



**PIEAS**

Pakistan Institute of Engineering and Applied Sciences

## **Report:**

### **"Online Grocery Store Management System – A Web-Based Application"**

**Name:** Wajeeha Jahangir  
**Department:** Computer Science  
**Session:** 22-26  
**Date:** 16 December, 2024  
**Course:** IAD  
**Submitted To:** Sir Irfan Hameed

# Online Grocery Store Management System – A Web-Based Application

## Introduction:

The Online Grocery Store Management System is a web-based application that allows customers to shop for groceries online, while enabling suppliers and administrators to manage products, orders, and users efficiently. It simplifies grocery shopping by offering a convenient and automated platform accessible through any browser.

## Codes:

### Role selection:

```
<%@ Page Language="VB" AutoEventWireup="false" CodeFile="RoleSelect.aspx.vb"
Inherits="RoleSelect" %>

<!DOCTYPE html>
<html>
<head runat="server">
    <title>Role Selection</title>
    <style>
        body {
            background-image:
url('https://staticvecteezy.com/system/resources/thumbnails/045/661/918/small_2x/empty-wooden-table-with-a-beautiful-grocery-store-background-featuring-aisles-of-colorful-products-ideal-for-food-marketing-and-promotions-photo.jpeg');
            background-size: cover;
            background-position: center;
            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
            margin: 0;
            padding: 0;
            display: flex;
            justify-content: center;
            align-items: center;
            min-height: 100vh;
        }

        #form1 {
            width: 100%;
            max-width: 480px; /* Optimal width */
            padding: 15px;
        }

        .container {
            text-align: center;
            padding: 25px 35px; /* Reduced vertical padding */
            background-color: rgba(255, 255, 255, 0.97);
            border-radius: 12px;
            box-shadow: 0 8px 25px rgba(0, 0, 0, 0.15);
            border: 1px solid rgba(255, 255, 255, 0.2);
            backdrop-filter: blur(2px);
        }

        .welcome-title {
```

```
        color: #2c3e50;
        margin: 0 0 5px 0;
        font-size: 22px;
        font-weight: 600;
    }

    .role-title {
        color: #5d5d5d;
        margin: 0 0 18px 0; /* Reduced margin */
        font-size: 15px;
        font-weight: 500;
    }

    .button {
        width: 100%;
        height: 42px; /* Slightly reduced height */
        font-size: 14px;
        margin: 8px 0; /* Reduced margin */
        border-radius: 8px;
        border: none;
        cursor: pointer;
        font-weight: 600;
        color: white;
        letter-spacing: 0.5px;
    }

    .button:hover {
        box-shadow: 0 5px 12px rgba(0, 0, 0, 0.15);
    }

    .btnUser {
        background: linear-gradient(135deg, #3a7ca5, #2c6082);
    }

    .btnSupplier {
        background: linear-gradient(135deg, #5d5d5d, #3d3d3d);
    }

    .btnAdmin {
        background: linear-gradient(135deg, #2e7d32, #1b5e20);
    }

    .logo {
        width: 50px;
        height: 50px;
        margin: 0 auto 15px auto; /* Reduced margin */
        background-color: rgba(248, 249, 250, 0.8);
        border-radius: 50%;
        display: flex;
        align-items: center;
        justify-content: center;
        box-shadow: 0 3px 10px rgba(0, 0, 0, 0.1);
    }

    .logo i {
        font-size: 24px;
        color: #2e7d32;
    }

```

</style>

<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-beta3/css/all.min.css">

```

</head>
<body>
    <form id="form1" runat="server">
        <div class="container">
            <div class="logo">
                <i class="fas fa-shopping-basket"></i>
            </div>
            <h2 class="welcome-title">Welcome to Grocery Store</h2>
            <p class="role-title">Select your role to continue</p>

            <asp:Button ID="btnUser" runat="server" Text="Customer"
            CssClass="button btnUser" OnClick="btnUser_Click" />
            <asp:Button ID="btnSupplier" runat="server" Text="Supplier"
            CssClass="button btnSupplier" OnClick="btnSupplier_Click" />
            <asp:Button ID="btnAdmin" runat="server" Text="Admin" CssClass="button
            btnAdmin" OnClick="btnAdmin_Click" />
        </div>
    </form>
</body>
</html>

