
17 Code for frmMENU

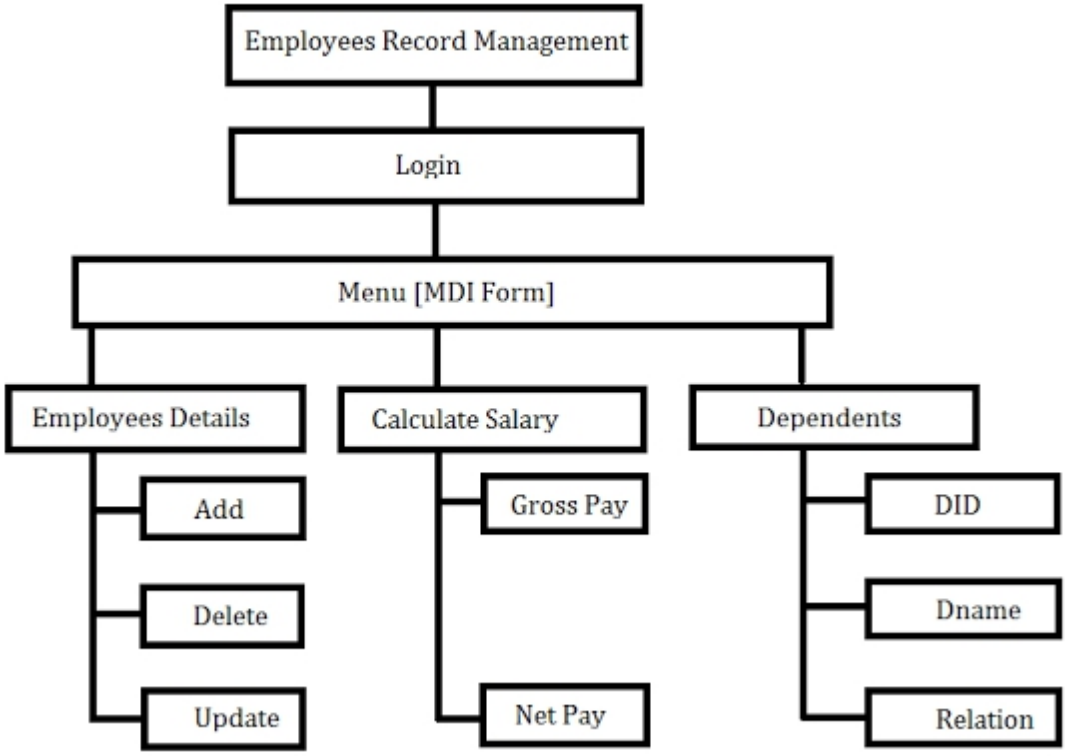
The application does following tasks

1. Insert, Delete, Update employee records.
2. Compute the Gross Pay and Net Pay of each employee.
3. Maintain records of departments for each employee.
4. Each employee should be able to login to the system with own password.

The development of the application – Employees Record Management is done in following ways.

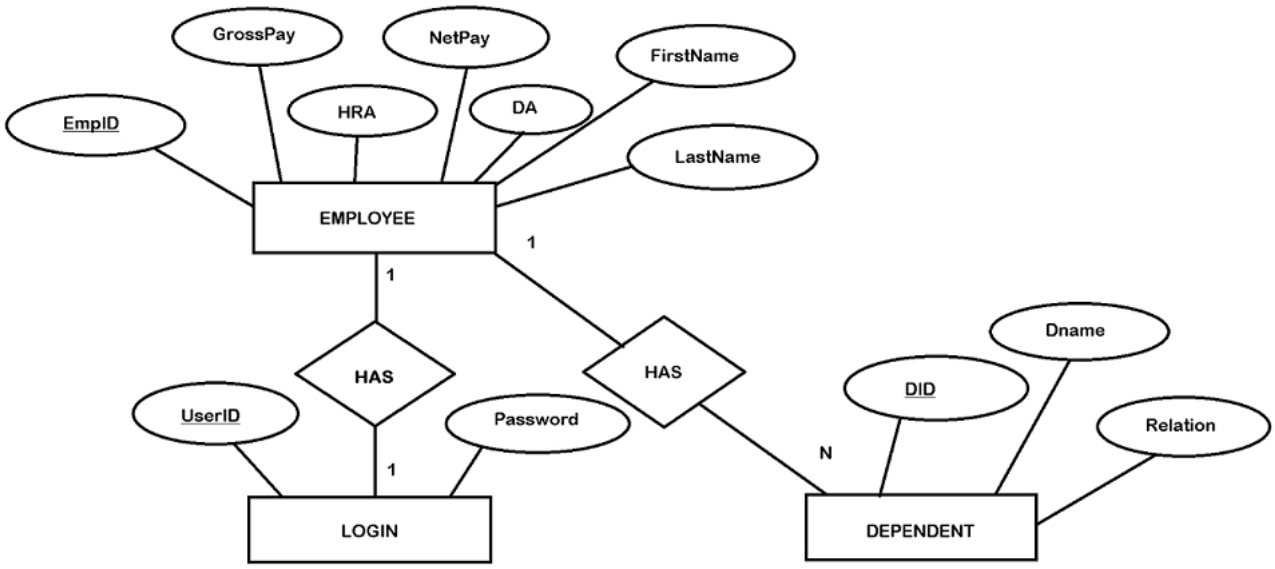
- Develop System Model.
- Define the Data Model (E-R Diagram)
- Design the User Interface in VB 6.
- Rename the components.
- Write codes for components.
- Test and verify Output.

