

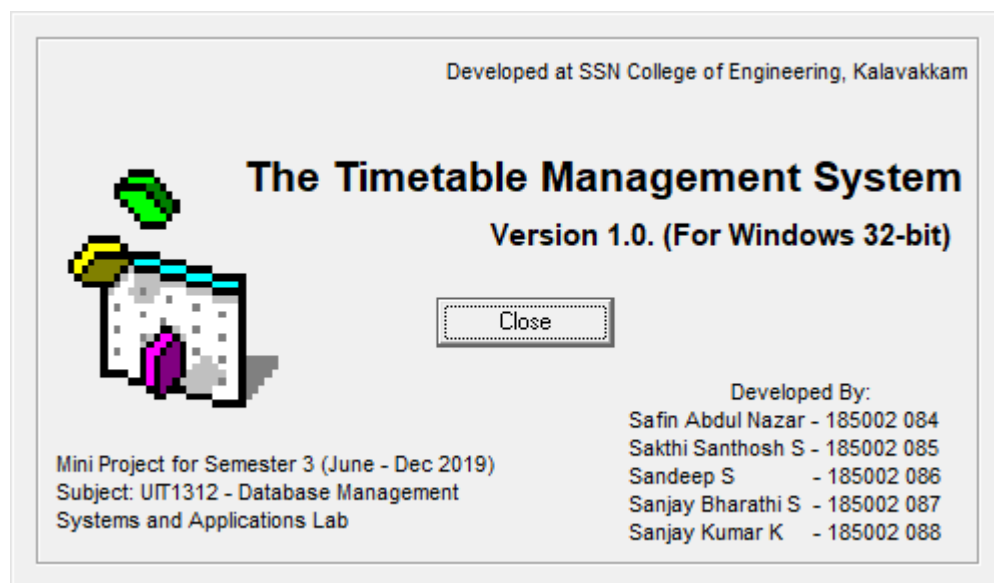
MINI PROJECT: TIME TABLE MANAGEMENT SYSTEM

SOURCE CODE AND SCREENSHOTS:

FRMABOUT:

```
Private Sub cmdOK_Click()  
    Me.Hide  
    Unload Me  
End Sub
```

SCREENSHOT:



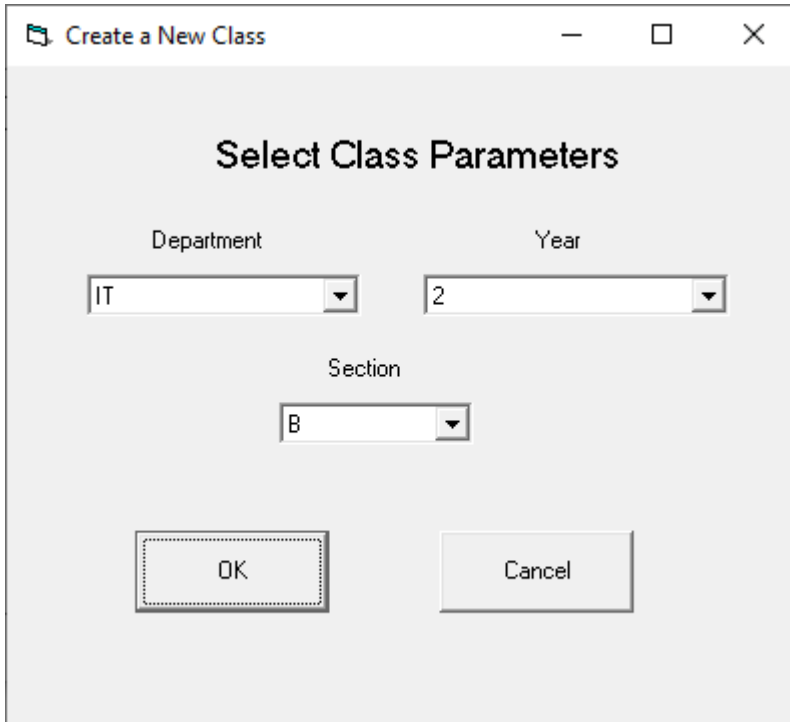
FRMADDCLASS:

```
Option Explicit  
Dim rst As Recordset  
Dim con As Connection  
Dim ctr As Integer  
Private Sub cmdOK_Click()  
    dept = cbxDept.Text  
    sec = cbxSection.Text  
    year = cbxYear.Text  
    Call createClass(UCase(dept), UCase(sec), UCase(year), currUID,  
con)  
    MsgBox "Class with ID: '" & UCase(dept) & UCase(sec) &  
UCase(year) & "' was created successfully.", vbInformation, "Class  
created successfully."  
End Sub  
Private Sub cmdCancel_Click()  
    Unload Me  
    Load frmDashboard  
    frmDashboard.Show
```

STUDENT ID: IT5086

```
End Sub
Private Sub Command2_Click()
    Me.Hide
    Unload Me
End Sub
Private Sub Form_Load()
    cbxDept.AddItem "IT"
    cbxDept.AddItem "CSE"
    cbxDept.AddItem "ECE"
    cbxDept.AddItem "EEE"
    cbxDept.AddItem "Mech."
    cbxDept.AddItem "BME"
    cbxDept.AddItem "Civil"
    cbxDept.AddItem "Chemical"
    cbxSection.AddItem "A"
    cbxSection.AddItem "B"
    cbxYear.AddItem "1"
    cbxYear.AddItem "2"
    cbxYear.AddItem "3"
    cbxYear.AddItem "4"
    Call establishDBConnection
End Sub
Function establishDBConnection()
    On Error GoTo errorHandler
    Set con = CreateObject("ADODB.Connection")
    con.ConnectionString = conString
    con.Open
    Set rst = CreateObject("ADODB.Recordset")
    Exit Function
errorHandler:
    con.Close
    MsgBox "The connection to the Oracle database failed! The
application will now terminate.", vbCritical + vbOKOnly, "Login
Failed!"
    Unload Me
End Function
```

SCREENSHOT :



The screenshot shows a Windows-style dialog box titled "Create a New Class". The dialog's main content area is titled "Select Class Parameters". It features three dropdown menus for selection: "Department" (currently showing "IT"), "Year" (currently showing "2"), and "Section" (currently showing "B"). At the bottom of the dialog, there are two buttons: "OK" and "Cancel". The "OK" button is highlighted with a dashed border.

FRMADMINSUBJECTDETAILS :

```
Dim con As New ADODB.Connection
Dim currClassID As String
Dim rst As New ADODB.Recordset
Option Explicit
Private Sub cmdRemoveClass_Click()
    frmDeleteClass.Show
End Sub
Private Sub cbxClasses_Change()
    Call getCurrClassID
    Call refreshSubjectsList
End Sub
Private Sub cmdAdd_Click()
    On Error Resume Next
    frmCreateSubject.Show
End Sub
Private Sub cmdRefresh_Click()
    Call getCurrClassID
    Call refreshSubjectsList
End Sub
Private Sub cmdRemove_Click()
    frmDeleteSubject.Show
End Sub
Private Sub dgSubjects_Click()
End Sub
```

STUDENT ID: IT5086

```
Private Sub Form_Load()
    con.ConnectionString = "Provider=MSDAORA.1;Password=admin;User
ID=system;Data Source=TTMSDB;Persist Security Info=True"
    con.CursorLocation = adUseClient
    con.Open
    rst.Open "SELECT * FROM Classes", con
    Do While rst.EOF = False
        cbxClasses.AddItem rst("dept") & " - " & rst("section") & "
(Year: " & rst("year") & ")"
        rst.MoveNext
    Loop
End Sub

Private Sub Form_Unload(Cancel As Integer)
    con.Close
End Sub

Function refreshSubjectsList()
    Dim temp As New Recordset
    temp.Open "SELECT SubCode AS Subject_Code, SubName AS
Subject_Name, FacultyID as Faculty_ID, Credits AS No_of_Credits
FROM Subjects WHERE ClassID = '" & currClassID & "'", con
    Set dgSubjects.DataSource = temp
End Function

Function getCurrClassID()
    rst.Close
    rst.Open "SELECT * FROM Classes", con

    Do While rst.EOF = False
        If rst("dept") & " - " & rst("section") & " (Year: " &
rst("year") & ")" = cbxClasses.Text Then
            currClassID = rst("ID")
            Exit Do
        End If
        rst.MoveNext
    Loop
End Function
```

SCREENSHOT :

The screenshot shows a window titled "Manage Subjects...". At the top, there is a label "Select Class ID:" followed by a dropdown menu showing "IT - B (Year: 2)". To the right of the dropdown is a button labeled "Refresh Subjects List!". Below this is a table with the following data:

	SUBJECT CODE	SUBJECT NAME	FACULTY ID	NO OF CREDITS
▶	UIT1301	Digital Electronics	faculty1	4
	UIT1302	Data Structures	faculty2	3

Below the table is a large empty rectangular area. At the bottom of the window, there are two buttons: "Add Subject!" and "Remove Subject!".

FRMCLASSES :

```

Dim con As New ADODB.Connection
Dim rst As New ADODB.Recordset
Option Explicit
Private Sub cmdAddClass_Click()
    frmAddClass.Show
End Sub
Private Sub cmdRefresh_Click()
    dgClasses.Visible = False
    If rst.State = 1 Then
        rst.Close
    End If
    con.Close
    con.ConnectionString = "Provider=MSDAORA.1;Password=admin;User
ID=system;Data Source=TTMSDB;Persist Security Info=True"
    con.CursorLocation = adUseClient
    con.Open
    rst.Open "SELECT * FROM Classes", con
    If IsNull(rst("ID")) Then

```

STUDENT ID: IT5086

```
        MsgBox "No Classes have been added! Please select the 'Add
Classes...' button on the right to add a class!", vbExclamation +
vbOKOnly, "No Classes found!"
        rst.Close
        cmdRemoveClass.Enabled = False
    ElseIf rst.EOF = True Then
        MsgBox "No Classes have been added! Please select the 'Add
Classes...' button on the right to add a class!", vbExclamation +
vbOKOnly, "No Classes found!"
        rst.Close
        cmdRemoveClass.Enabled = False
    Else
        Set dgClasses.DataSource = rst
    End If
    dgClasses.Visible = True
End Sub

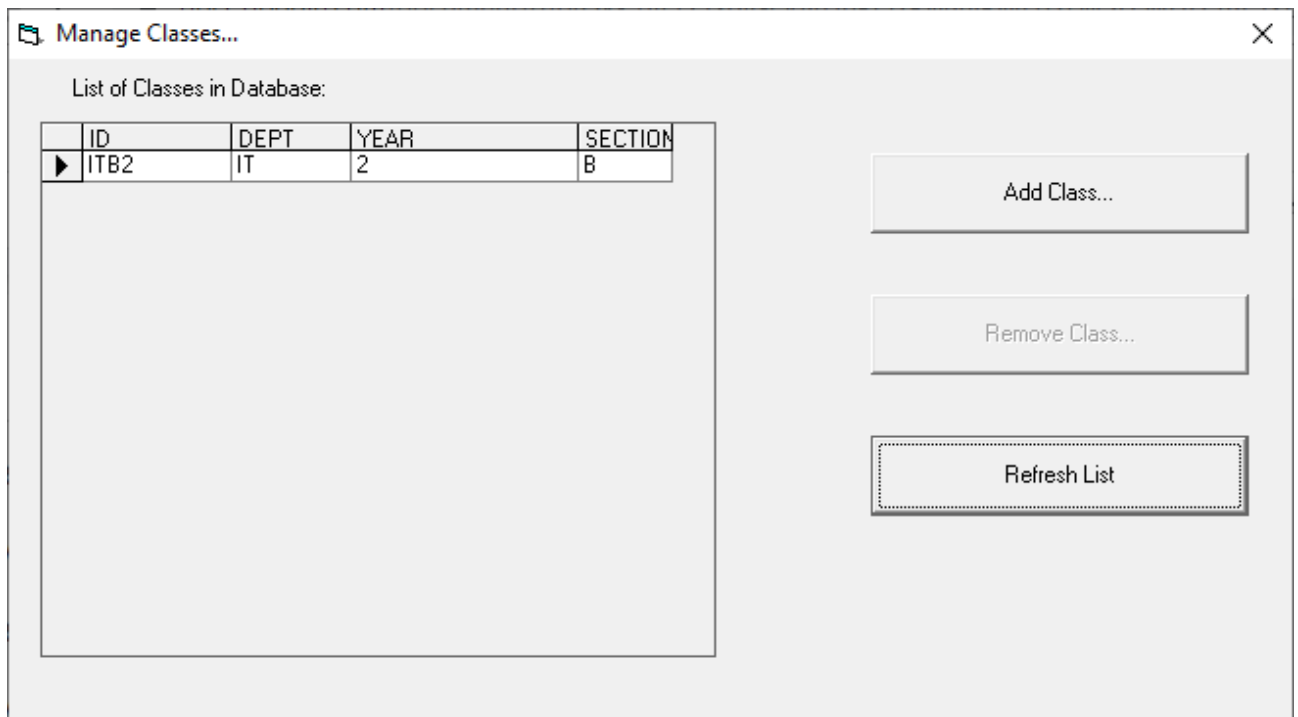
Private Sub cmdRemoveClass_Click()
    frmDeleteClass.Show
End Sub

Private Sub Form_Load()
    con.ConnectionString = "Provider=MSDAORA.1;Password=admin;User
ID=system;Data Source=TTMSDB;Persist Security Info=True"
    con.CursorLocation = adUseClient
    con.Open
    rst.Open "SELECT * FROM Classes", con
    If IsNull(rst("ID")) Then
        MsgBox "No Classes have been added! Please select the 'Add
Classes...' button on the right to add a class!", vbExclamation +
vbOKOnly, "No Classes found!"
        rst.Close
        cmdRemoveClass.Enabled = False
    ElseIf rst.EOF = True Then
        MsgBox "No Classes have been added! Please select the 'Add
Classes...' button on the right to add a class!", vbExclamation +
vbOKOnly, "No Classes found!"
        rst.Close
        cmdRemoveClass.Enabled = False
    Else
        Set dgClasses.DataSource = rst
    End If
End Sub

Private Sub Form_Unload(Cancel As Integer)
    con.Close
End Sub
```