```

### **VB code:**

```

Partial Class RoleSelect
    Inherits System.Web.UI.Page

    Protected Sub btnUser_Click(sender As Object, e As EventArgs) Handles
    btnUser.Click
        Response.Redirect("User_Login.aspx")
    End Sub

    Protected Sub btnSupplier_Click(sender As Object, e As EventArgs) Handles
    btnSupplier.Click
        Response.Redirect("Supplier_Login.aspx")
    End Sub

    Protected Sub btnAdmin_Click(sender As Object, e As EventArgs) Handles
    btnAdmin.Click
        Response.Redirect("Admin_Login.aspx")
    End Sub
End Class

```

### **Registration:**

```

<%@ Page Language="VB" AutoEventWireup="false" CodeFile="Registration.aspx.vb"
Inherits="Registration" %>

<!DOCTYPE html>
<html>
<head runat="server">
    <title>Registration</title>
    <link
    href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;500&display=swap"
    rel="stylesheet">
    <style>

```

```
body {
    font-family: 'Poppins', sans-serif;
    background:
url('https://static.vecteezy.com/system/resources/thumbnails/045/661/918/small_2x/e
mpty-wooden-table-with-a-beautiful-grocery-store-background-featuring-aisles-of-
colorful-products-ideal-for-food-marketing-and-promotions-photo.jpeg') center/cover
fixed;
    display: flex;
    justify-content: center;
    align-items: center;
    min-height: 100vh;
    margin: 0;
}
.reg-box {
    background: rgba(255, 255, 255, 0.98);
    padding: 20px;
    border-radius: 8px;
    box-shadow: 0 4px 15px rgba(0, 0, 0, 0.08);
    width: 100%;
    max-width: 380px;
}
h2 {
    text-align: center;
    margin: 0 0 12px;
    color: #00796b;
    font-size: 20px;
    font-weight: 600;
}
label {
    display: block;
    margin: 6px 0 3px;
    font-weight: 400;
    font-size: 11px;
    color: #555;
    letter-spacing: 0.2px;
}
input {
    width: 100%;
    padding: 8px 10px;
    margin-bottom: 6px;
    border: 1px solid #e0e0e0;
    border-radius: 4px;
    font-size: 13px;
    transition: border 0.2s;
    height: 36px;
    box-sizing: border-box;
}
input:focus {
    border-color: #00796b;
    outline: none;
}
.btn {
    width: 100%;
    padding: 9px;
    background: #00796b;
    color: white;
    border: none;
    border-radius: 4px;
    font-weight: 500;
    font-size: 13px;
    margin-top: 4px;
}
```

```

        cursor: pointer;
        transition: background 0.2s;
        height: 36px;
    }
    .btn:hover {
        background: #00695c;
    }
    .message {
        color: #e53935;
        text-align: center;
        margin-top: 8px;
        font-size: 11px;
        min-height: 16px;
    }
    .role {
        text-align: center;
        color: #00796b;
        font-weight: 500;
        margin-bottom: 12px;
        font-size: 12px;
    }

```

</style>

</head>

<body>

<form id="form1" runat="server">

<div class="reg-box">

<h2>Registration</h2>

<asp:Label ID="lblRole" runat="server" CssClass="role"></asp:Label>

<br />

<!-- First Name -->

<asp:Label runat="server" Text="First Name"></asp:Label>

<asp:TextBox ID="FirstName" runat="server"></asp:TextBox>

<asp:RequiredFieldValidator ID="rfvFirstName" runat="server" ControlToValidate="FirstName" ErrorMessage="First name is required." ForeColor="Red" Display="Dynamic" />

<asp:RegularExpressionValidator ID="revFirstName" runat="server" ControlToValidate="FirstName" ValidationExpression="^[A-Za-z]{3,}+\$" ErrorMessage="Only alphabets, min 3 chars." ForeColor="Red" Display="Dynamic" />

<!-- Last Name -->

<asp:Label runat="server" Text="Last Name"></asp:Label>

<asp:TextBox ID="lastName" runat="server"></asp:TextBox>

<asp:RequiredFieldValidator ID="rfvLastName" runat="server" ControlToValidate="lastName" ErrorMessage="Last name is required." ForeColor="Red" Display="Dynamic" />