System Model



System Model – Employee Record Management

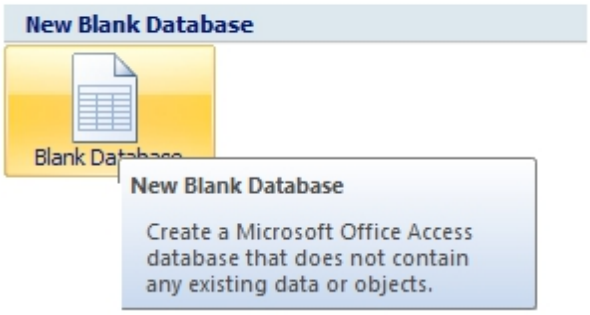
E-R Diagram for Employee Record Management



E-R Diagram – Employee Record Management

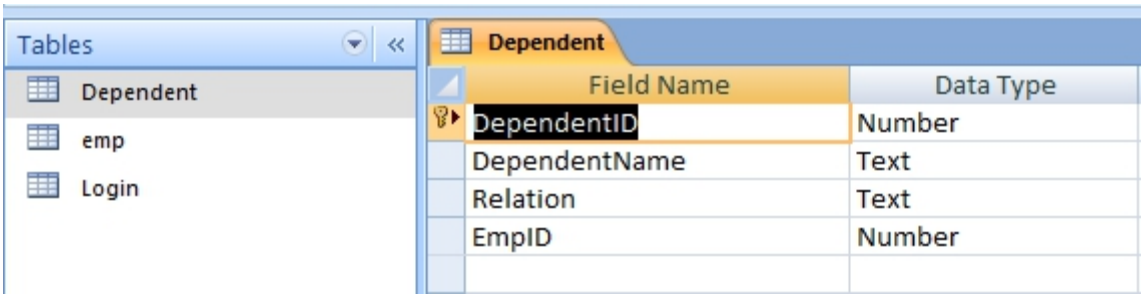
Working with MS Access Database

Based on the E-R Model you can create a New Access Database and save it as the **emp.mdb** file. To create a blank database
Open Microsoft Office Access 2007 (Check MS office documentation for other versions) > Click **"Blank Database".**