STUDENT ID: IT5086

SCREENSHOT :



FRMCLASSELECTION:

```
Dim rst As New ADODB.Recordset
Dim con As New ADODB.Connection
Option Explicit
Private Sub Command1_Click()
    On Error Resume Next
    TTString = "tt" & UCase(cbxDept.Text) & cbxSection.Text &
    cbxYear.Text
    On Error Resume Next
    Set con = New ADODB.Connection
    Set rst = New ADODB.Recordset
    con.Open "Provider=MSDAORA.1;Password=admin;User ID=system;Data
Source=TTMSDB;Persist Security Info=True"
    con.CursorLocation = adUseClient
    rst.Open "SELECT pl FROM " & TTString, con
    If IsNull(rst("pl")) Then
        MsgBox "The timetable for this class has not been prepared
yet! Please wait till you are allocated a timetable by the
administrator!", vbExclamation, "Timetable not found!"
        Exit Sub
    Else
        frmTimetableGridView.Show
    End If
    con.Close
End Sub
Private Sub Command2_Click()
Unload Me
```

STUDENT ID: IT5086

```
Load frmDashboard
frmDashboard.Show
End Sub
Private Sub Form_Load()
    Call establishDBConnection
    cbxDept.AddItem "IT"
    cbxDept.AddItem "CSE"
    cbxDept.AddItem "ECE"
    cbxDept.AddItem "EEE"
    cbxDept.AddItem "Mech."
    cbxDept.AddItem "BME"
    cbxDept.AddItem "Civil"
    cbxDept.AddItem "Chemical"
    cbxSection.AddItem "A"
    cbxSection.AddItem "B"
    cbxYear.AddItem "1"
    cbxYear.AddItem "2"
    cbxYear.AddItem "3"
    cbxYear.AddItem "4"
End Sub
Function establishDBConnection()
    On Error GoTo errorHandler
    Set con = CreateObject("ADODB.Connection")
    con.ConnectionString = conString
    con.Open
    Exit Function
errorHandler:
    con.Close
    MsgBox "The connection to the Oracle database failed! The
application will now terminate.", vbCritical + vbOKOnly, "Login
Failed!"
    Unload Me
End Function
```


SCREENSHOT :

Choose a Class...

Select a Class using the options below:

Department: IT Year: 2

Section: B

OK Cancel

FRMCLASSELECTIONFORCREATETT

```

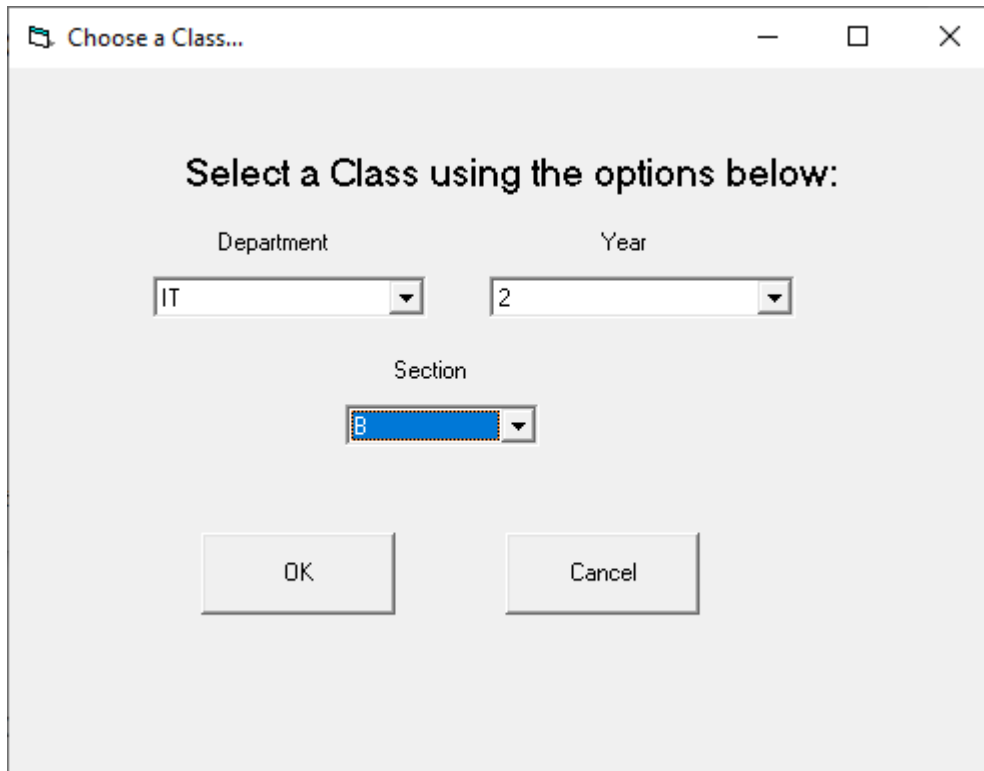
Dim rst As New ADODB.Recordset
Dim con As New ADODB.Connection
Option Explicit
Private Sub Command1_Click()
    'On Error Resume Next
    TTString = "tt" & UCase(cbxDept.Text) & cbxSection.Text &
    cbxYear.Text
    On Error Resume Next
    Set con = New ADODB.Connection
    Set rst = New ADODB.Recordset
    con.Open "Provider=MSDAORA.1;Password=admin;User ID=system;Data
Source=TTMSDB;Persist Security Info=True"
    con.CursorLocation = adUseClient
    rst.Open "SELECT TTString FROM Timetables WHERE TTString = '" &
    TTString & "'", con
    If IsNull(rst("TTString")) Or rst.EOF = True Then
        If MsgBox("The timetable for this class has not been
prepared yet! Create new timetable now?", vbQuestion + vbYesNo,
"Create timetable?") = vbNo Then
            rst.Close
            Unload Me
            Exit Sub
        Else
            MsgBox Mid(TTString, 3)

```

STUDENT ID: IT5086

```
        Call createTimetable(Mid(TTString, 3), TTString,
currUID, con)
        con.Execute "commit"
    End If
End If
con.Close
frmTimetableGrid.Show
Me.Hide
End Sub
Private Sub Command2_Click()
Unload Me
Load frmDashboard
frmDashboard.Show
End Sub
Private Sub Form_Load()
    Call establishDBConnection
    cbxDept.AddItem "IT"
    cbxDept.AddItem "CSE"
    cbxDept.AddItem "ECE"
    cbxDept.AddItem "EEE"
    cbxDept.AddItem "Mech."
    cbxDept.AddItem "BME"
    cbxDept.AddItem "Civil"
    cbxDept.AddItem "Chemical"
    cbxSection.AddItem "A"
    cbxSection.AddItem "B"
    cbxYear.AddItem "1"
    cbxYear.AddItem "2"
    cbxYear.AddItem "3"
    cbxYear.AddItem "4"
End Sub
Function establishDBConnection()
    On Error GoTo errorHandler
    Set con = CreateObject("ADODB.Connection")
    con.ConnectionString = conString
    con.Open
    Exit Function
errorHandler:
    con.Close
    MsgBox "The connection to the Oracle database failed! The
application will now terminate.", vbCritical + vbOKOnly, "Login
Failed!"
    Unload Me
End Function
```

SCREENSHOT :



Choose a Class...

Select a Class using the options below:

Department: IT Year: 2

Section: B

OK Cancel

FRMCREATESUBJECT :

```
Dim con As Connection
Dim rstFaculty As Recordset
Dim rstClasses As Recordset
Private Sub cmdCancel_Click()
    If MsgBox("Close screen?", vbQuestion + vbYesNo, "Confirmation")
= vbYes Then
        If con.State = 1 Then
            con.Close
        End If
        Me.Hide
        Unload Me
    End If
End Sub
Private Sub cmdCreate_Click()
    'On Error GoTo errorHandler
    rstFaculty.MoveFirst
    rstClasses.MoveFirst
    If txtSubCode.Text = "" Or cbxClassID.Text = "" Or
txtSubName.Text = "" Or txtCredits.Text = "" Then
        MsgBox "One or more required values are empty! Please fill
them up before continuing!", vbExclamation + vbOKOnly, "Empty values
not allowed!"
        Exit Sub
    End If
```

```
If Len(txtSubCode.Text) > 7 Then
    MsgBox "The entered subject code is too long! Maximum
allowed: 7 characters.", vbExclamation + vbOKOnly, "Invalid subject
code!"
    Exit Sub
End If
If Len(txtSubName.Text) > 30 Then
    MsgBox "The entered subject name is too long! Maximum
allowed: 30 characters.", vbExclamation + vbOKOnly, "Invalid subject
name!"
    Exit Sub
End If
If (txtCredits.Text) > "6" Or txtCredits.Text < "0" Then
    MsgBox "Invalid number of credits! Maximum value: 6.",
vbExclamation + vbOKOnly, "Invalid number of credits!"
    Exit Sub
End If
If IsNotCompatibleString(txtSubCode.Text) Or
isNotCompatibleString(txtSubName.Text) Or
isNotCompatibleString(txtCredits.Text) Or
isNotCompatibleString(cbxFacultyID.Text) Or
isNotCompatibleString(cbxClaSSID.Text) Then
    MsgBox "One or more entered values contains special
characters, which are not allowed! Please enter only alphanumeric
characters!", vbExclamation + vbOKOnly, "Invalid value(s) entered!"
    Exit Sub
End If
Dim okflag As Boolean
okflag = False
Do While rstFaculty.EOF = False
    If rstFaculty("ID") = cbxFacultyID.Text Then
        okflag = True
        Exit Do
    End If
    rstFaculty.MoveNext
Loop
If okflag = True Then
    okflag = False
    Do While rstClasses.EOF = False
        If rstClasses("ID") = cbxClaSSID.Text Then
            okflag = True
            Exit Do
        End If
        rstClasses.MoveNext
    Loop
End If
If okflag = False Then
```

STUDENT ID: IT5086

```
        MsgBox "The entered faculty ID or class ID does not exist in  
the database! Please re-enter proper values!", vbExclamation +  
vbOKOnly, "Invalid Class ID/Faculty ID!"
```

```
        Exit Sub
```

```
    End If
```

```
    Call createSubject(txtSubCode.Text, txtSubName.Text,  
cbxFacultyID.Text, cbxClassID.Text, txtCredits.Text, currUID, con)
```

```
    MsgBox "The subject " & txtSubCode.Text & " - " &  
txtSubName.Text & " was created successfully.", vbInformation, "User  
Creation Successful. "
```

```
    Exit Sub
```

```
'errorHandler:
```

```
    '    MsgBox "A critical error has occurred! The operation will now  
rollback and terminate!", vbCritical + vbOKOnly, "Critical Error"
```

```
End Sub
```

```
Function establishDBConnection()
```

```
    On Error GoTo errorHandler
```

```
    Set con = CreateObject("ADODB.Connection")
```

```
    con.ConnectionString = conString
```

```
    con.Open
```

```
    Set rstFaculty = CreateObject("ADODB.Recordset")
```

```
    Set rstClasses = CreateObject("ADODB.Recordset")
```

```
    Exit Function
```

```
errorHandler:
```

```
    con.Close
```

```
    MsgBox "The connection to the Oracle database failed! The  
application will now terminate.", vbCritical + vbOKOnly, "Login  
Failed!"
```

```
    Unload Me
```

```
End Function
```

```
Private Sub Form_Load()
```

```
    Call establishDBConnection
```

```
    rstClasses.Open "SELECT ID FROM Classes", con
```

```
    rstFaculty.Open "SELECT ID FROM Users WHERE Role = 'Faculty'",  
con
```

```
    If IsNull(rstFaculty("ID")) Or IsNull(rstClasses("ID")) Or  
rstFaculty.EOF = True Or rstClasses.EOF = True Then
```

```
        MsgBox "Cannot create subject without faculty and class!  
Please add at least one faculty and a class to continue!",  
vbExclamation + vbOKOnly, "Error"
```

```
        Unload Me
```

```
        Exit Sub
```

```
    End If
```

```
    Do While rstClasses.EOF = False
```

```
        cbxClassID.AddItem rstClasses("ID")
```

```
        rstClasses.MoveNext
```

```
    Loop
```

```
    rstClasses.MoveFirst
```

```
    Do While rstFaculty.EOF = False
```

```
        cbxFacultyID.AddItem rstFaculty("ID")
```

STUDENT ID: IT5086

```
        rstFaculty.MoveNext
    Loop
End Sub
```

SCREENSHOT :

Add a New Subject...