<asp:RegularExpressionValidator ID="revLastName" runat="server" ControlToValidate="lastName" ValidationExpression="^[A-Za-z]{3,}+\$" ErrorMessage="Only alphabets, min 3 chars." ForeColor="Red" Display="Dynamic" />

<!-- Username -->

```

<asp:Label runat="server" Text="Username"></asp:Label>
<asp:TextBox ID="txtUsername" runat="server"></asp:TextBox>
<asp:RequiredFieldValidator ID="rfvUsername" runat="server"
    ControlToValidate="txtUsername"
    ErrorMessage="Username is required."
    ForeColor="Red"
    Display="Dynamic" />
<asp:RegularExpressionValidator ID="revUsername" runat="server"
    ControlToValidate="txtUsername"
    ValidationExpression="^\w{4,}+$"
    ErrorMessage="Min 4 characters, letters or numbers only."
    ForeColor="Red"
    Display="Dynamic" />

<!-- Password -->
<asp:Label runat="server" Text="Password"></asp:Label>
<asp:TextBox ID="txtPassword" runat="server"
    TextMode="Password"></asp:TextBox>
<asp:RequiredFieldValidator ID="rfvPassword" runat="server"
    ControlToValidate="txtPassword"
    ErrorMessage="Password is required."
    ForeColor="Red"
    Display="Dynamic" />
<asp:RegularExpressionValidator ID="revPassword" runat="server"
    ControlToValidate="txtPassword"
    ValidationExpression="^.{6,}+$"
    ErrorMessage="Password must be at least 6 characters."
    ForeColor="Red"
    Display="Dynamic" />

<!-- Address -->
<asp:Label runat="server" Text="Address"></asp:Label>
<asp:TextBox ID="txtAddress" runat="server"></asp:TextBox>
<asp:RequiredFieldValidator ID="rfvAddress" runat="server"
    ControlToValidate="txtAddress"
    ErrorMessage="Address is required."
    ForeColor="Red"
    Display="Dynamic" />
<asp:RegularExpressionValidator ID="revAddress" runat="server"
    ControlToValidate="txtAddress"
    ValidationExpression="^.{5,}+$"
    ErrorMessage="Address must be at least 5 characters."
    ForeColor="Red"
    Display="Dynamic" />

<!-- Register Button -->
<asp:Button ID="btnRegister" runat="server" Text="Register"
    CssClass="btn"
    OnClick="btnRegister_Click"
    CausesValidation="true" />

<asp:Label ID="lblMessage" runat="server"
    CssClass="message"></asp:Label>

<asp:ValidationSummary ID="ValidationSummary1" runat="server"
    ForeColor="Red"
    ShowMessageBox=False"
    ShowSummary="True" />
</div>
</form>
</body>

```

```
</html>
```

**VB code:**

```
Imports System.Data.SqlClient
Imports System.Security.Cryptography
Imports System.Text

Partial Class Registration
    Inherits System.Web.UI.Page

    Protected Sub Page_Load(sender As Object, e As EventArgs) Handles Me.Load
        If Not IsPostBack Then
            Dim role As String = Request.QueryString("role")
            If role Is Nothing Then
                lblMessage.Text = "Error: No role provided in query string."
                btnRegister.Enabled = False
            Else
                lblRole.Text = "Registering as: " & role
                ViewState("Role") = role ' Store role in ViewState for use on
postback
            End If
        End If
    End Sub

    Protected Sub btnRegister_Click(sender As Object, e As EventArgs)
        Dim role As String = ViewState("Role").ToString()
        Dim Fname As String = FirstName.Text.Trim()
        Dim Lname As String = lastName.Text.Trim()
        Dim username As String = txtUsername.Text.Trim()
        Dim password As String = txtPassword.Text.Trim()
        Dim address As String = txtAddress.Text.Trim()
        Dim customerId As Integer = 0

        ' Generate a salt for password hashing
        Dim salt As String = GenerateSalt()
        Dim hashedPassword As String = HashPassword(password, salt)

        Dim conStr As String = "Data Source=DESKTOP-1T8MKP4;Initial
Catalog=Grocery_store;Integrated Security=True"
        Using con As New SqlConnection(conStr)
            con.Open()
            Dim query As String = ""

            If role = "Customer" Then
                Dim getMaxIdQuery As String = "SELECT ISNULL(MAX(Customer_ID), 0)
FROM Customer"
                Using cmdMaxId As New SqlCommand(getMaxIdQuery, con)
                    customerId = Convert.ToInt32(cmdMaxId.ExecuteScalar()) + 1
                End Using
                query = "INSERT INTO Customer (Customer_ID, First_Name, Last_Name,
Cust_UserName, Cust_password, Cust_address, Salt) VALUES (@Cust_ID, @FName, @LName,
@Username, @Password, @Address, @Salt)"
            Else
                lblMessage.Text = "Invalid role."
                Exit Sub
            End If

            Using cmd As New SqlCommand(query, con)
                cmd.Parameters.AddWithValue("@Cust_ID", customerId)
                cmd.Parameters.AddWithValue("@FName", Fname)
                cmd.Parameters.AddWithValue("@LName", Lname)
            End Using
        End Using
    End Sub