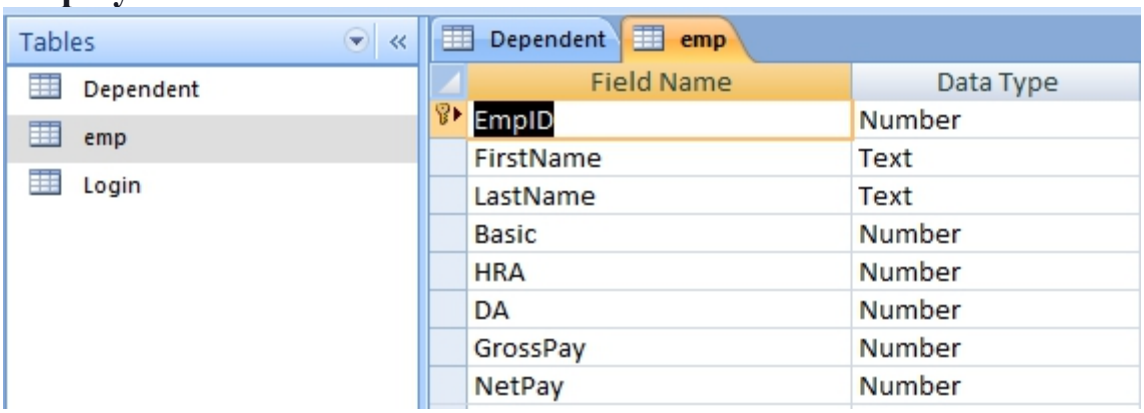
Click Blank Database

Create three table for the **emp.mdb** database.
Dependent Table Schema



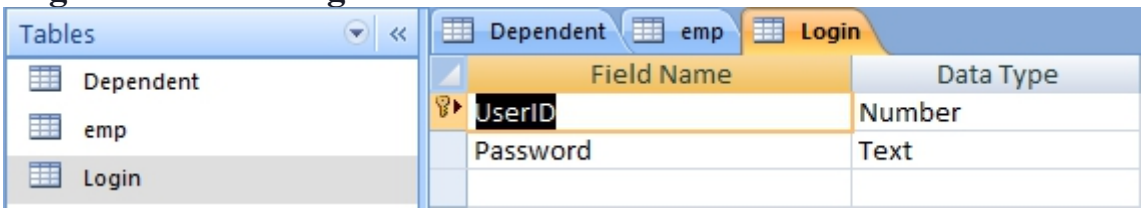
Dependent Table in Design View

Employee Table Schema



Employee Table in Design View

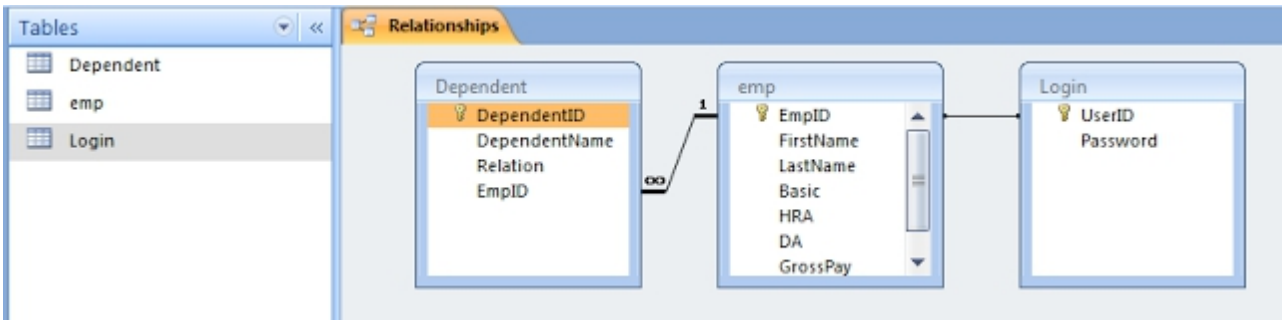
Login Table in Design View



Login Table in Design View

Note: Each Table should have at least one field as Primary Key.

Relationship between Tables

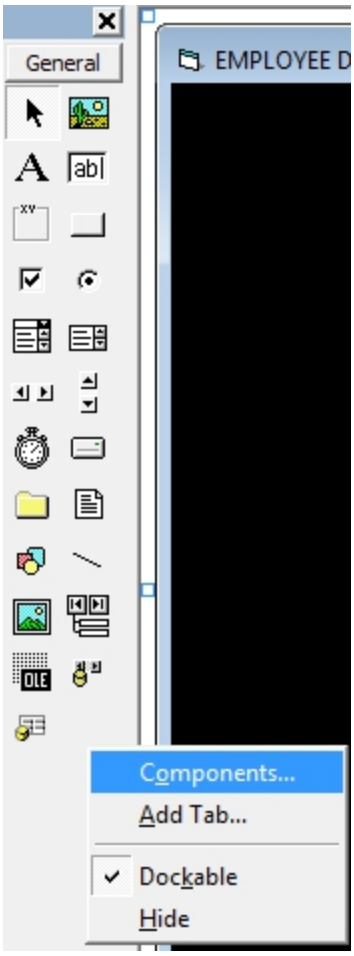


Relationship Between Tables

Accessing Access Database from Visual Basic 6

You can use a common module from visual basic to connect to the access database. Functions from this module will be called from all Forms in the visual basic project, so that we do not need to rewrite the same code for database connectivity.

Add the ADODB components in the VB Toolbox.