Please fill up the details below to create the new subject:

Subject Code: UIT1302

Subject Name: Data Structures

No. of Credits: 3

Faculty ID: faculty2

Class ID: ITB2

Create Subject! Close Screen

FRMCREATEUSER :

```
Dim con As Connection
Dim rst As Recordset
Private Sub cmdCancel_Click()
    If MsgBox("Close screen?", vbQuestion + vbYesNo, "Confirmation")
= vbYes Then
        If con.State = 1 Then
            con.Close
        End If
        Me.Hide
        Unload Me
    End If
End Sub
Private Sub cmdClear_Click()
    If MsgBox("Are you sure you want to clear all values?",
vbQuestion + vbYesNo, "Clear All Values?") = vbYes Then
        txtUID.Text = ""
        txtPassword.Text = ""
        txtConfPassword.Text = ""
        txtUName.Text = ""
    End If
End Sub
Private Sub cmdCreate_Click()
    'On Error GoTo errorHandler
    If txtUID.Text = "" Or txtUName.Text = "" Or txtPassword.Text =
```

STUDENT ID: IT5086

```
"" Or txtConfPassword.Text = "" Then
    MsgBox "One or more required values (Username, User ID,
Password) textboxes are empty! Please fill them up before
continuing!", vbExclamation + vbOKOnly, "Invalid User ID!"
    Exit Sub
End If
If Len(txtUID.Text) > 10 Then
    MsgBox "The entered user ID is too long! Maximum allowed: 10
characters.", vbExclamation + vbOKOnly, "Invalid User ID!"
    Exit Sub
End If
If Len(txtUName.Text) > 30 Then
    MsgBox "The entered username is too long! Maximum allowed:
30 characters.", vbExclamation + vbOKOnly, "Invalid User ID!"
    Exit Sub
End If
If Len(txtPassword.Text) > 10 Then
    MsgBox "The entered password is too long! Maximum allowed:
10 characters.", vbExclamation + vbOKOnly, "Invalid User ID!"
    Exit Sub
End If
If IsNotCompatibleString(txtUID.Text) Or
IsNotCompatibleString(txtUName.Text) Or
IsNotCompatibleString(txtPassword.Text) Or
IsNotCompatibleString(txtConfPassword.Text) Then
    MsgBox "One or more entered values contains special
characters, which are not allowed! Please enter only alphanumeric
characters!", vbExclamation + vbOKOnly, "Invalid value(s) entered!"
    Exit Sub
End If

If txtConfPassword.Text <> txtPassword.Text Then
    MsgBox "Password entries in both the fields do not match!
Please enter the same password for both locations!", vbExclamation +
vbOKOnly, "Invalid Password/Confirm Password!"
    Exit Sub
End If
rst.Open "SELECT ID FROM Users WHERE ID = '" & txtUID.Text &
"", con
If rst.EOF = False Then
    MsgBox "The entered user ID already exists in the database!
Please choose a different User ID!", vbExclamation + vbOKOnly,
"Invalid User ID!"
    rst.Close
    Exit Sub
End If
Call createUser(txtUID.Text, txtUName.Text, txtPassword.Text,
cbxRole.Text, currUID, con)
MsgBox "The user was created successfully.", vbInformation,
"User Creation Successful. "
```

STUDENT ID: IT5086

```
        rst.Close
    Exit Sub
'errorHandler:
'    MsgBox "A critical error has occurred! The operation will now
rollback and terminate!", vbCritical + vbOKOnly, "Critical Error"
End Sub
Function establishDBConnection()
    On Error GoTo errorHandler
    Set con = CreateObject("ADODB.Connection")
    con.ConnectionString = conString
    con.Open
    Set rst = CreateObject("ADODB.Recordset")
    Exit Function
errorHandler:
    con.Close
    MsgBox "The connection to the Oracle database failed! The
application will now terminate.", vbCritical + vbOKOnly, "Login
Failed!"
    Unload Me
End Function
Private Sub Form_Load()
    Call establishDBConnection
    cbxRole.AddItem "Student"
    cbxRole.AddItem "Faculty"
    cbxRole.AddItem "Administrator"
    cbxRole.Text = cbxRole.List(0)
End Sub
```

SCREENSHOT :

Add New User

Please fill up the details below to create the new user:

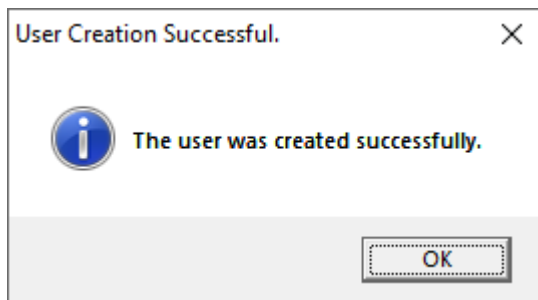
User ID:

Name of the User:

Password:

Confirm Password:

Role:



Add New User

Please fill up the details below to create the new user:

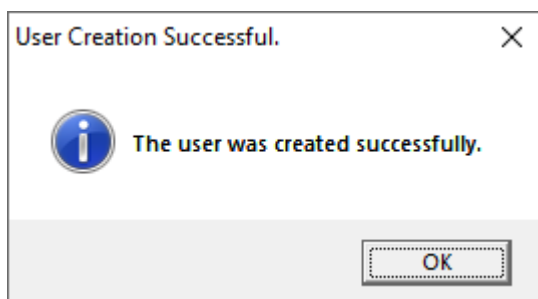
User ID:

Name of the User:

Password:

Confirm Password:

Role:



FRMDASHBOARD :

```
Dim con As Connection
Dim rst As Recordset
Option Explicit
Private Sub aboutTTMS_Click()
    Load frmAbout
    frmAbout.Show
End Sub
Private Sub changePassword_Click(Index As Integer)
    Load frmUpdatePassword
    frmUpdatePassword.Show
End Sub
Private Sub class_Click()
    Load frmClasses
    frmClasses.Show
```

STUDENT ID: IT5086

```
End Sub
Private Sub createTT_Click(Index As Integer)
    frmclassselectionforCreateTT.Show
End Sub
Private Sub createUser_Click(Index As Integer)
    Load frmCreateUser
    frmCreateUser.Show
End Sub
Private Sub editUser_Click(Index As Integer)
    Load frmUpdateUser
    frmUpdateUser.Show
End Sub
Private Sub Form_Load()
    Call establishDBConnection
    Call hideOptionsAccordingtoUser
    If (currUID = "system") Then
        lblUserDetails.Caption = "Account Name: " & currUName
    End If
    If lastLoginTime = "FIRST_LOGIN" Then
        If currUID <> "system" Then
            lblUserDetails.Caption = "Welcome, " & currUName & "!"
        End If
        lblLoginTimes.Caption = "
You have logged in for the first time!"
    Else
        If currUID <> "system" Then
            lblUserDetails.Caption = "Welcome back, " & currUName &
"!"
        End If
        lblLoginTimes.Caption = "Last successful login time: " &
lastLoginTime
    End If
    If lastLogoutTime <> "FIRST_LOGOUT" Then
        lblLoginTimes.Caption = lblLoginTimes.Caption & " Last
successful logout time: " & lastLogoutTime & " "
    End If
    lblRole.Caption = "Your Role: " & currURole
    If lastLoginTime > lastLogoutTime Then
        MsgBox "It looks like the application has crashed the
previous time you logged in! For database consistency, it is
recommended that you logout and re-login immediately!",
vbExclamation + vbOKOnly
    End If
End Sub
Private Sub helpDocs_Click()
    frmHelpDocs.Show
End Sub
Private Sub logOut_Click(Index As Integer)
    If con.State <> 1 Then
        Call establishDBConnection
```

```
End If
If MsgBox("Are you sure you want to log out of TTMS?",
vbQuestion + vbYesNo, "Confirm Log Out") = vbYes Then
    Call updateLogoutTimeForUser(currUID, con)
    con.Close
    Load frmLogin
    frmLogin.Show
    Me.Hide
    Unload Me
    Exit Sub
Else
    Exit Sub
End If
errorHandler:
    MsgBox "The connection to the Oracle database failed! The
application will now terminate.", vbCritical + vbOKOnly, "Login
Failed!"
    Me.Hide
    End
End Sub
Function establishDBConnection()
    On Error GoTo errorHandler
    Set con = CreateObject("ADODB.Connection")
    con.ConnectionString = conString
    con.Open
    Set rst = CreateObject("ADODB.Recordset")
    Exit Function
errorHandler:
    MsgBox "The connection to the Oracle database failed! The
application will now terminate.", vbCritical + vbOKOnly, "Login
Failed!"
    Me.Hide
    End
End Function
Function hideOptionsAccordingtoUser()
    If currUID = "system" Then
        changePassword.Item(12).Visible = False
    End If
    If currURole = "Student" Then
        Actions.Item(1).Visible = False
    ElseIf currURole = "Teacher" Then
        Actions.Item(11).Visible = False
    End If
End Function
Private Sub removeTT_Click()
    frmDeleteTimetable.Show
End Sub
Private Sub subjects_Click()
    Load frmadminssubjectdetails
    frmadminssubjectdetails.Show
```

STUDENT ID: IT5086

```
End Sub
Private Sub viewLogs_Click()
    Load frmViewLogs
    frmViewLogs.Show
End Sub
Private Sub viewSub_Click()
    Load frmsubjectdetails
    frmsubjectdetails.Show
End Sub
Private Sub viewtimetable_Click()
    Load frmclassselection
    frmclassselection.Show
End Sub
Private Sub viewUsers_Click()
    Load frmViewUsers
    frmViewUsers.Show
End Sub
```

SCREENSHOT :



FRMDELETECLASS :

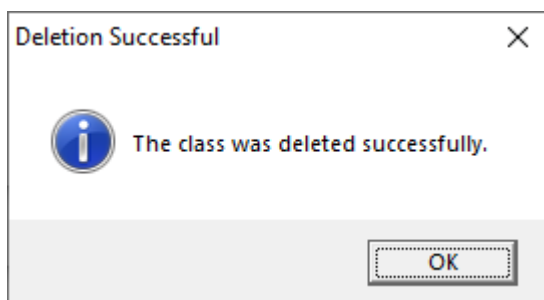
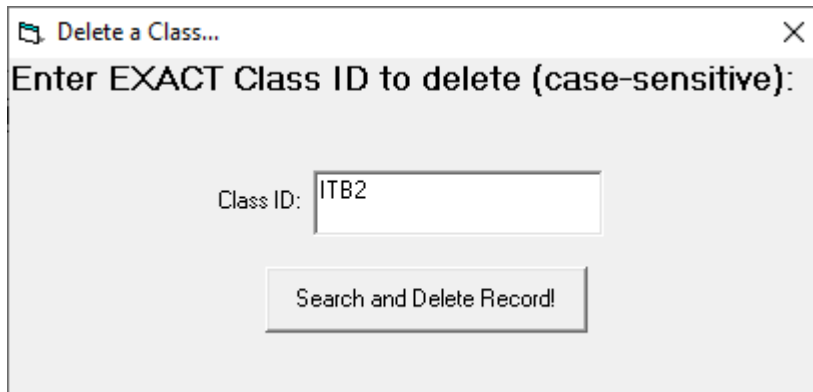
```
Dim con As Connection
Dim rst As Recordset
Private Sub cmdDelete_Click()
    cmdDelete.Enabled = False
    If txtUID.Text = "" Then
        MsgBox "Please enter a valid class ID in the textbox!",
vbExclamation + vbOKOnly
        Exit Sub
    End If
    If Len(txtUID.Text) > 10 Then
        MsgBox "The entered entered class ID is invalid! Please try
again!", vbExclamation + vbOKOnly, "Invalid User ID!"
        Exit Sub
    End If
```

