

- [Home](#)
- [Registration](#)
- [Catalogue Update](#)
- [Help](#)

ID	Description	Finish	Price	
1	End Table	Cherry	210.00	Order & View
2	Coffee Table	Natural Ash	200.00	Order & View
3	Computer Desk	Natural Ash	375.00	Order & View
4	Entertainment Center	Natural Maple	650.00	Order & View
5	Writers Desk	Cherry	325.00	Order & View
6	8-Drawer Desk	White Ash	750.00	Order & View
7	Dining Table	Natural Ash	900.00	Order & View
8	Computer Desk	Walnut	250.00	Order & View

Login

Username:

Password:

New Customer? [Registor](#)

Customer Registration Form

Customer id:

Customer Name:

Address:

City:

State:

Postal Address:

Username:

Password:

Note: Data could only be updated if customer is already registered

Selected Product Details

Product: Computer Desk

Finish: Natural Ash

Unit Price: \$375

Customer ID:

Quantity:

Total: \$375.00

Proceed to Payment

Finalize Order

Select Payment Method:

Bank Card Cash On Delivery

Card Details

Place Order Now

localhost:56859 says

Order Successful! Your Order ID is: 1016

OK

Employee Login Page

Username:

Password:

Login

Update Catalogue

Enter Product ID

Enter Product Line ID

Enter Product Description

Enter Product Finish

Enter Product Price

Confirm Update

Employee Information Form

Employee ID (SSN):

Full Name:

Address:

City:

State:

Zip Code:

Birth Date:

Date Hired:

Supervisor ID:

Username:

Password:

Note: This form is supposed to be filled by manager (so assumption is that he would already know Emp ID and Supervisor ID)

CODE:

Home:

Imports System.Data

Imports System.Data.SqlClient

Partial Class _Default

Inherits System.Web.UI.Page

' Page Load and GridView binding

Protected Sub Page_Load(ByVal sender As Object, ByVal e As EventArgs) Handles Me.Load

If Not IsPostBack Then

' Session.Clear()

LoadCropData()

End If

End Sub

' Load all products into GridView

Private Sub LoadCropData()

**Dim connStr As String = "Server=ABDULLAH\SQLEXPRESS;Database=PVFC_DB;User
Id=FarmAdmin;Password=abcd8552;"**

**Dim query As String = "SELECT Product_Id, Product_Description, Product_Finish,
Standard_Price FROM PRODUCT_t"**

Using conn As New SqlConnection(connStr)

Using cmd As New SqlCommand(query, conn)

Dim dt As New DataTable()

Try

conn.Open()

Dim adapter As New SqlDataAdapter(cmd)

adapter.Fill(dt)

gvCrops.DataSource = dt

```

        gvCrops.DataBind()
    Catch ex As Exception
        Response.Write("Error: " & ex.Message)
    End Try
End Using
End Using
End Sub

' When user clicks "View & Order"
Protected Sub gvCrops_SelectedIndexChanged(sender As Object, e As EventArgs)
    Dim row As GridViewRow = gvCrops.SelectedRow
    Dim productId As String = gvCrops.SelectedDataKey.Value.ToString()
    Dim description As String = row.Cells(1).Text
    Dim finish As String = row.Cells(2).Text
    Dim price As Double = Convert.ToDouble(row.Cells(3).Text)

    ' Use the App_Code Product class
    Dim prod As New Product() With {
        .Product_Id = productId,
        .Product_Name = description,
        .Product_Finish = finish,
        .Price = price
    }

    ' Store in Session
    Session("SelectedProduct") = prod

    ' Redirect depending on login
    If Session("UserID") Is Nothing Then
        Response.Redirect("customerlogin.aspx")
    Else
        Response.Redirect("order.aspx")
    End If
End Sub

' Search functionality
Protected Sub BtnSearch_Click(sender As Object, e As EventArgs)
    Dim keyword As String = txtSearch.Text.Trim()
    Dim connStr As String = "Server=ABDULLAH\SQLEXPRESS;Database=PVFC_DB;User
Id=FarmAdmin;Password=abcd8552;"
    Dim query As String = "SELECT Product_Id, Product_Description, Product_Finish,
Standard_Price " &
        "FROM PRODUCT_t " &
        "WHERE Product_Description LIKE '%' + @Keyword + '%" " &
        "OR Product_Finish LIKE '%' + @Keyword + '%" "

    Using conn As New SqlConnection(connStr)
        Using cmd As New SqlCommand(query, conn)
            cmd.Parameters.AddWithValue("@Keyword", keyword)
            Dim dt As New DataTable()
            Try
                conn.Open()
                Dim adapter As New SqlDataAdapter(cmd)
                adapter.Fill(dt)
            End Try
        End Using
    End Using
End Sub