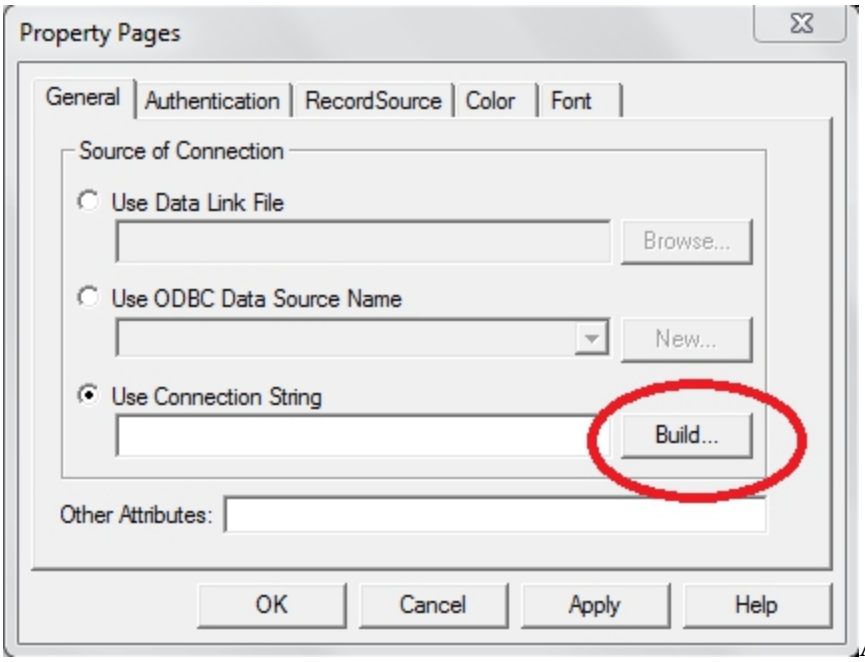


Add ADOBD component

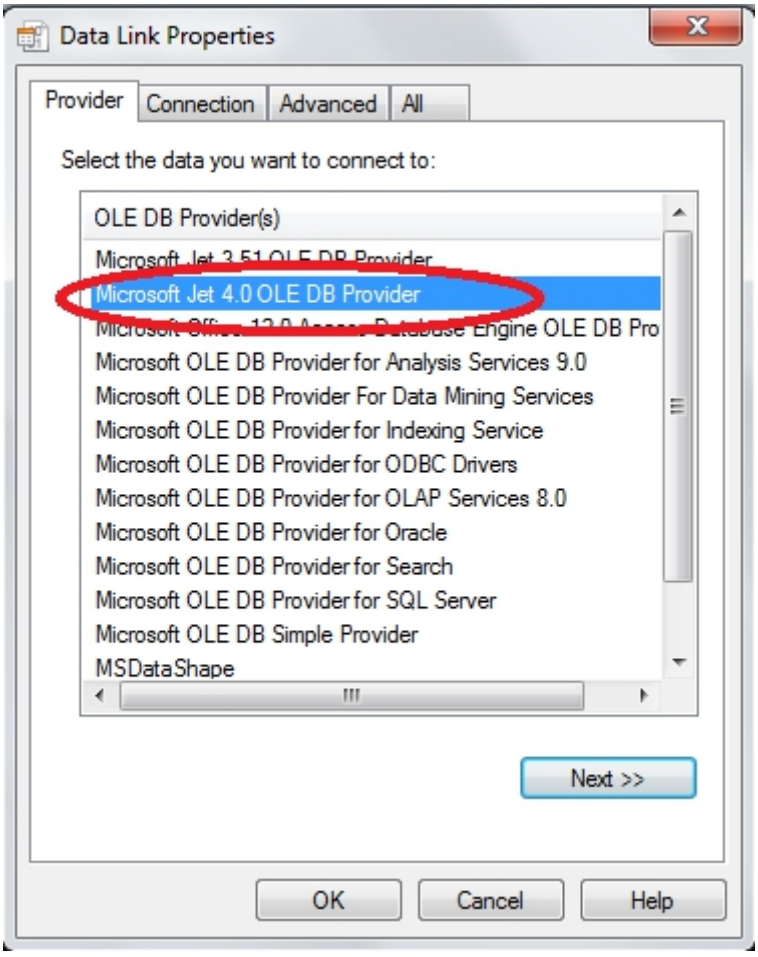
Add an **Adodb component**, [Right Click] on the **Control Object** > Click **Properties**.

ADODB Component

Under the **Adodc properties** windows > Click **Build**.