STUDENT ID: IT5086

```
End If
rst.Open "SELECT ID FROM Classes WHERE ID = '" & txtUID.Text &
"'", con
If IsNull(rst("ID")) Then
    MsgBox "The entered entered class ID does not exist in the
database! Please enter a valid ID!", vbExclamation + vbOKOnly,
"Invalid User ID!"
End If
rst.Close
If MsgBox("Changes made once cannot be rolled back after
successful update. Continue with updation?", vbExclamation +
vbYesNo, "Confirm deletion?") = vbYes Then
    If MsgBox("Deleting a class will also delete ALL the
subjects and timetables associated with that class. Continue with
updation?", vbExclamation + vbYesNo, "Confirm deletion?") = vbYes
Then
        Call deleteClass(txtUID.Text, currUID, con)
        MsgBox "The class was deleted successfully.",
vbInformation, "Deletion Successful"
    End If
End If
Exit Sub
'errorHandler:
'MsgBox "A critical error has occurred! The operation will now
rollback and terminate!", vbCritical + vbOKOnly, "Critical Error"
End Sub
Function establishDBConnection()
    On Error GoTo errorHandler
    Set con = CreateObject("ADODB.Connection")
    con.ConnectionString = conString
    con.Open

    Set rst = CreateObject("ADODB.Recordset")
    Exit Function
errorHandler:
    con.Close
    MsgBox "The connection to the Oracle database failed! The
application will now terminate.", vbCritical + vbOKOnly, "Login
Failed!"
    Unload Me
End Function
Private Sub Form_Load()
    Call establishDBConnection
End Sub
Private Sub txtUID_Change()
    cmdDelete.Enabled = True
End Sub
```

SCREENSHOT :



FRMDELETESUBJECT :

```
Dim con As Connection
Dim rst As Recordset
Private Sub cmdDelete_Click()
    cmdDelete.Enabled = False
    If txtSubCode.Text = "" Then
        MsgBox "Please enter a valid subject code in the textbox!",
vbExclamation + vbOKOnly
        Exit Sub
    End If
    If Len(txtSubCode.Text) > 7 Then
        MsgBox "The entered entered subject code is invalid! Please
try again!", vbExclamation + vbOKOnly, "Invalid subject code!"
        Exit Sub
    End If
    rst.Open "SELECT SubCode FROM Subjects WHERE SubCode = '" &
txtSubCode.Text & "'", con
    If rst.EOF = True Then
        MsgBox "The entered entered subject code does not exist in
the database! Please try again!", vbExclamation + vbOKOnly, "Invalid
subject code!"
    End If
    rst.Close
    If MsgBox("Changes made once cannot be rolled back after
successful update. Continue with updation?", vbExclamation +
vbYesNo, "Confirm updation") = vbYes Then
        Call deleteSubject(txtSubCode.Text, currUID, con)
```

STUDENT ID: IT5086

```
        MsgBox "The subject record was deleted successfully.",
vbInformation, "Deletion Successful"
    End If
    Exit Sub
'errorHandler:
    'MsgBox "A critical error has occurred! The operation will now
rollback and terminate!", vbCritical + vbOKOnly, "Critical Error"
End Sub
Function establishDBConnection()
    On Error GoTo errorHandler
    Set con = CreateObject("ADODB.Connection")
    con.ConnectionString = conString
    con.Open
    Set rst = CreateObject("ADODB.Recordset")
    Exit Function
errorHandler:
    con.Close
    MsgBox "The connection to the Oracle database failed! The
application will now terminate.", vbCritical + vbOKOnly, "Login
Failed!"
    Unload Me
End Function
Private Sub Form_Load()
    Call establishDBConnection
End Sub
Private Sub txtSubCode_Change()
    cmdDelete.Enabled = True
End Sub
```

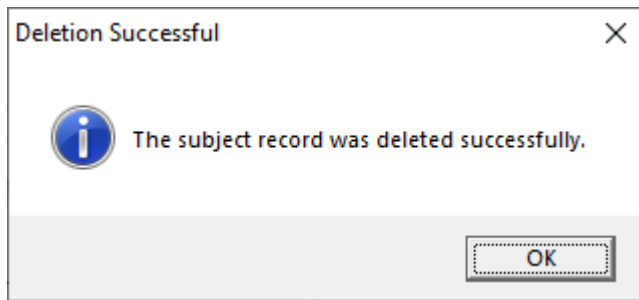
SCREENSHOT :



Delete a Subject: X

Enter EXACT Subject Code to delete (case-sensitive):

Subject Code:

**FRMDELETETIMETABLE :**

```

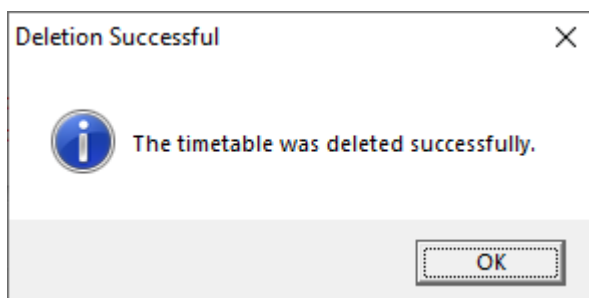
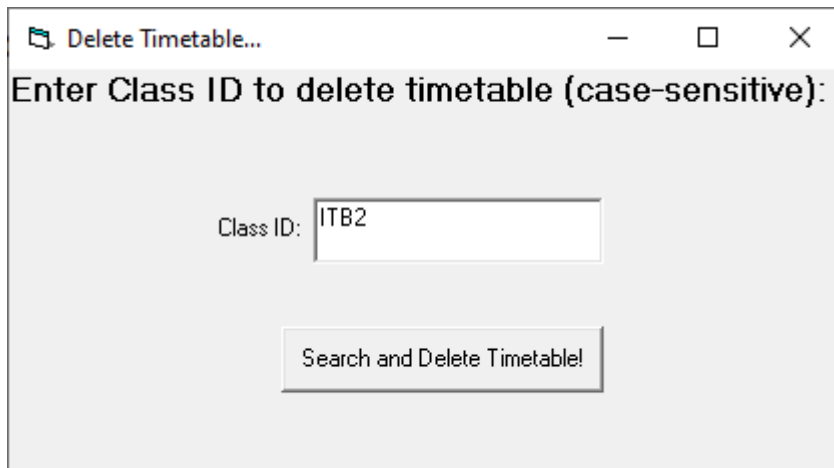
Dim con As Connection
Dim rst As Recordset
Private Sub cmdDelete_Click()
    On Error Resume Next
    cmdDelete.Enabled = False
    If txtClassID.Text = "" Then
        MsgBox "Please enter a valid class ID in the textbox!",
vbExclamation + vbOKOnly
        Exit Sub
    End If
    If Len(txtClassID.Text) > 7 Then
        MsgBox "The entered entered class ID is invalid! Please try
again!", vbExclamation + vbOKOnly, "Invalid class ID!"
        Exit Sub
    End If
    rst.Open "SELECT ClassID FROM Timetables WHERE ClassID = '" &
txtClassID.Text & "'", con
    If IsNull(rst("ClassID")) Or rst.EOF = True Then
        MsgBox "The entered class ID does not exist in the database!
Please try again!", vbExclamation + vbOKOnly, "Invalid class ID!"
        Exit Sub
    End If
    rst.Close
    If MsgBox("Changes made once cannot be rolled back after
successful update. Continue with updation?", vbExclamation +
vbYesNo, "Confirm deletion") = vbYes Then
        Call deleteTimetable(txtClassID.Text, currUID, con)
        MsgBox "The timetable was deleted successfully.",
vbInformation, "Deletion Successful"
    End If
    Exit Sub
'errorHandler:
    'MsgBox "A critical error has occurred! The operation will now
rollback and terminate!", vbCritical + vbOKOnly, "Critical Error"
End Sub
Function establishDBConnection()
    On Error GoTo errorHandler
    Set con = CreateObject("ADODB.Connection")
    con.ConnectionString = conString
    con.Open
    Set rst = CreateObject("ADODB.Recordset")

```


STUDENT ID: IT5086

```
Exit Function
ErrorHandler:
    con.Close
    MsgBox "The connection to the Oracle database failed! The
application will now terminate.", vbCritical + vbOKOnly, "Login
Failed!"
    Unload Me
End Function
Private Sub Form_Load()
    Call establishDBConnection
End Sub
Private Sub txtClassID_Change()
    cmdDelete.Enabled = True
End Sub
```

SCREENSHOT :



FRMDELETEUSER:

```
Dim con As Connection
Dim rst As Recordset
Private Sub cmdDelete_Click()
    cmdDelete.Enabled = False
    If txtUID.Text = "" Then
        MsgBox "Please enter a valid user ID in the textbox!",
vbExclamation + vbOKOnly
        Exit Sub
    End If
```

STUDENT ID: IT5086

```

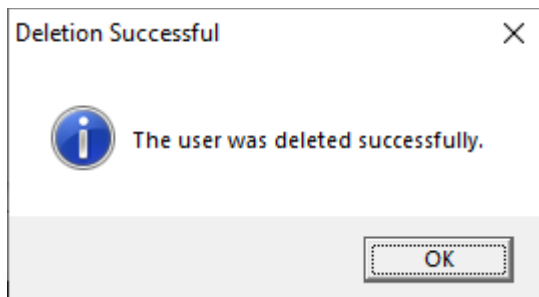
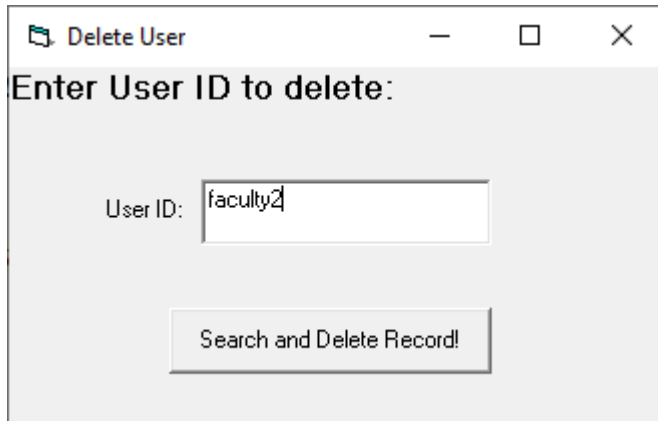
    If txtUID.Text = "system" Then
        MsgBox "You are prohibited from deleting the system recovery account!", vbCritical + vbOKOnly
        Exit Sub
    End If
    If txtUID.Text = currUID Then
        MsgBox "You are prohibited from deleting your own account while logged in!", vbCritical + vbOKOnly
        Exit Sub
    End If
    If Len(txtUID.Text) > 10 Then
        MsgBox "The entered entered user ID is invalid! Please try again!", vbExclamation + vbOKOnly, "Invalid User ID!"
        Exit Sub
    End If
    rst.Open "SELECT ID FROM Users WHERE ID = '" & txtUID.Text & "'", con
    If rst.EOF = True Then
        MsgBox "The entered entered user ID does not exist in the database! Please try again!", vbExclamation + vbOKOnly, "Invalid User ID!"
    End If
    rst.Close
    If MsgBox("Changes made once cannot be rolled back after successful update. Continue with updatation?", vbExclamation + vbYesNo, "Confirm deletion") = vbYes Then
        If MsgBox("Deleting a user will also delete any subject he/she teaches, if he/she is a faculty. Continue with updatation?", vbExclamation + vbYesNo, "Confirm deletion") = vbYes Then
            Call deleteUser(txtUID.Text, currUID, con)
            MsgBox "The user was deleted successfully.", vbInformation, "Deletion Successful"
        End If
    End If
    Exit Sub
'errorHandler:
    'MsgBox "A critical error has occurred! The operation will now rollback and terminate!", vbCritical + vbOKOnly, "Critical Error"
End Sub
Function establishDBConnection()
    On Error GoTo errorHandler
    Set con = CreateObject("ADODB.Connection")
    con.ConnectionString = conString
    con.Open
    Set rst = CreateObject("ADODB.Recordset")
    Exit Function
errorHandler:
    con.Close
    MsgBox "The connection to the Oracle database failed! The application will now terminate.", vbCritical + vbOKOnly, "Login Failed!"