```

```

        gvCrops.DataSource = dt
        gvCrops.DataBind()
    Catch ex As Exception
        Response.Write("Error: " & ex.Message)
    End Try
End Using
End Using
End Sub
End Class

```

' Product class shared via Session (App_Code folder)

Order:

Partial Class OrderPage

Inherits System.Web.UI.Page

Protected Sub Page_Load(sender As Object, e As EventArgs) Handles Me.Load

If Not IsPostBack Then

' Retrieve CustomerID from session

Dim custID As String = ""

If Session("CustomerID") IsNot Nothing Then

custID = Session("CustomerID").ToString()

' Optionally pre-fill textbox

CustID_TB.Text = custID

CustID_TB.Enabled = False ' prevent editing if needed

End If

' Load product from Session

If Session("SelectedProduct") IsNot Nothing Then

Dim prod As Product = CType(Session("SelectedProduct"), Product)

lblid.Text = "Product: " & prod.Product_Name

lblDesc.Text = "Finish: " & prod.Product_Finish

lblPrice.Text = "Unit Price: \$" & prod.Price

hfPrice.Value = prod.Price.ToString()

' Calculate total

UpdateTotal()

Else

' No product selected, redirect to homepage

Response.Redirect("home.aspx")

End If

End If

End Sub

Protected Sub Quantity_TB_TextChanged(sender As Object, e As EventArgs) Handles

Quantity_TB.TextChanged

UpdateTotal()

End Sub

```

Private Sub UpdateTotal()
    Dim qty As Integer = If(IsNumeric(Quantity_TB.Text), Convert.ToInt32(Quantity_TB.Text), 1)
    Dim price As Double = If(IsNumeric(hfPrice.Value), Convert.ToDouble(hfPrice.Value), 0)
    lbltotal.Text = "Total: $" & (qty * price).ToString("0.00")
End Sub

Protected Sub btnProceed_Click(sender As Object, e As EventArgs) Handles btnProceed.Click
    ' Save order info in session
    Session("OrderQuantity") = Quantity_TB.Text
    Session("CustomerID") = CustID_TB.Text ' ensure CustomerID is available
    Response.Redirect("payment.aspx")
End Sub
End Class

```

Customer Registration:

```

Imports System.Data.SqlClient
Imports System.Configuration

```

```

Partial Class _Default
    Inherits System.Web.UI.Page

```

```

    ' Connection string remains consistent using your FarmAdmin credentials
    Dim connStr As String = "Server=ABDULLAH\SQLEXPRESS;Database=PVFC_DB;User
Id=FarmAdmin;Password=abcd8552;TrustServerCertificate=True;"

```

```

Private Sub GenerateCustomerID()
    Dim query As String = "SELECT ISNULL(MAX(Customer_Id), 0) + 1 FROM CUSTOMER_t"

    Using conn As New SqlConnection(connStr)
        Using cmd As New SqlCommand(query, conn)
            Try
                conn.Open()
                Dim newID As Integer = Convert.ToInt32(cmd.ExecuteScalar())

                TextBox8.Text = newID.ToString()
                TextBox8.ReadOnly = True ' 🔒 prevent user editing

            Catch ex As Exception
                Response.Write("Error generating ID: " & ex.Message)
            End Try
        End Using
    End Using
End Sub

```

```

Private Sub _Default_Load(sender As Object, e As EventArgs) Handles Me.Load
    If Not IsPostBack Then
        GenerateCustomerID()
    End If
End Sub