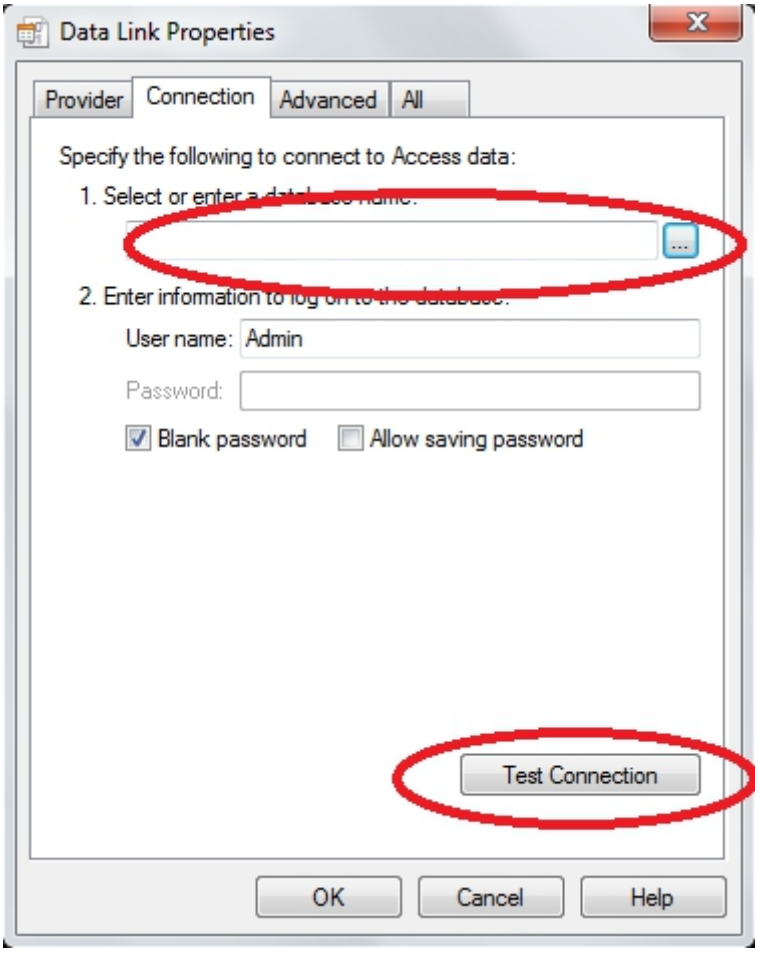


ADODC Properties



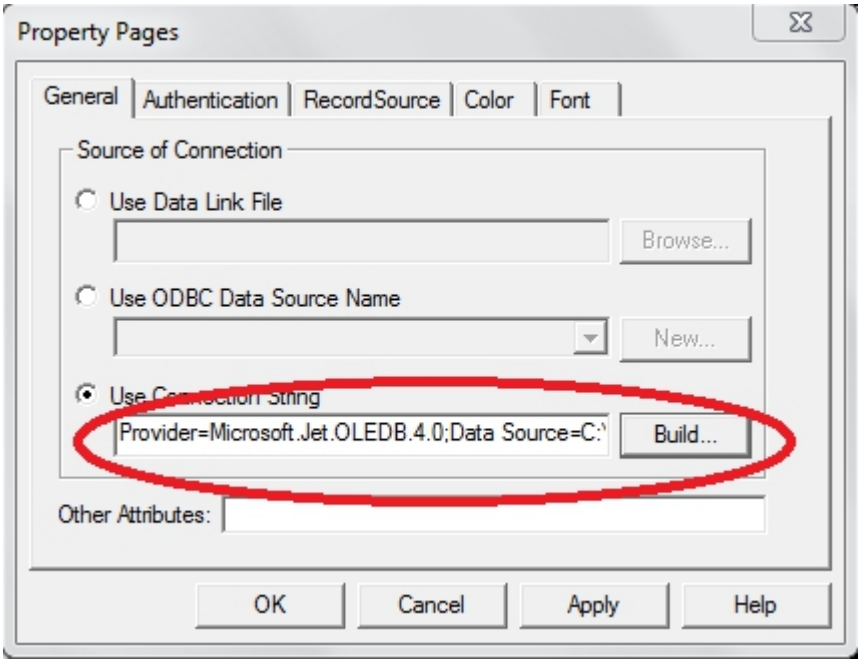
Select the Provider

Under the **Connection** tab > Select the **Database** you created previously.



Select the Database

Now Click **"Test Connection"** to check if the database connection is successful. Click on the "**Provider**" tab again and we now have our *connection string*.

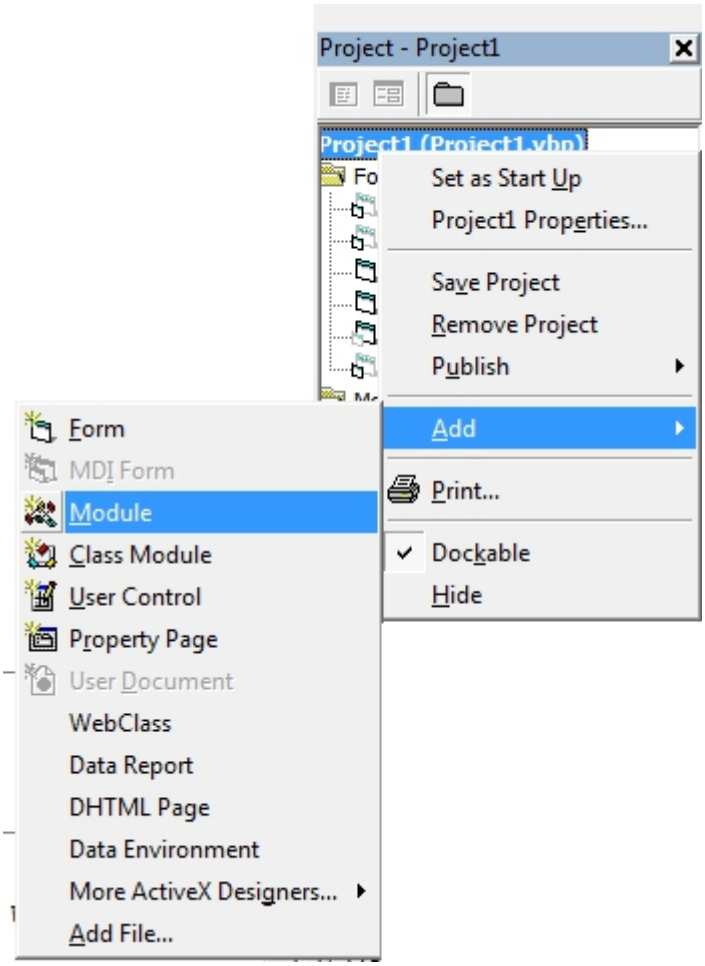


Connection String

This new Connection String will be used in Module.Bas file for accessing the database at run-time. The existing ADODC control is no longer required and you can delete it.

Creating Module1 for Database Connectivity

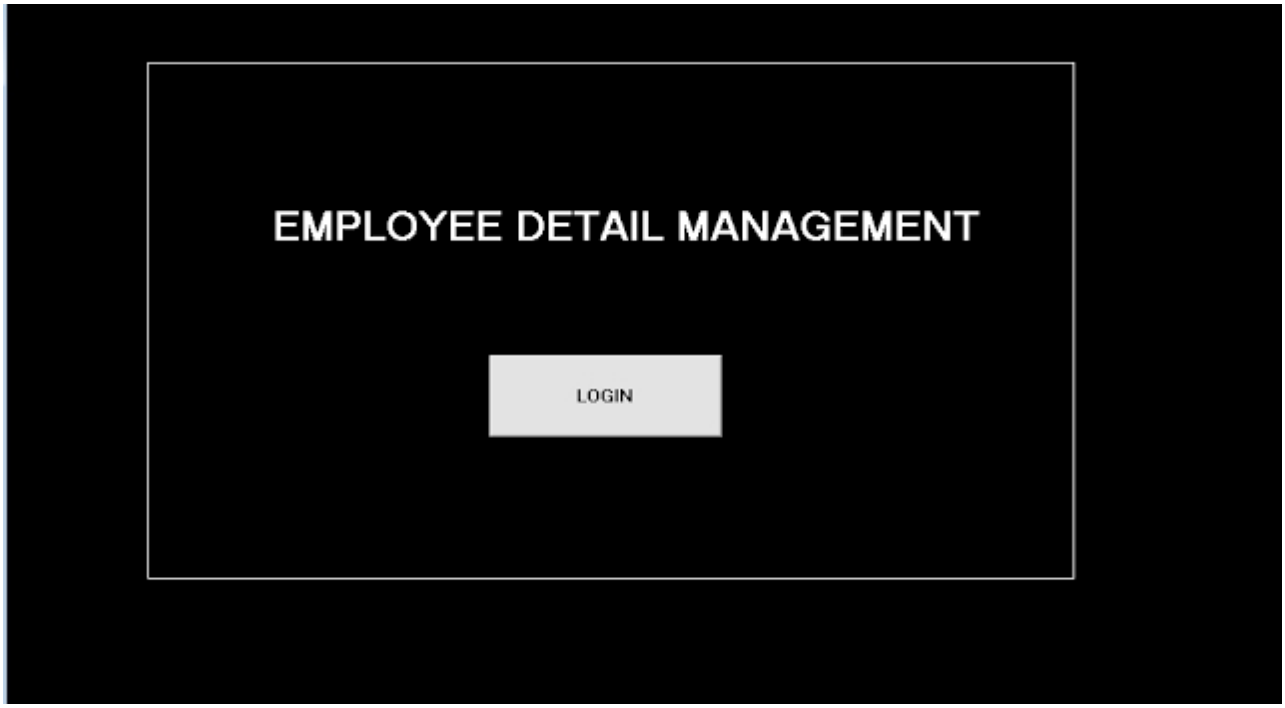
To create a new database connection at run-time, do the following.
Under **Project.**vbp windows > **Project1** [Right-
Click] > **Add** [Select] > **Module** [Click].
A new module will open as shown in the following figure.