```

STUDENT ID: IT5086

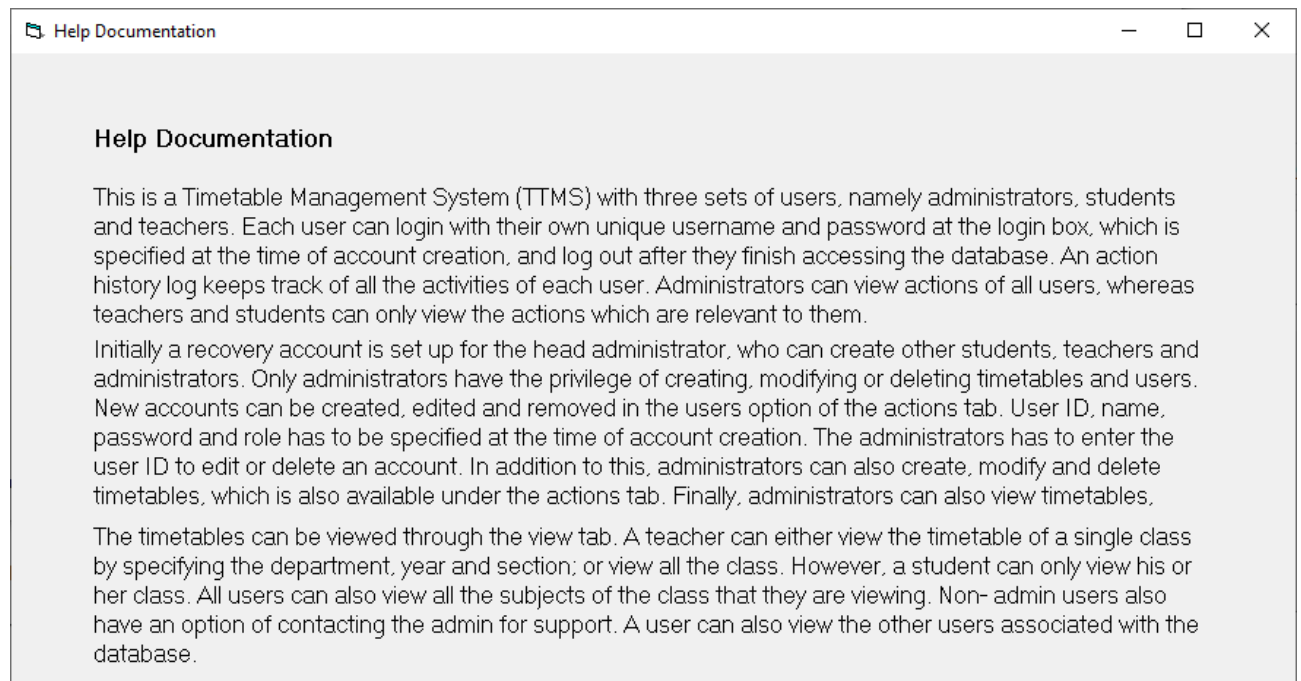
```
Unload Me
End Function
Private Sub Form_Load()
    Call establishDBConnection
End Sub
Private Sub txtUID_Change()
    cmdDelete.Enabled = True
End Sub
```

SCREENSHOT :



FRMHELPDOCS :

SCREENSHOT :



FRMLOGIN:

Option Explicit

Dim rst As Recordset

Dim con As Connection

Private Sub cmdCancel_Click()

 If MsgBox("Are you sure you want to quit TTMS?", vbQuestion + vbYesNo, "Quit TTMS?") = vbYes Then

 Unload Me

 End

End If

End Sub

Private Sub cmdOK_Click()

 Call establishDBConnection

 rst.Open "SELECT ID, Name, Password, Role FROM Users WHERE ID = '" & txtUserName.Text & "'", con

 If rst.State <> 1 Then

 MsgBox "The connection to the Oracle database failed! The application will now terminate.", vbCritical + vbOKOnly, "Login Failed!"

 Me.Hide

 End

End If

If rst.EOF = False Then

 If rst("Password") = txtPassword.Text Then

 currUID = txtUserName.Text

 currUName = rst("Name")

 currURole = rst("Role")

 rst.Close

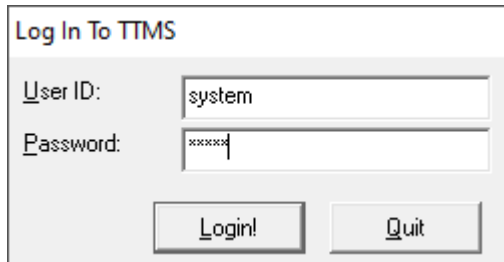
```
        Call updateDBAndLoadDashboard
    Else
        MsgBox "The entered username and/or password does not
exist! Please retry with valid credentials!", vbCritical + vbOKOnly,
"Login Failed!"
    End If
Else
        MsgBox "The entered username and/or password does not
exist! Please retry with valid credentials!", vbCritical + vbOKOnly,
"Login Failed!"
    End If
End Sub
Function establishDBConnection()
    On Error GoTo errorHandler
    Set con = CreateObject("ADODB.Connection")
    con.ConnectionString = conString
    con.Open
    Set rst = CreateObject("ADODB.Recordset")
    Exit Function
errorHandler:
    MsgBox "The connection to the Oracle database failed! The
application will now terminate.", vbCritical + vbOKOnly, "Login
Failed!"
    Me.Hide
    End
End Function
Function updateDBAndLoadDashboard()
    If con.State <> 1 Then
        MsgBox "The connection to the Oracle database failed! The
application will now terminate.", vbCritical + vbOKOnly, "Login
Failed!"
        Me.Hide
        Unload Me
        End
    End If
    rst.Open "SELECT TO_CHAR(LASTLOGINTIME, 'DD-MON-YYYY hh:MI:SS
AM') AS logIntime, TO_CHAR(LASTLOGOUTTIME, 'DD-MON-YYYY hh:MI:SS
AM') AS logOutTime FROM Users WHERE ID = '" & txtUserName.Text &
"'", con
    If IsNull(rst("logIntime")) Then
        lastLoginTime = "FIRST_LOGIN"
    Else
        lastLoginTime = rst("logIntime")
    End If

    If IsNull(rst("logOutTime")) Then
        lastLogoutTime = "FIRST_LOGOUT"
    Else
        lastLogoutTime = rst("logOutTime")
    End If
```

STUDENT ID: IT5086

```
        rst.Close
        con.BeginTrans
        Call updateLoginTimeForUser(currUID, con)
        con.CommitTrans
        con.Close
        Load frmDashboard
        frmDashboard.Show
        Unload Me
    End Function
```

SCREENSHOT :



The screenshot shows a Windows-style dialog box titled "Log In To TTMS". It has a light gray background. Inside, there are two text input fields. The first is labeled "User ID:" and contains the text "system". The second is labeled "Password:" and contains masked characters "xxxxx". Below these fields are two buttons: "Login!" and "Quit". The "Login!" button is slightly larger and more prominent than the "Quit" button.

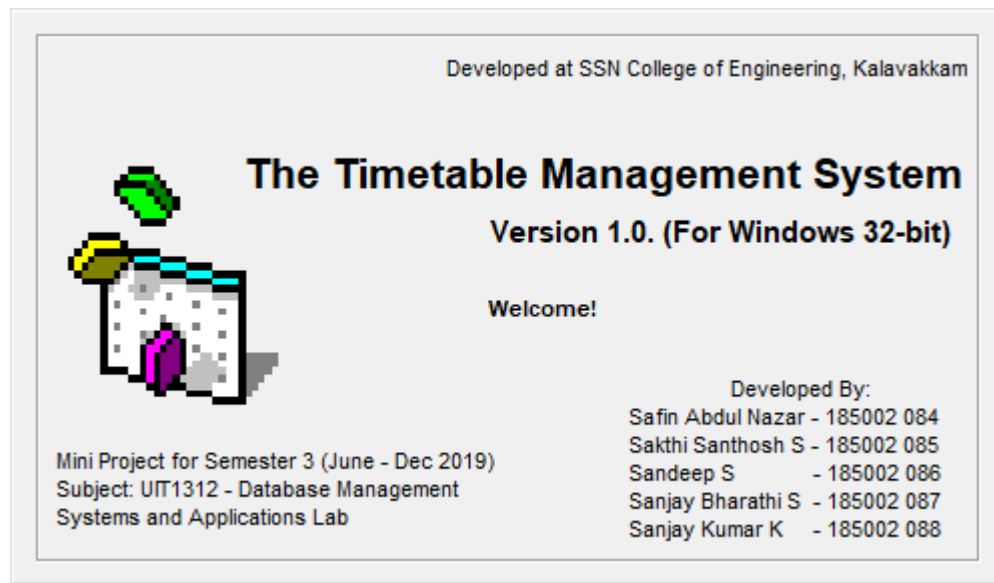
FRMSPLASH :

```
Option Explicit
Dim rst As Recordset
Dim con As Connection
Dim ctr As Integer
Private Sub Form_Load()
    On Error Resume Next
    Call initDBLoginCredentials
    ctr = 0
    Set con = CreateObject("ADODB.Connection")
    Set rst = CreateObject("ADODB.Recordset")
    lblStatus.Caption = "Status: Verifying connection to Oracle Database..."
    conString = "Provider=MSDAORA.1;Password=" & DBPassword & ";User ID=" & DBUsername & ";Data Source=" & DBName & ";Persist Security Info=True"
    con.ConnectionString = conString
    con.Open
    If con.State <> 1 Then
        MsgBox "Could not establish connection to Oracle Database! The application will now terminate.", vbCritical + vbOKOnly, "Failed to establish database connection."
        Unload Me
        Exit Sub
    Else
        lblStatus.Caption = "Status: Connected to Oracle Database successfully..."
    End If
    rst.Open "SELECT ID FROM Users", con
    If IsNull(rst("ID")) Then
        rst.Close
```

STUDENT ID: IT5086

```
        lblStatus.Caption = "Status: Setting up database for first-  
time use..."  
        Call initialize  
    End If  
    rst.Close  
    con.Close  
    lblStatus.Caption = "                Welcome!"  
    Call startTimer  
End Sub  
Function startTimer()  
    Timer1.Enabled = True  
End Function  
Function initialize()  
    con.Execute "CREATE TABLE Users (ID VARCHAR2(10) PRIMARY  
KEY, Name VARCHAR2(30), Password VARCHAR2(10), Role VARCHAR2(15),  
DateUserCreated DATE, LastLoginTime TIMESTAMP, LastLogoutTime  
TIMESTAMP)"  
    con.Execute "CREATE TABLE UserActionsLog (ID VARCHAR2(10)  
REFERENCES Users(ID), ActionsLog VARCHAR2(250))"  
    con.Execute "CREATE TABLE Classes (ID VARCHAR2(10) PRIMARY  
KEY, dept VARCHAR2(7), year NUMBER, Section CHAR(1))"  
    con.Execute "CREATE TABLE Subjects (SubCode VARCHAR2(7),  
SubName VARCHAR2(30), FacultyID VARCHAR2(10) REFERENCES Users(ID),  
ClassID VARCHAR2(10) REFERENCES Classes(ID), Credits NUMBER)"  
    con.Execute "CREATE TABLE Timetables (ClassID VARCHAR2(10)  
UNIQUE REFERENCES Classes(ID), TTString VARCHAR2(10))"  
  
    Call createUser("system", "SYSTEM (RECOVERY ACCOUNT)",  
"admin", "Administrator", "SYSTEM", con)  
End Function  
Private Sub Timer1_Timer()  
    ctr = ctr + 1  
    If ctr = 4 Then  
        Load frmLogin  
        frmLogin.Show  
        Unload Me  
    End If  
End Sub
```

SCREENSHOT :