```

```
End If
End Sub
```

```
' --- SUBMIT BUTTON LOGIC (INSERT) ---
```

```
Protected Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
```

```
Dim customerQuery As String = "INSERT INTO CUSTOMER_† (Customer_Id,
Customer_Name, Customer_Address, Customer_City, Customer_State, Postal_Code) " &
"VALUES (@Id, @Name, @Address, @City, @State, @Zip)"
```

```
Dim userQuery As String = "INSERT INTO USER_† (Username, PasswordHash, RoleID,
CustomerID) " &
"VALUES (@Username, @Password, 1, @Id)"
```

```
Using conn As New SqlConnection(connStr)
```

```
conn.Open()
```

```
' Start transaction so both inserts succeed or fail together
```

```
Dim transaction As SqlTransaction = conn.BeginTransaction()
```

```
Try
```

```
' --- Insert into CUSTOMER_† ---
```

```
Using cmdCustomer As New SqlCommand(customerQuery, conn, transaction)
```

```
cmdCustomer.Parameters.AddWithValue("@Id", TextBox8.Text)
cmdCustomer.Parameters.AddWithValue("@Name", TextBox1.Text)
cmdCustomer.Parameters.AddWithValue("@Address", TextBox7.Text)
cmdCustomer.Parameters.AddWithValue("@City", TextBox2.Text)
cmdCustomer.Parameters.AddWithValue("@State", TextBox3.Text)
cmdCustomer.Parameters.AddWithValue("@Zip", TextBox4.Text)
```

```
cmdCustomer.ExecuteNonQuery()
```

```
End Using
```

```
' --- Insert into USER_† ---
```

```
Using cmdUser As New SqlCommand(userQuery, conn, transaction)
```

```
cmdUser.Parameters.AddWithValue("@Username", TextBox5.Text)
cmdUser.Parameters.AddWithValue("@Password", TextBox6.Text)
cmdUser.Parameters.AddWithValue("@Id", TextBox8.Text)
```

```
cmdUser.ExecuteNonQuery()
```

```
End Using
```

```
' Commit if both succeed
```

```
transaction.Commit()
```

```
Response.Redirect("customerlogin.aspx")
```

```
Response.Write("<script>alert('Customer " & TextBox8.Text & " Registered  
Successfully!');</script>")  
ClearForm()
```

```
Catch ex As Exception
```

```
' Rollback if anything fails  
transaction.Rollback()
```

```
Response.Write("Error: " & ex.Message)
```

```
End Try
```

```
End Using
```

```
End Sub
```

```
' --- UPDATE BUTTON LOGIC (UPDATE) ---
```

```
Protected Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
```

```
' SQL Update requires a WHERE clause to target the specific Customer_Id entered in  
TextBox8
```

```
Dim query As String = "UPDATE CUSTOMER_† Set " &  
"Customer_Name = @Name, " &  
"Customer_Address = @Address, " &  
"Customer_City = @City, " &  
"Customer_State = @State, " &  
"Postal_Code = @Zip " &  
"WHERE Customer_Id = @Id"
```

```
ExecuteDatabaseOperation(query, "Updated", False)
```

```
End Sub
```

```
' Helper method to avoid repeating code for both buttons
```

```
Private Sub ExecuteDatabaseOperation(query As String, actionName As String, clearAfter As  
Boolean)
```

```
Using conn As New SqlConnection(connStr)
```

```
Using cmd As New SqlCommand(query, conn)
```

```
' Mapping parameters from your TextBoxes
```

```
cmd.Parameters.AddWithValue("@Id", TextBox8.Text)  
cmd.Parameters.AddWithValue("@Name", TextBox1.Text)  
cmd.Parameters.AddWithValue("@Address", TextBox7.Text)  
cmd.Parameters.AddWithValue("@City", TextBox2.Text)  
cmd.Parameters.AddWithValue("@State", TextBox3.Text)  
cmd.Parameters.AddWithValue("@Zip", TextBox4.Text)
```

```
Try
```

```
conn.Open()
```

```
Dim rowsAffected As Integer = cmd.ExecuteNonQuery()
```

```
If rowsAffected > 0 Then
```

```

        Response.Write("<script>alert('Customer " & TextBox8.Text & " " & actionName & "
Successfully!');</script>")
        If clearAfter Then ClearForm()
        Else
            Response.Write("<script>alert('No record found with ID " & TextBox8.Text & " to " &
actionName.ToLower() & ".');</script>")
        End If
        Catch ex As Exception
            Response.Write("Error: " & ex.Message)
        End Try

    End Using
End Using
End Sub