Create a New Module

Type the following code in the Module Code Editor.

```
Module Connection

Public con As New adodb.Connection
Public rs As New adodb.Recordset
Public constr As String

Public Sub loadcon()
    constr = "Provider=Microsoft.Jet.OLEDB.4.0;Data
Source=C:\Users\Peter\Documents\emp.mdb;Persist Security Info=False"

    con.Open constr
    MsgBox ("connected")

End Sub
```



FORM EMPLOYEE DETAILS MANAGEMENT

Labels
Name: lblEMPMANAGE

Caption: EMPLOYEE RECORD MANAGEMENT

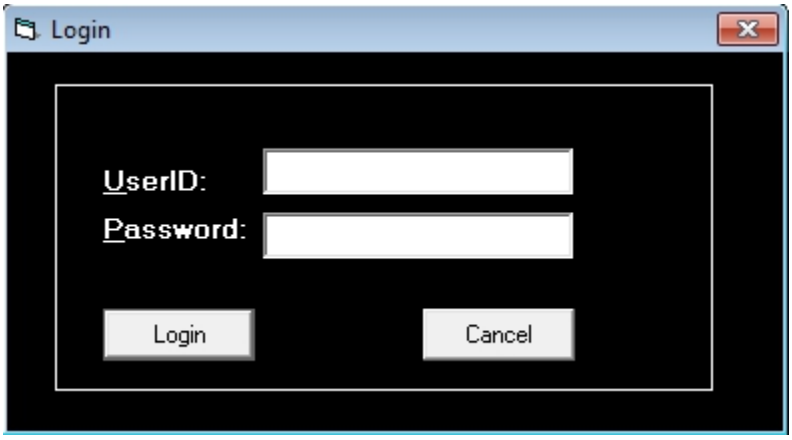
Buttons
Name: cmdLogin

Caption: Login

Code for frmEMPMANAGE

```
Login Button

Private Sub cmdLogin_Click()

    Unload Me
    frmLogin.Show

End Sub
```

Components of frmLogin



Form Login

Labels
Name: lblUserID

Caption:UserID

Name: lblPassword

Caption: Password

Textboxes
Name: txtUserID

Text: ‘leave blank’

Name: txtPassword

Text; ‘leave blank’

Buttons
Name; vbOK

Caption: Login

Name: vbCancel

Caption: Cancel

Code for frmLogin

```
Form Login

Option Explicit
Public LoginSucceeded As Boolean

Private Sub cmdCancel_Click()
    'set the global var to false
    'to denote a failed login
    LoginSucceeded = False
    Me.Hide
End Sub

Private Sub cmdOK_Click()
    With
        rs .Open "SELECT * FROM LOGIN", con, adOpenDynamic,
adLockOptimistic

        'check the username

        .MoveFirst While Not .EOF

    If txtUserID = !UserID And txtPassword = !Password Then
        frmMenu.Show
        LoginSucceeded = True
        con.Close
        MsgBox ("Connection Closed")
        Exit Sub
        Unload Me
    ElseIf
        txtUserID <> !UserID Then
        .MoveNext
    Else
        MsgBox "Invalid Password, try again!", , "Login"
        txtPassword.SetFocus SendKeys "{Home}+{End}"
    Exit Sub
    End If
    Wend
    .Close
    End With

End Sub
```

```
Private Sub Form_Load()
    Call loadcon
    Unload frmEMPMANAGE
End Sub
```

Components for frmEmpDetails

Form Employee Details

Labels

Name: lblEmpID

Caption: EMPID

Name: lblFirstname

Caption: FirstName

Name: lblLastname

Caption: LastName

Name: lblHRA

Caption: HRA

Name: lblDA

Caption: DA

Name: lblBasic

Caption: Basic

TextBoxes
Name: txtEmpID

Text: ‘leave blank’

Name: txtFirstname

Text: ‘leave blank’

Name: txtLastname

Text: ‘leave blank’