FRMSUBJECTDETAILS :

```
Dim con As New ADODB.Connection
Dim currClassID As String
Dim rst As New ADODB.Recordset
Option Explicit
Private Sub cbxClasses_Change()
    Me.Caption = "List of All Subjects For Class: " &
    cbxClasses.Text
    Call getCurrClassID
    Call refreshSubjectsList
End Sub
Private Sub cmdRefresh_Click()
    Call getCurrClassID
    Call refreshSubjectsList
End Sub
Function refreshSubjectsList()
    Dim temp As New Recordset
    temp.Open "SELECT SubCode AS Subject_Code, SubName AS
    Subject_Name, FacultyID as Faculty_ID, Credits AS No_of_Credits
    FROM Subjects WHERE ClassID = '" & currClassID & "'", con
    Set DataGrid1.DataSource = temp
End Function
Function getCurrClassID()
    If rst.State = 1 Then
        rst.Close
    End If
    rst.Open "SELECT * FROM Classes", con
    Do While rst.EOF = False
        If rst("dept") & " - " & rst("section") & " (Year: " &
        rst("year") & ")" = cbxClasses.Text Then
            currClassID = rst("ID")
        End If
    Loop
End Function
```


STUDENT ID: IT5086

```
        Exit Do
    End If
    rst.MoveNext
Loop
End Function
Private Sub Form_Load()
    If con.State <> 1 Then
        con.ConnectionString = conString
        con.Open
    End If
    con.CursorLocation = adUseClient
    If rst.State = 1 Then
        rst.Close
    End If
    con.ConnectionString = conString
    con.Open
    con.CursorLocation = adUseClient
    rst.Open "SELECT * FROM Classes", con
    Do While rst.EOF = False
        cbxClasses.AddItem rst("dept") & " - " & rst("section") & "
(Year: " & rst("year") & ")"
        rst.MoveNext
    Loop
    rst.Close
    If currURole = "Administrator" Then
        Labell.Visible = False
        cmdRefresh.Visible = False
        cbxClasses.Visible = False
        rst.Open "SELECT * FROM Subjects", con
        frmsubjectdetails.Caption = "List of all Subjects stored in
the database"
        Set DataGrid1.DataSource = rst
    ElseIf currURole = "Faculty" Then
        Labell.Visible = False
        cmdRefresh.Visible = False
        cbxClasses.Visible = False
        rst.Open "SELECT SubCode, SubName, ClassID, Credits FROM
Subjects WHERE FacultyID='" & currUID & "'", con
        frmsubjectdetails.Caption = "List of all Subjects associated
with User ID: " & currUID
        Set DataGrid1.DataSource = rst
    End If
End Sub
```

STUDENT ID: IT5086

SCREENSHOT :

List of all Subjects stored in the database

List of subjects stored in database:

SUBCODE	SUBNAME	FACULTYID	CLASSID	CREDITS
UIT1301	Digital Electronics	faculty1	ITB2	4
UIT1302	Data Structures	faculty2	ITB2	3

FRMTIMETABLEGRID:

```
Dim con As New ADODB.Connection
Dim rst As New ADODB.Recordset, rstSubCodes As New ADODB.Recordset
Dim tableString As String
Dim updFlag As Boolean

Private Sub cmdaddNew_Click()
    On Error Resume Next
    If rst.RecordCount < 6 Then
        rst.AddNew
        cmdaddNew.Enabled = True
        dgTT.Enabled = True
    Else
        MsgBox "Cannot have more than five working days in a week!",
vbExclamation + vbOKOnly, "Can't insert new row!"
        cmdaddNew.Enabled = False
    End If
End Sub

Private Sub cmdCancel_Click()
    If MsgBox("Confirm Exit? Unsaved changes (if any) will be
lost.", vbExclamation + vbYesNo) = vbYes Then
        If con.State = 1 Then
            con.Close
            If rst.State = 1 Then
                rst.CancelUpdate
                rst.Cancel
            End If
        End If
        Unload Me
    End If
End Sub
```

STUDENT ID: IT5086

```
Private Sub cmdUpdate_Click()  
rstSubCodes.MoveFirst  
    Dim ok As Boolean  
    ok = False  
    Do While rstSubCodes.EOF = False  
        If dgTT.Text = rstSubCodes("sc") Then  
            ok = True  
            Exit Do  
        End If  
        rstSubCodes.MoveNext  
    Loop  
    If ok = False Then  
        MsgBox "Invalid entry detected! Please enter only a subject  
code valid for this class!", vbExclamation  
        MsgBox "Failed to update database because of incorrect  
value(s)!", vbCritical  
        rst.CancelUpdate  
    Else  
        updFlag = True  
        rst.UpdateBatch  
        rst.Requery  
        MsgBox "Changes saved successfully.", vbInformation  
    End If  
End Sub
```

```
Private Sub dgTT_BeforeUpdate(Cancel As Integer)  
rstSubCodes.MoveFirst  
    Dim ok As Boolean  
    ok = False  
    Do While rstSubCodes.EOF = False  
        If dgTT.Text = rstSubCodes("sc") Then  
            ok = True  
            Exit Do  
        End If  
        rstSubCodes.MoveNext  
    Loop  
    If ok = False Then  
        MsgBox "Invalid entry '" & dgTT.Text & "' detected at last  
inserted location! Please enter only a subject code valid for this  
class!", vbExclamation  
        dgTT.Text = ""  
    Else  
        If updFlag = False Then  
            rst.CancelUpdate  
        End If  
    End If  
End Sub
```

```
Private Sub Form_Load()  
    On Error Resume Next
```

STUDENT ID: IT5086

```
Set con = New ADODB.Connection
Set rst = New ADODB.Recordset
Set rstSubCodes = New ADODB.Recordset

con.Open "Provider=MSDAORA.1;Password=admin;User ID=system;Data
Source=TTMSDB;Persist Security Info=True"
con.CursorLocation = adUseClient

rst.Open "SELECT * FROM " & TTString, con, adOpenDynamic,
adLockOptimistic

If rst.EOF = True Then
    cmdUpdate.Enabled = False
    dgTT.Enabled = False
End If

If IsNull(rst("p1")) Then
    MsgBox "Timetable not found! Please ensure that you have
created valid faculty users, class and subject entries before
creating a timetable!", vbExclamation
    rstSubCodes.Close
    con.Close
    Unload Me
    Exit Sub
End If
rstSubCodes.Open "SELECT DISTINCT SubCode as sc FROM Subjects
WHERE ClassID='" & Mid(TTString, 3) & "'", con

If rstSubCodes.EOF = True Then
    MsgBox "Please add some subjects for the current class to
edit/create timetable!", vbExclamation
    rstSubCodes.Close
    con.Close
    Unload Me
    Exit Sub
End If
If rst.RecordCount < 6 Then
    cmdaddNew.Enabled = True
Else
    cmdaddNew.Enabled = False
End If
Set dgTT.DataSource = rst
Labell1.Caption = Labell1.Caption & Mid(TTString, 3)
Me.Caption = "Edit Timetable details for class ID: " &
Mid(TTString, 3)
End Sub
```

SCREENSHOT :

Edit Timetable details for class ID: ITB2

Timetable For Class ID: ITB2

Day	Create or Edit Timetable							
	P1	P2	P3	P4	P5	P6	P7	P8
Mon	UIT1301	UIT1301	UIT1302	UIT1302	UIT1301	UIT1302	UIT1301	UIT1301
Tue	UIT1302	UIT1301	UIT1302	UIT1301	UIT1301	UIT1302	UIT1302	UIT1301
Wed	UIT1301	UIT1302	UIT1302	UIT1301	UIT1302	UIT1302	UIT1301	UIT1301
Thur	UIT1302	UIT1301	UIT1301	UIT1301	UIT1302	UIT1302	UIT1301	UIT1301
Fri	UIT1301	UIT1301	UIT1302	UIT1302	UIT1301	UIT1301	UIT1302	UIT1302

Please Note: While updating each row, please click the 'Save Changes' button to avoid problems with the DataGridControl.

FRMTIMETABLEGRIDVIEW

```

Dim con As New ADODB.Connection
Dim rst As New ADODB.Recordset, rstSubCodes As New ADODB.Recordset
Dim tableString As String
Private Sub cmdCancel_Click()
    Unload Me
End Sub
Private Sub Form_Load()
    Set con = New ADODB.Connection
    Set rst = New ADODB.Recordset
    Set rstSubCodes = New ADODB.Recordset
    con.Open "Provider=MSDAORA.1;Password=admin;User ID=system;Data
Source=TTMSDB;Persist Security Info=True"
    con.CursorLocation = adUseClient
    rst.Open "SELECT * FROM " & TTString, con
    Set dgTT.DataSource = rst
    Label1.Caption = Label1.Caption & Mid(TTString, 3)
    frmTimetableGridView.Caption = frmTimetableGridView.Caption &
Mid(TTString, 3)
End Sub

```

SCREENSHOT :

Timetable for Class ID: ITB2

Timetable For Class ID: ITB2

Day	Create or Edit Timetable							
	P1	P2	P3	P4	P5	P6	P7	P8
Mon	UIT1301	UIT1301	UIT1302	UIT1302	UIT1301	UIT1302	UIT1301	UIT1301
Tue	UIT1302	UIT1301	UIT1302	UIT1301	UIT1301	UIT1302	UIT1302	UIT1301
Wed	UIT1301	UIT1302	UIT1302	UIT1301	UIT1302	UIT1302	UIT1301	UIT1301
Thur	UIT1302	UIT1301	UIT1301	UIT1301	UIT1302	UIT1302	UIT1301	UIT1301
Fri	UIT1301	UIT1301	UIT1302	UIT1302	UIT1301	UIT1301	UIT1302	UIT1302

FRMUPDATEPASSWORD:

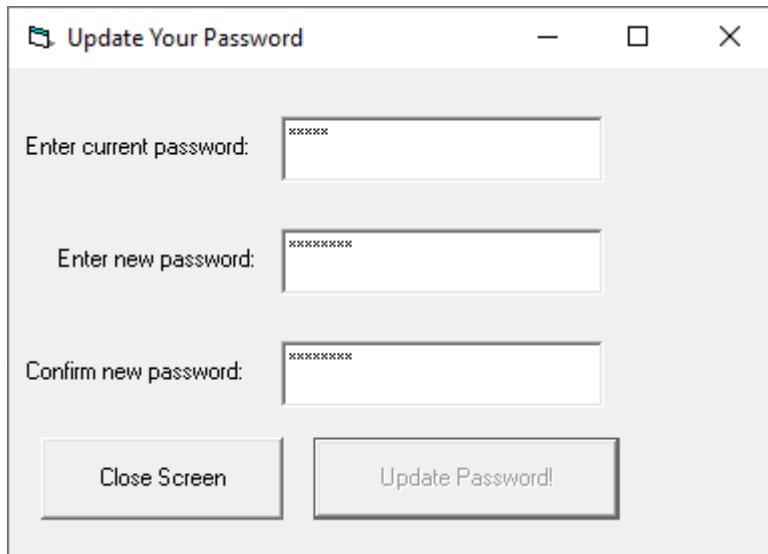
```
Dim con As Connection
Dim rst As Recordset
Private Sub cmdCancel_Click()
    If MsgBox("Close screen?", vbQuestion + vbYesNo, "Confirmation")
= vbYes Then
        If con.State = 1 Then
            con.Close
        End If
        Me.Hide
        Unload Me
    End If
End Sub
Private Sub cmdUpdate_Click()
    cmdUpdate.Enabled = False
    '    On Error GoTo errorHandler
    cmdUpdate.Enabled = False
    If txtPwd.Text = "" Or txtConfirmPwd.Text = "" Then
        MsgBox "Please fill your new password in both the textboxes
provided!", vbExclamation + vbOKOnly
        Exit Sub
    End If
    If Len(txtPwd.Text) > 10 Then
        MsgBox "The entered password is too long! Maximum allowed:
10 characters.", vbExclamation + vbOKOnly, "Invalid User ID!"
        Exit Sub
    End If
    If IsNotCompatibleString(txtPwd.Text) Then
        MsgBox "The password contains special characters, which are
not allowed! Please enter only alphanumeric characters!",
vbExclamation + vbOKOnly, "Invalid value(s) entered!"
        Exit Sub
    End If
    If txtPwd.Text <> txtConfirmPwd.Text Then
        MsgBox "Please enter identical values for both the new
password textboxes!", vbExclamation + vbOKOnly, "Invalid value(s)
entered!"
        Exit Sub
    End If
    If rst.State = 1 Then
        rst.Close
    End If
    rst.Open "SELECT Password FROM Users WHERE ID = '" & currUID
& "'", con
    If txtPwd.Text = txtCurrentPwd.Text Then
        MsgBox "Your new password cannot be the same as your
existing password! Please enter a new password!", vbExclamation +
vbOKOnly, "Invalid value(s) entered!"
        Exit Sub
    End If
```

```

End If
If txtCurrentPwd.Text <> rst("Password") Then
    MsgBox "The entered value for the current password does not
match the entry stored in the database! Please enter the correct
value!", vbExclamation + vbOKOnly, "Invalid value(s) entered!"
    rst.Close
    Exit Sub
End If
If MsgBox("Changes made once cannot be rolled back after
successful update. Continue with updation?", vbExclamation +
vbYesNo, "Confirm updation") = vbYes Then
    Call updatePassword(currUID, txtPwd.Text, con)
    MsgBox "Your password was updated successfully.",
vbInformation, "Updation Successful"
End If
Exit Sub
'errorHandler:
'    MsgBox "A critical error has occurred! The operation will now
rollback and terminate!", vbCritical + vbOKOnly, "Critical Error"
End Sub
Function establishDBConnection()
    On Error GoTo errorHandler
    Set con = CreateObject("ADODB.Connection")
    con.ConnectionString = conString
    con.Open
    Set rst = CreateObject("ADODB.Recordset")
    Exit Function
errorHandler:
    con.Close
    MsgBox "The connection to the Oracle database failed! The
application will now terminate.", vbCritical + vbOKOnly, "Login
Failed!"
    Unload Me
End Function
Private Sub Form_Load()
    Call establishDBConnection
End Sub
Private Sub txtConfirmPwd_Change()
    cmdUpdate.Enabled = True
End Sub
Private Sub txtCurrentPwd_Change()
    cmdUpdate.Enabled = True
End Sub
Private Sub txtPwd_Change()
    cmdUpdate.Enabled = True
End Sub