```

```

        cmd.Parameters.AddWithValue("@Username", username)
        cmd.Parameters.AddWithValue("@Password", hashedPassword)
        cmd.Parameters.AddWithValue("@Address", address)
        cmd.Parameters.AddWithValue("@Salt", salt)

    Try
        cmd.ExecuteNonQuery()
        lblMessage.ForeColor = Drawing.Color.Green
        lblMessage.Text = "Registration successful! You can now log
in."
    Catch ex As Exception
        lblMessage.Text = "Error: " & ex.Message
    End Try
    End Using
End Using
End Sub

' Method to generate a salt using RNGCryptoServiceProvider
Private Function GenerateSalt() As String
    Dim saltBytes(31) As Byte ' 256 bits of salt
    Using rng As New RNGCryptoServiceProvider()
        rng.GetBytes(saltBytes)
    End Using
    Return Convert.ToString(saltBytes)
End Function

' Method to hash password with salt using SHA-256
Private Function HashPassword(password As String, salt As String) As String
    Using sha256 As SHA256 = SHA256.Create()
        Dim saltedPassword As String = password & salt
        Dim hashBytes As Byte() =
sha256.ComputeHash(Encoding.UTF8.GetBytes(saltedPassword))
        Return Convert.ToString(hashBytes)
    End Using
End Function
End Class

Admin_Login: <%@ Page Language="VB" AutoEventWireup="false"
CodeFile="Admin_Login.aspx.vb" Inherits="Admin_Login" %>

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Admin Login</title>
    <link
    href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;500;600&display=swa
p" rel="stylesheet" />
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css" />
    <style>
        body {
            font-family: 'Poppins', sans-serif;
            display: flex;
            justify-content: center;
            align-items: center;
            min-height: 100vh;
            margin: 0;
            background-image:
url('https://staticvecteezy.com/system/resources/thumbnails/045/661/918/small_2x/e
mpty-wooden-table-with-a-beautiful-grocery-store-background-featuring-aisles-of-
colorful-products-ideal-for-food-marketing-and-promotions-photo.jpeg');