Name: txtHRA

Text: ‘leave blank’

Name: txtDA

Text: ‘leave blank’

Name: txtBasic

Text: ‘leave blank’

Buttons
Name: cmdAdd

Caption: &ADD

Name: cmdDel

Caption: &DELETE

Name: cmdUpdate

Caption: &Update

Name: cmdClr

Caption: &Clear

Name: cmdExit

Caption: &Exit

Name: cmdFirstname

Caption: &FirstName

Name: cmdLastname

Capttion: &LastName

Name: cmdNext

Caption: Next Record

Name: cmdPrev

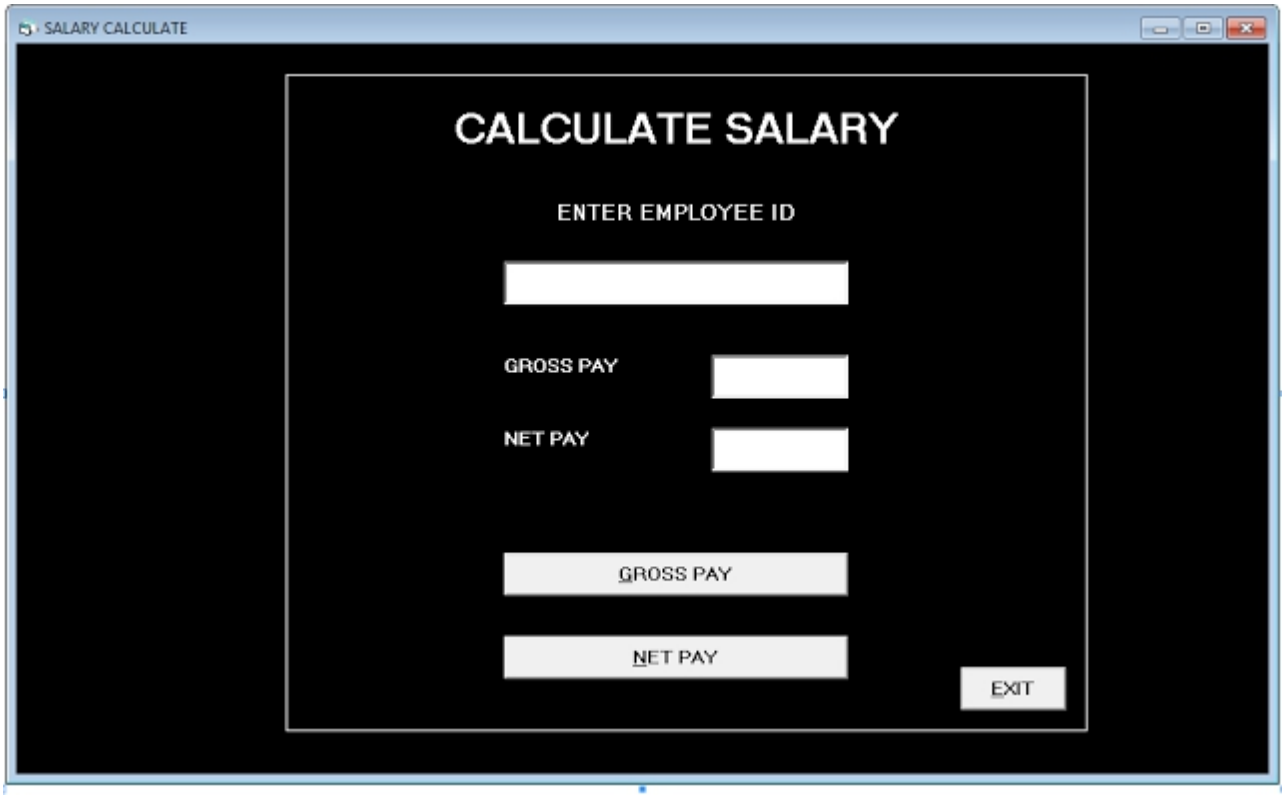
Caption: Previous Record

Code for frmEmpDetails

```
Form Employee Details

Private Sub cmdAdd_Click()
    con.Execute "INSERT INTO emp(EmpID,FirstName,
    LastName,Basic, HRA, DA)
    VALUES (" &   txtEmpID & ",'" &
    txtFirstname & "','" &
    txtLastname & "'," &
    txtBasic & "," &
    txtHRA & "," &
    txtDA & " );"

    MsgBox ("Record Inserted")
End Sub

Private Sub cmdCLR_Click()

    txtEmpID = ""
    txtFirstname = ""
    txtLastname = ""
    txtBasic = ""
    txtHRA = ""
    txtDA = ""

End Sub

Private Sub cmdDel_Click()

    con.Execute "DELETE * FROM emp
    WHERE EmpID = " &
    txtEmpID & ""

    MsgBox ("Record Deleted")
    txtEmpID = ""

End Sub

Private Sub cmdExit_Click()
    Unload Me
    con.Close
End Sub
```

```
Private Sub cmdUpdate_Click()
    con.Execute "UPDATE emp SET FirstName = '" &
    txtFirstname & "', LastName = ' " &
    txtLastname & " ' , Basic = " &
    txtBasic & " , HRA = " &
    txtHRA & " , DA = " &
    txtDA & " WHERE EmpID = " &
    txtEmpID.Text & ""
End Sub

Private Sub Form_Load()
    Call loadcon
End Sub
```