```

SCREENSHOT :



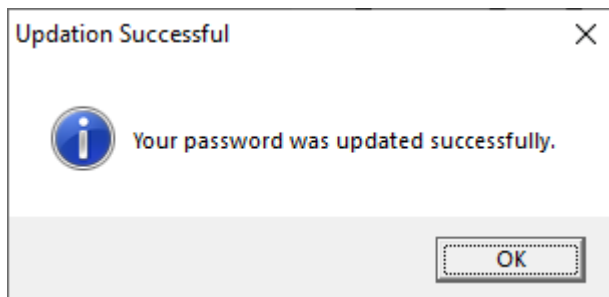
Update Your Password

Enter current password: xxxxxx

Enter new password: xxxxxxxx

Confirm new password: xxxxxxxx

Close Screen Update Password!



FRMUPDATEUSER:

```
Dim con As Connection
Dim rst As Recordset
Dim changesMade As Boolean
Private Sub cbxRole_Change()
    cmdRollback.Enabled = True
End Sub
Private Sub cmdCancel_Click()
    If MsgBox("Close screen?", vbQuestion + vbYesNo, "Confirmation") = vbYes Then
        If con.State = 1 Then
            con.Close
        End If
        Me.Hide
        Unload Me
    End If
End Sub

Private Sub cmdClear_Click()
    If MsgBox("Are you sure you want to clear all values?", vbQuestion + vbYesNo, "Clear All Values?") = vbYes Then
        txtPassword.Text = ""
        txtUName.Text = ""
    End If
```


STUDENT ID: IT5086

```
End Sub
Private Sub cmdRollback_Click()
    rst.Open "SELECT Name, Password, Role FROM Users WHERE ID = '" &
txtUID.Text & "'"
    txtUName.Text = rst("Name")
    txtPassword.Text = rst("Password")
    cbxRole.Text = rst("Role")
    cmdUpdate.Enabled = False
    rst.Close
End Sub
Private Sub cmdUpdate_Click()
    cmdUpdate.Enabled = False
    If txtUID.Text = "" Or txtUName.Text = "" Or txtPassword.Text =
"" Then
        MsgBox "One or more required values (Username or Password)
textboxes are empty! Please fill them up before continuing!",
vbExclamation + vbOKOnly, "Invalid User ID!"
        Exit Sub
    End If
    If Len(txtUName.Text) > 30 Then
        MsgBox "The entered username is too long! Maximum allowed:
30 characters.", vbExclamation + vbOKOnly, "Invalid User ID!"
        Exit Sub
    End If
    If Len(txtPassword.Text) > 10 Then
        MsgBox "The entered password is too long! Maximum allowed:
10 characters.", vbExclamation + vbOKOnly, "Invalid User ID!"
        Exit Sub
    End If
    If IsNotCompatibleString(txtUName.Text) Or
IsNotCompatibleString(txtPassword.Text) Then
        MsgBox "The username or password contains special
characters, which are not allowed! Please enter only alphanumeric
characters!", vbExclamation + vbOKOnly, "Invalid value(s) entered!"
        Exit Sub
    End If
    If MsgBox("Changes made once cannot be rolled back after
successful update. Continue with updation?", vbExclamation +
vbYesNo, "Confirm updation") = vbYes Then
        Call updateUser(txtUID.Text, txtUName.Text,
txtPassword.Text, cbxRole.Text, currUID, con)
        MsgBox "The user's details were updated successfully.",
vbInformation, "Updation Successful"
        cmdRollback.Enabled = False
    End If
    Exit Sub
ErrorHandler:
    MsgBox "A critical error has occurred! The operation will now
rollback and terminate!", vbCritical + vbOKOnly, "Critical Error"
End Sub
Function establishDBConnection()
```

STUDENT ID: IT5086

```
On Error GoTo errorHandler
Set con = CreateObject("ADODB.Connection")
con.ConnectionString = conString
con.Open
Set rst = CreateObject("ADODB.Recordset")
Exit Function
errorHandler:
    con.Close
    MsgBox "The connection to the Oracle database failed! The
application will now terminate.", vbCritical + vbOKOnly, "Login
Failed!"
    Unload Me
End Function
Private Sub cmdSearch_Click()
    If txtUID.Text = "system" Then
        MsgBox "Cannot edit system recovery account! Please enter
any other valid User ID!", vbCritical + vbOKOnly, "Cannot edit
system account"
        Exit Sub
    ElseIf txtUID.Text = currUID Then
        MsgBox "Cannot edit current user from the same account!
Please enter any other valid User ID!", vbCritical + vbOKOnly,
"Cannot edit your own account"
        Exit Sub
    ElseIf txtUID.Text = "" Then
        MsgBox "User ID field is empty! Please enter a valid User ID
to continue!", vbExclamation + vbOKOnly, "Cannot edit your own
account"
        Exit Sub
    End If
    txtUID.Enabled = False
    okflag = 0
    rst.Open "SELECT ID,Name,Password,Role FROM Users WHERE ID = '"
& txtUID.Text & "'", con
    If rst.EOF = False Then
        Call showControls
        txtUName.Text = rst("Name")
        txtPassword.Text = rst("Password")
        cbxRole.Text = rst("Role")
    Else
        MsgBox "The record with the specified user ID does not exist
in the database! Please enter a valid User ID!", vbExclamation +
vbOKOnly, "Error!"
        Call hideControls
    End If
    cmdRollback.Enabled = False
    rst.Close
End Sub
Private Sub Form_Load()
    Call establishDBConnection
    Call hideControls
```

STUDENT ID: IT5086

```
        cbxRole.AddItem "Student"
        cbxRole.AddItem "Faculty"
        cbxRole.AddItem "Administrator"
        cbxRole.Text = cbxRole.List(0)
End Sub
Function hideControls()
    Label2.Visible = False
    Label3.Visible = False
    Label5.Visible = False
    txtUName.Visible = False
    txtPassword.Visible = False
    cbxRole.Visible = False
    cmdUpdate.Visible = False
    cmdRollback.Visible = False
End Function
Function showControls()
    Label2.Visible = True
    Label3.Visible = True
    Label5.Visible = True
    txtUName.Visible = True
    txtPassword.Visible = True
    cbxRole.Visible = True
    cmdUpdate.Visible = True
    cmdRollback.Visible = True
End Function
Private Sub txtPassword_Change()
    cmdRollback.Enabled = True
    cmdUpdate.Enabled = True
End Sub
Private Sub txtUID_Change()
    cmdRollback.Enabled = True
    cmdUpdate.Enabled = True
End Sub
Private Sub txtUName_Change()
    cmdRollback.Enabled = True
    cmdUpdate.Enabled = True
End Sub
```

SCREENSHOT :

Update User Details

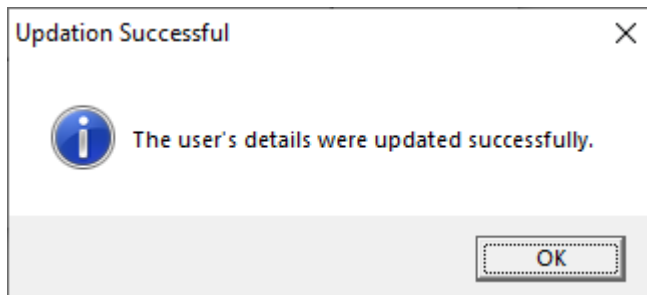
Enter User ID to obtain information to edit:

User ID:

Name of the User:

Password:

Role:



FRMVIEWLOGS :

```
Private Sub Form_Load()  
    Dim con As New Connection, rst As New Recordset  
    con.ConnectionString = conString  
    con.Open  
    con.CursorLocation = adUseClient  
    If currUID <> "system" Then  
        rst.Open "SELECT ActionsLog as Your_Actions_History FROM  
UserActionsLog WHERE ID = '" & currUID & "'", con  
        frmViewLogs.Caption = "Action History Log for User ID: " &  
currUID  
    Else  
        rst.Open "SELECT ID as UserID, ActionsLog as ActionHistory  
FROM UserActionsLog", con  
        frmViewLogs.Caption = "Action History Log for all Users"  
    End If  
    Set dgLogs.DataSource = rst  
End Sub
```

SCREENSHOT :

Action History Log for User ID: faculty1	
YOUR ACTIONS HISTORY	
▶	USER CREATED ON: 12-OCT-2019 05:03:52 PM BY User ID: system
	Logged in on: 12-OCT-2019 05:05:19 PM
	Password changed by: faculty1 on: 12-OCT-2019 05:05:50 PM

Action History Log for all Users	
USERID	ACTIONHISTORY
▶ system	USER CREATED ON: 12-OCT-2019 04:33:07 PM BY User ID: SYSTEM
system	Logged in on: 12-OCT-2019 04:33:19 PM
system	Deleted Subject UIT1302 ON: 12-OCT-2019 05:02:42 PM
faculty2	USER CREATED ON: 12-OCT-2019 05:04:27 PM BY User ID: system
system	Created Class ITB2 ON: 12-OCT-2019 04:33:57 PM
system	Created Subject UIT1301-Digital Electronics ON: 12-OCT-2019 04:35:12 PM
system	Created Subject UIT1302-Data Structures ON: 12-OCT-2019 04:35:42 PM
system	Created Timetable tltB2 ON: 12-OCT-2019 04:39:55 PM
system	Logged in on: 12-OCT-2019 04:47:22 PM
system	Logged in on: 12-OCT-2019 04:50:53 PM
system	Deleted Subject UIT1301 ON: 12-OCT-2019 05:02:38 PM
system	Deleted user: faculty1 on: 12-OCT-2019 05:02:38 PM
system	Deleted user: faculty2 on: 12-OCT-2019 05:02:42 PM
faculty1	USER CREATED ON: 12-OCT-2019 05:03:52 PM BY User ID: system
system	Logged out on: 12-OCT-2019 05:05:12 PM
faculty1	Logged in on: 12-OCT-2019 05:05:19 PM
faculty1	Password changed by: faculty1 on: 12-OCT-2019 05:05:50 PM
faculty1	Logged out on: 12-OCT-2019 05:09:08 PM
system	Logged in on: 12-OCT-2019 05:09:11 PM