```

```

Private Sub ClearForm()
    TextBox1.Text = ""
    TextBox2.Text = ""
    TextBox3.Text = ""
    TextBox4.Text = ""
    TextBox7.Text = ""
    TextBox8.Text = ""
End Sub

```

```
End Class
```

Employee Registration:

```
Imports System.Data.SqlClient
Imports System.Configuration
```

```
Partial Class _Default
    Inherits System.Web.UI.Page
```

```

'Connection string
Dim connStr As String = "Server=ABDULLAH\SQLEXPRESS;Database=PVFC_DB;User
Id=FarmAdmin;Password=abcd8552;TrustServerCertificate=True;"

```

```

'Submit button
Protected Sub empRegSub_Click(sender As Object, e As EventArgs) Handles empRegSub.Click

```

```

'Insert employee query
Dim empQuery As String = "INSERT INTO EMPLOYEE_† (Employee_Id, Employee_Name,
Employee_Address, Employee_BirthDate, Employee_City, Employee_Date_Hired,
Employee_State, Employee_Supervisor, Employee_Zip) " &

```

```
        "VALUES (@Id, @Name, @Address, @BD, @City, @Hire, @State, @Sup,  
@Zip)"
```

```
'Insert user login query
```

```
Dim userQuery As String = "INSERT INTO USER_t (Username, PasswordHash, RoleID,  
EmployeeID) " &
```

```
        "VALUES (@Username, @Password, 2, @Id)"
```

```
Using conn As New SqlConnection(connStr)
```

```
    conn.Open()
```

```
'Start transaction
```

```
Dim transaction As SqlTransaction = conn.BeginTransaction()
```

```
Try
```

```
'Insert into EMPLOYEE_t
```

```
Using cmdEmp As New SqlCommand(empQuery, conn, transaction)
```

```
    cmdEmp.Parameters.AddWithValue("@Id", txtEmployeeId.Text)  
    cmdEmp.Parameters.AddWithValue("@Name", txtEmployeeName.Text)  
    cmdEmp.Parameters.AddWithValue("@Address", txtEmployeeAddress.Text)  
    cmdEmp.Parameters.AddWithValue("@BD", txtEmployeeBirthDate.Text)  
    cmdEmp.Parameters.AddWithValue("@City", txtEmployeeCity.Text)  
    cmdEmp.Parameters.AddWithValue("@Hire", txtEmployeeDateHired.Text)  
    cmdEmp.Parameters.AddWithValue("@State", txtEmployeeState.Text)  
    cmdEmp.Parameters.AddWithValue("@Sup", txtEmployeeSupervisor.Text)  
    cmdEmp.Parameters.AddWithValue("@Zip", txtEmployeeZip.Text)
```

```
    cmdEmp.ExecuteNonQuery()
```

```
End Using
```

```
'Insert into USER_t
```

```
Using cmdUser As New SqlCommand(userQuery, conn, transaction)
```

```
    cmdUser.Parameters.AddWithValue("@Username", TextBox5.Text)  
    cmdUser.Parameters.AddWithValue("@Password", TextBox6.Text)  
    cmdUser.Parameters.AddWithValue("@Id", txtEmployeeId.Text)
```

```
    cmdUser.ExecuteNonQuery()
```

```
End Using
```

```
'Commit if both succeed
```

```
transaction.Commit()
```

```
Response.Write("<script>alert('Employee " & txtEmployeeId.Text & " Registered  
Successfully!');</script>")
```

```
ClearForm()
```

Catch ex As Exception

'Rollback if error occurs
transaction.Rollback()

Response.Write("Error: " & ex.Message)