Component **for** frmSalary



Form Salary – Employee Record Management

Labels

Name: lblCalculateS

Option: CALCULATE SALARY

Name: lblEmpID

Caption: ENTER EMPLOYEE ID

Name: lblGrossP

Caption: GROSS PAY

Name: lblNetP

Caption: NET PAY

Textboxes

Name: txtEmpID

Text: ‘leave blank’

Name: txtGross

Text: ‘leave blank’

Name: txtNet

Text: ‘leave blank’

Buttons
Name : cmdGross

Caption: GROSS PAY

Name: cmdNet

Caption: NET PAY

Name: cmdExit

Caption: EXIT

Code for frmSalary

```
Form Salary

Private Sub cmdExit_Click()
    Unload Me
    con.Close
End Sub

Private Sub cmdGross_Click()
Dim gross As Double
Dim net As Double
Dim tax As Double
    With
```

```
        rs rs.Open "SELECT * FROM emp WHERE EmpID = " &
        txtEmpID & "", con, adOpenDynamic, adLockPessimistic
        gross = !Basic + !HRA + !DA
    End With
        txtGross.Text = gross

    con.Execute "UPDATE emp SET GrossPay =  ' " &
    gross & " ' WHERE EmpID = " &
    txtEmpID & ""

    rs.Close
End Sub

Private Sub cmdNet_Click()
Dim gross As Double
Dim net As Double
Dim tax As Double
    With rs
        rs.Open "SELECT * FROM emp
        WHERE EmpID = " &
        txtEmpID & "",
        con, adOpenDynamic, adLockPessimistic

        gross = !Basic + !HRA + !DA
        tax = gross * 10 / 100
        net = gross - tax
    End With
    rs.Close
    txtNet.Text = net
    con.Execute "UPDATE emp
    SET NetPay =  ' " & net & " '
    WHERE EmpID = " & txtEmpID & ""

End Sub

Private Sub Form_Load()
    Call loadcon
End Sub
```

Components for frmDependent

Form Dependent – Employee Record Management System

Labels

Name: lblDID

Caption: Department ID

Name: lblDname

Caption: Department Name

Name: lblRelation

Caption: Relation

Name: lblDependent

Caption: Employee Dependent

Textboxes
Name: txtDID

Text: ‘leave blank’

Name: txtDname

Text: ‘leave blank’

Name: txtRel

Text: ‘leave blank’

Name: txtEmpID

Text: ‘leave blank’

Buttons
Name; cmdAdd

Caption: &ADD

Name: cmdDel

Caption: &DELETE

Name: cmdUpdate

Caption: &UPDATE

Name: cmdClr

Caption: &CLEAR

Name: cmdExit

Caption: &EXIT

Code for frmDependent

```
Form Dependent

Private Sub cmdAdd_Click()
    con.Execute "INSERT INTO
    Dependent(DependentID, DependentName,Relation,EmpID )
    VALUES (" & txtDID & ", '" &
    txtDName & "','" &
    txtRel & "'," &
    txtEmpID & ");"

    txtDID.Text = ""
    txtDName.Text = ""
    txtRel.Text = ""
    txtEmpID = ""

    con.Execute "DELETE * FROM Dependent
    WHERE DependentID = " & txtDID & ""
End Sub

Private Sub cmdClr_Click()
    txtDID.Text = ""
    txtDName.Text = ""
    txtRel.Text = ""
    txtEmpID = ""
End Sub

Private Sub cmdExit_Click()
    Unload Me
End Sub

Private Sub cmdUpdate_Click()

    con.Execute "UPDATE Dependent
    SET  DependentName = ' " &
    txtDName & " ' , Relation = ' " &
    txtRel & " ' WHERE DependentID = " &
    txtDID & " "
    MsgBox ("Record Updated")

End Sub
```

```
Private Sub Form_Load()
    Call loadcon
End Sub
```

frmMenu [MDI Form]

MDI Form are Multiple-Document Interface in which you can open multiple forms. All our previously created will be child form of MDI.

This form is called **frmMENU.**
This form has menu editor to add different menu items and each menu item is linked to one of the forms created earlier. Before we get started with MDI form, you must set the **MDI Child** property of following Forms to 'True'.
frmEmpDetails
frmSalary
frmDependent

Create a New MDI Form

To create a new MDI From, go to **Project1.vbp > [**Right Click] and then [Select] **Add** > [Click] **MDI Form.**
Right Click the form and you will see the **Menu Editor,** using the editor create following menu items.
Caption : Employee Details

Name : employee

Index : 0

Caption : Salary Calculation

Name : salary

Index : 1

Caption : Dependents

Name : dependents

Index : 2

Caption : Exit

Name : exit

Index : 3

Code for frmMENU

The code for frmMENU is simple, you click on one of the menu item and a new form opens up, however the new form does not open in a separate windows, but it will open within frmMENU as MDI Child form.

```
Form Menu

Private Sub depend_Click(Index As Integer)
    frmDependent.Show
End Sub

Private Sub empdetails_Click(Index As Integer)
    frmEmpDetails.Show
End Sub

Private Sub exit_Click(Index As Integer)
    Unload Me
    Unload frmLogin
End Sub

Private Sub salary_Click(Index As Integer)
    frmSalary.Show
End Sub
```