FRMVIEWUSERS :

```

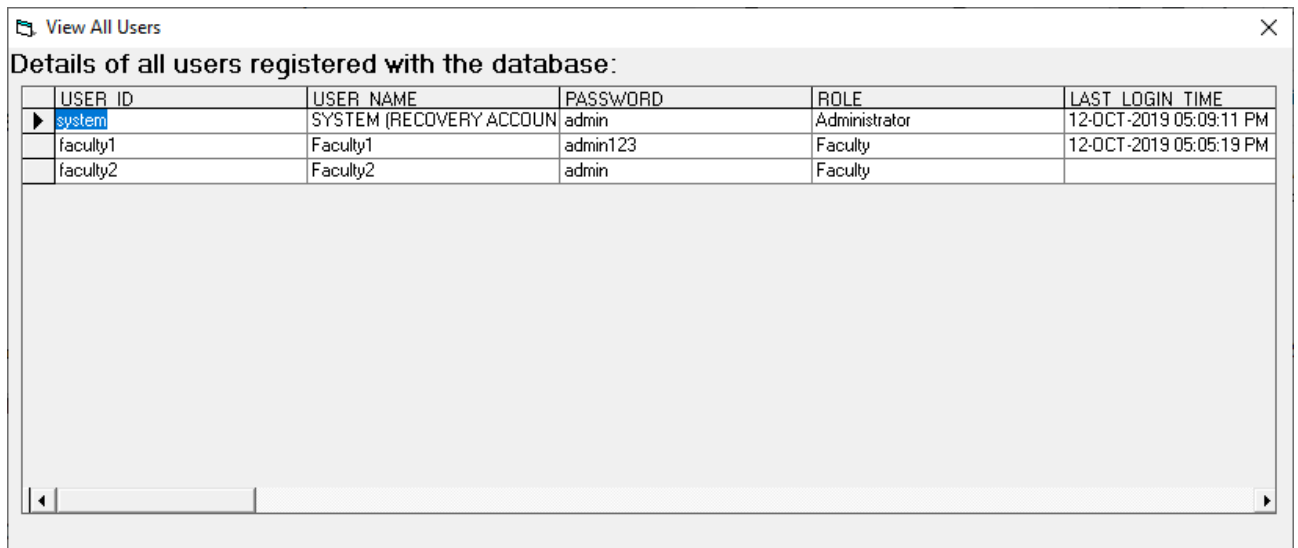
Private Sub Form_Load()
    Dim con As New Connection, rst As New Recordset
    con.ConnectionString = conString
    con.Open
    con.CursorLocation = adUseClient
    If currURole <> "Administrator" Then
        rst.Open "SELECT Name AS User_Name, Role,
TO_CHAR>LastLoginTime, 'DD-MON-YYYY hh:MI:SS AM') AS
Last_Login_Time, TO_CHAR>LastLogoutTime, 'DD-MON-YYYY hh:MI:SS AM')
AS Last_Logout_Time FROM Users", con
        frmViewUsers.Caption = "List of all Users (filtered): " &
currUID
    Else

```

STUDENT ID: IT5086

```
rst.Open "SELECT ID AS User_ID, Name AS User_Name, Password,
Role, TO_CHAR>LastLoginTime, 'DD-MON-YYYY hh:MI:SS AM') AS
Last_Login_Time, TO_CHAR>LastLogoutTime, 'DD-MON-YYYY hh:MI:SS AM')
AS Last_Logout_Time FROM Users", con
frmViewLogs.Caption = "Action History Log for all Users"
End If
Set dgUsers.DataSource = rst
End Sub
```

SCREENSHOT :



The screenshot shows a window titled "View All Users" with a close button (X) in the top right corner. Below the title bar, there is a header "Details of all users registered with the database:". Underneath the header is a table with five columns: USER ID, USER NAME, PASSWORD, ROLE, and LAST LOGIN TIME. The table contains three rows of data. The first row is highlighted with a blue background. Below the table, there is a large empty rectangular area. At the bottom of the window, there is a horizontal scrollbar.

USER ID	USER NAME	PASSWORD	ROLE	LAST LOGIN TIME
system	SYSTEM (RECOVERY ACCOUNT)	admin	Administrator	12-OCT-2019 05:09:11 PM
faculty1	Faculty1	admin123	Faculty	12-OCT-2019 05:05:19 PM
faculty2	Faculty2	admin	Faculty	

MODULES :

MODVARS .BAS

```
Global conString As String
Global canSeeTT As Boolean
Global editTT As Boolean
Global lastLoginTime As String
Global lastLogoutTime As String
Global DBPassword As String
Global DBUsername As String
Global currUName As String
Global TTString As String
Global currURole As String
Global currUID As String
Global DBName As String
Global year As String
Global dept As String
Global sec As String
Option Explicit

Function initDBLoginCredentials()
    DBPassword = "admin"
    DBUsername = "system"
    DBName = "TTMSDB"
End Function
```

STUDENT ID: IT5086

```
Function createUser(UID As String, UName As String, Upassword As String, URole As String, UCreatorID As String, con As Connection)
    con.Execute "INSERT INTO Users VALUES('" & UID & "', '" & UName & "', '" & Upassword & "', '" & URole & "', SYSDATE, NULL, NULL)"
```

```
    con.Execute "INSERT INTO UserActionsLog VALUES ('" & UID & "', CONCAT(CONCAT('USER CREATED ON: ', TO_CHAR(CURRENT_TIMESTAMP, 'DD-MON-YYYY hh:MI:SS AM')), ' BY User ID: " & UCreatorID & "'))"
End Function
```

```
Function.createClass(dept As String, sec As String, year As String, UCreatorID As String, con As Connection)
    con.Execute "INSERT INTO Classes VALUES('" & dept & sec & year & "', '" & dept & "', " & year & ", '" & sec & "')"

```

```
    con.Execute "INSERT INTO UserActionsLog VALUES ('" & UCreatorID & "', CONCAT('Created Class " & dept & sec & year & " ON: ', TO_CHAR(CURRENT_TIMESTAMP, 'DD-MON-YYYY hh:MI:SS AM')))"
End Function
```

```
Function.createSubject(SubCode As String, SubName As String, facultyID As String, classID As String, credits As String, UCreatorID As String, con As Connection)
    con.Execute "INSERT INTO Subjects VALUES('" & SubCode & "', '" & SubName & "', '" & facultyID & "', '" & classID & "', TO_NUMBER('" & credits & "'))"

```

```
    con.Execute "INSERT INTO UserActionsLog VALUES ('" & UCreatorID & "', CONCAT('Created Subject " & SubCode & "-" & SubName & " ON: ', TO_CHAR(CURRENT_TIMESTAMP, 'DD-MON-YYYY hh:MI:SS AM')))"
End Function
```

```
Function.createTimetable(classID As String, tt As String, UCreatorID As String, con As Connection)
```

```
    con.Execute "INSERT INTO Timetables VALUES('" & classID & "', '" & tt & "')"

```

```
    con.Execute "CREATE TABLE " & tt & " (p1 VARCHAR2(7), p2 VARCHAR2(7), p3 VARCHAR2(7), p4 VARCHAR2(7), p5 VARCHAR2(7), p6 VARCHAR2(7), p7 VARCHAR2(7), p8 VARCHAR2(7))"

```

```
    con.Execute "INSERT INTO UserActionsLog VALUES ('" & UCreatorID & "', CONCAT('Created Timetable " & tt & " ON: ', TO_CHAR(CURRENT_TIMESTAMP, 'DD-MON-YYYY hh:MI:SS AM')))"
End Function
```

```
Function.deleteTimetable(tt As String, UCreatorID As String, con As Connection)
```

```
    con.Execute "DELETE FROM Timetables WHERE TTString = '" & tt & "'"

```

```
    con.Execute "DROP TABLE " & tt
```

STUDENT ID: IT5086

```
con.Execute "INSERT INTO UserActionsLog VALUES ('" & UCreatorID
& "', CONCAT('Deleted Timetable " & tt & " ON: ',
TO_CHAR(CURRENT_TIMESTAMP, 'DD-MON-YYYY hh:MI:SS AM')))"
End Function
```

```
Function deleteSubject(SubCode As String, UCreatorID As String, con
As Connection)
```

```
con.Execute "DELETE FROM Subjects WHERE SubCode = '" & SubCode &
"""
```

```
con.Execute "INSERT INTO UserActionsLog VALUES ('" & UCreatorID
& "', CONCAT('Deleted Subject " & SubCode & " ON: ',
TO_CHAR(CURRENT_TIMESTAMP, 'DD-MON-YYYY hh:MI:SS AM')))"
End Function
```

```
Function updateUser(UID As String, UName As String, Upassword As
String, URole As String, UCreatorID As String, con As Connection)
```

```
con.Execute "UPDATE Users SET Name = '" & UName & "', Password =
'" & Upassword & "', Role = '" & URole & "' WHERE ID = '" & UID &
"""
```

```
con.Execute "INSERT INTO UserActionsLog VALUES ('" & UID & "',
CONCAT('Details changed by: " & UCreatorID & " on: ',
TO_CHAR(CURRENT_TIMESTAMP, 'DD-MON-YYYY hh:MI:SS AM')))"
```

```
con.Execute "INSERT INTO UserActionsLog VALUES ('" & UCreatorID
& "', CONCAT('Details changed for: " & UID & " on: ',
TO_CHAR(CURRENT_TIMESTAMP, 'DD-MON-YYYY hh:MI:SS AM')))"
End Function
```

```
Function deleteUser(UID As String, UCreatorID As String, con As
Connection)
```

```
Dim rst As New Recordset
rst.Open "SELECT SubCode FROM Subjects WHERE FacultyID='" & UID
& "'", con
```

```
If Not IsNull(rst("SubCode")) Then
```

```
Do While rst.EOF = False
```

```
Call deleteSubject(rst("SubCode"), UCreatorID, con)
```

```
rst.MoveNext
```

```
Loop
```

```
End If
```

```
con.Execute "DELETE FROM UserActionsLog WHERE ID = '" & UID &
"""
```

```
con.Execute "DELETE FROM Users WHERE ID = '" & UID & """
```

```
con.Execute "INSERT INTO UserActionsLog VALUES ('" & UCreatorID
& "', CONCAT('Deleted user: " & UID & " on: ',
TO_CHAR(CURRENT_TIMESTAMP, 'DD-MON-YYYY hh:MI:SS AM')))"
```

```
End Function
```


STUDENT ID: IT5086

```
Function deleteClass(classID As String, UCreatorID As String, con As Connection)
```

```
    Dim rst1 As New Recordset
    rst1.Open "SELECT SubCode FROM Subjects WHERE ClassID='" & classID & "'", con
```

```
    If Not IsNull(rst1("SubCode")) Then
        Do While rst1.EOF = False
            Call deleteSubject(rst1("SubCode"), UCreatorID, con)
            rst1.MoveNext
        Loop
    End If
```

```
    con.Execute "DELETE FROM Timetables WHERE classID = '" & classID & "'"
```

```
    con.Execute "DELETE FROM Classes WHERE ID = '" & classID & "'"
```

```
    con.Execute "INSERT INTO UserActionsLog VALUES ('" & UCreatorID & "', CONCAT('Deleted class: " & classID & " on: ', TO_CHAR(CURRENT_TIMESTAMP, 'DD-MON-YYYY hh:MI:SS AM')))"
End Function
```

```
Function updatePassword(UID As String, Upassword As String, con As Connection)
```

```
    con.Execute "UPDATE Users SET Password = '" & Upassword & "' WHERE ID = '" & UID & "'"
```

```
    con.Execute "INSERT INTO UserActionsLog VALUES ('" & UID & "', CONCAT('Password changed by: " & UID & " on: ', TO_CHAR(CURRENT_TIMESTAMP, 'DD-MON-YYYY hh:MI:SS AM')))"
```

```
End Function
```

```
Function updateLoginTimeForUser(UID As String, con As Connection)
```

```
    con.Execute "UPDATE Users SET LastLoginTime = CURRENT_TIMESTAMP WHERE ID = '" & UID & "'"
```

```
    con.Execute "INSERT INTO UserActionsLog VALUES ('" & UID & "', CONCAT('Logged in on: ', TO_CHAR(CURRENT_TIMESTAMP, 'DD-MON-YYYY hh:MI:SS AM')))"
```

```
End Function
```

```
Function updateLogoutTimeForUser(UID As String, con As Connection)
```

```
    con.Execute "UPDATE Users SET LastLogoutTime = CURRENT_TIMESTAMP WHERE ID = '" & UID & "'"
```

```
    con.Execute "INSERT INTO UserActionsLog VALUES ('" & UID & "', CONCAT('Logged out on: ', TO_CHAR(CURRENT_TIMESTAMP, 'DD-MON-YYYY hh:MI:SS AM')))"
```

```
End Function
```

STUDENT ID: IT5086

```
Public Function IsNotCompatibleString(s As String) As Boolean
    Dim flg As Boolean, i As Integer
    If InStr(1, s, ";") <> 0 Or InStr(1, s, "'") <> 0 Then
        IsNotCompatibleString = True
        Exit Function
    End If
    IsNotCompatibleString = False
End Function
```