End Try

End Using

End Sub

'Clear form fields

Private Sub ClearForm()

txtEmployeeName.Text = ""
txtEmployeeCity.Text = ""
txtEmployeeState.Text = ""
txtEmployeeZip.Text = ""
txtEmployeeAddress.Text = ""
txtEmployeeId.Text = ""

TextBox5.Text = ""

TextBox6.Text = ""

End Sub

End Class

Payment:

Imports System.Data.SqlClient

Imports System.Configuration

Partial Class PaymentPage

Inherits System.Web.UI.Page

' Keeping your connection string as requested

Dim connStr As String = "Server=ABDULLAH\SQLEXPRESS;Database=PVFC_DB;User
Id=FarmAdmin;Password=abcd8552;TrustServerCertificate=True;"

Protected Sub cardRad_CheckedChanged(sender As Object, e As EventArgs) Handles
cardRad.CheckedChanged, cashRad.CheckedChanged

pnlCardDetails.Visible = cardRad.Checked

End Sub

Protected Sub btnConfirm_Click(sender As Object, e As EventArgs) Handles btnConfirm.Click

' 1. Retrieve data from Session

Dim custID As String = ""

Dim prodID As String = ""

Dim qty As Integer = 0

If Session("CustomerID") IsNot Nothing Then

 custID = Session("CustomerID").ToString()

End If

If Session("SelectedProduct") IsNot Nothing Then

 Dim prod As Product = CType(Session("SelectedProduct"), Product)

 prodID = prod.Product_Id ' make sure property name matches your class

End If

If Session("OrderQuantity") IsNot Nothing Then

 qty = Convert.ToInt32(Session("OrderQuantity"))

End If

'//////////

' 2. VALIDATION (very important)

If custID = "" Or prodID = "" Or qty <= 0 Then

 Response.Write("Same problem bro")

 ' Response.Write("<script>alert('Session expired or Invalid Data. Please place order again.');

 Return

End If

Dim orderDate As DateTime = DateTime.Now

Using conn As New SqlConnection(connStr)

 conn.Open()

 ' Start Transaction

 Dim trans As SqlTransaction = conn.BeginTransaction()

 Try

 ' 3. Generate next Order_Id

 Dim nextOID As Integer

 Dim idSql As String = "SELECT ISNULL(MAX(Order_Id), 1010) + 1 FROM ORDER_t"

 Using idCmd As New SqlCommand(idSql, conn, trans)

 nextOID = Convert.ToInt32(idCmd.ExecuteScalar())

 End Using

 ' 4. Insert into ORDER_t

```

    Dim orderSql As String = "INSERT INTO ORDER_† (Order_Id, Customer_Id,
Order_Date) VALUES (@OID, @CID, @Date)"

    Using oCmd As New SqlCommand(orderSql, conn, trans)
        oCmd.Parameters.AddWithValue("@OID", nextOID)
        oCmd.Parameters.AddWithValue("@CID", custID)
        oCmd.Parameters.AddWithValue("@Date", orderDate)
        oCmd.ExecuteNonQuery()
    End Using

' 5. Insert into ORDER_LINE_†
    Dim lineSql As String = "INSERT INTO ORDER_LINE_† (Order_Id, Product_Id,
Ordered_Quantity) VALUES (@OID, @PID, @Qty)"

    Using lCmd As New SqlCommand(lineSql, conn, trans)
        lCmd.Parameters.AddWithValue("@OID", nextOID)
        lCmd.Parameters.AddWithValue("@PID", prodID)
        lCmd.Parameters.AddWithValue("@Qty", qty)
        lCmd.ExecuteNonQuery()
    End Using

' 6. Commit transaction
    trans.Commit()

' OPTIONAL: clear session to avoid duplicate orders
    Session.Remove("SelectedProduct")
    Session.Remove("OrderQuantity")

    Response.Write("<script>alert('Order Successful! Your Order ID is: " & nextOID & "');
window.location='home.aspx';</script>")

    Catch ex As Exception
        trans.Rollback()
        Response.Write("<script>alert('Database Error: " & ex.Message.Replace("'", "'") &
"'");</script>")
    End Try
End Using
End Sub
End Class

```