```

```
        background-size: cover;
        background-position: center;
        background-attachment: fixed;
    }

.login-box {
    background-color: rgba(255, 255, 255, 0.95);
    padding: 25px 30px;
    border-radius: 10px;
    box-shadow: 0 8px 25px rgba(0, 0, 0, 0.12);
    width: 150%;
    max-width: 600px;
    text-align: center;
    backdrop-filter: blur(3px);
    border: 1px solid rgba(255, 255, 255, 0.3);
}

.login-box:hover {
    box-shadow: 0 12px 30px rgba(0, 0, 0, 0.15);
}

.login-box h2 {
    color: #2c3e50;
    font-size: 24px;
    font-weight: 600;
    margin-bottom: 20px;
    position: relative;
    display: inline-block;
}

.login-box h2:after {
    content: "";
    position: absolute;
    width: 40px;
    height: 3px;
    background: linear-gradient(90deg, #00796b, #004d40);
    bottom: -8px;
    left: 50%;
    border-radius: 3px;
}

.input-group {
    margin-bottom: 16px;
    text-align: left;
}

label {
    display: block;
    margin-bottom: 6px;
    font-weight: 500;
    color: #00796b;
    font-size: 14px;
}

.input-field {
    width: 100%;
    padding: 12px 14px;
    border: 2px solid #e0e0e0;
    border-radius: 6px;
}
```

```
        box-sizing: border-box;
        font-size: 14px;
        transition: all 0.3s;
        background-color: #f8f9fa;
    }

    .input-field:focus {
        border-color: #00796b;
        background-color: #fff;
        outline: none;
        box-shadow: 0 0 0 3px rgba(0, 121, 107, 0.1);
    }

    .password-container {
        position: relative;
    }

.login-btn {
    width: 100%;
    padding: 12px;
    background: linear-gradient(135deg, #00796b, #004d40);
    border: none;
    color: white;
    font-weight: 600;
    font-size: 15px;
    cursor: pointer;
    border-radius: 6px;
    transition: all 0.3s;
    margin-top: 8px;
    letter-spacing: 0.5px;
}

.login-btn:hover {
    background: linear-gradient(135deg, #00695c, #003d33);
    box-shadow: 0 4px 12px rgba(0, 77, 64, 0.25);
}

.message {
    text-align: center;
    color: #e53935;
    margin: 12px 0;
    font-size: 13px;
    min-height: 18px;
}

</style>
</head>
<body>
    <form id="form1" runat="server">
        <div class="login-box">
            <h2>Admin Login</h2>

            <div class="input-group">
                <label for="txtAdminName">Username</label>
                <asp:TextBox ID="txtAdminName" runat="server" CssClass="input-field" placeholder="Enter admin username" />
            </div>

            <div class="input-group password-container">
                <label for="txtPassword">Password</label>
```

```

        <asp:TextBox ID="txtPassword" runat="server" TextMode="Password"
CssClass="input-field" placeholder="Enter your password" />

    </div>

    <asp:Button ID="btnLogin" runat="server" Text="Login"
OnClick="btnLogin_Click" CssClass="login-btn" />

    <asp:Label ID="lblMessage" runat="server"
CssClass="message"></asp:Label>
</div>
</form>

</body>
</html>

```

### VB code:

```

Imports System.Data
Imports System.Data.SqlClient

Partial Class Admin_Login
    Inherits System.Web.UI.Page

    Public connectionString As String = "Data Source=DESKTOP-1T8MKP4;Initial Catalog=Grocery_store;Integrated Security=True"

    Protected Sub btnLogin_Click(sender As Object, e As EventArgs)
        ' Retrieve username and password from form
        Dim username As String = txtAdminName.Text
        Dim password As String = txtPassword.Text

        Using conn As New SqlConnection(connectionString)
            Dim cmd As New SqlCommand("SELECT Admin_ID FROM Admin WHERE Admin_Name = @username AND Admin_Password = @password", conn)
            cmd.Parameters.AddWithValue("@username", username)
            cmd.Parameters.AddWithValue("@password", password)

            conn.Open()
            Dim result = cmd.ExecuteScalar()

            If result IsNot Nothing Then
                ' Successful login
                Session("Admin_ID") = result.ToString()
                Response.Redirect("Admin_Dash.aspx") ' Redirect to dashboard
            Else
                lblMessage.Text = "Invalid login credentials!"
            End If
        End Using
    End Sub
End Class

```

### Admin\_Dashboard:

```
<%@ Page Language="VB" AutoEventWireup="false" CodeFile="Admin_Dash.aspx.vb"
Inherits="Admin_Dash" %>

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Admin Dashboard</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f9;
            margin: 0;
            padding: 0;
        }
        #form1 {
            max-width: 1200px;
            margin: 0 auto;
            padding: 20px;
        }
        .header {
            background-color: #2c3e50;
            color: white;
            padding: 20px;
            text-align: center;
            border-radius: 10px 10px 0 0;
        }
        .header h1 {
            margin: 0;
        }
        .main-btn {
            background-color: #3498db;
            color: white;
            padding: 10px 20px;
            border: none;
            border-radius: 5px;
            margin: 5px;
            cursor: pointer;
            font-size: 16px;
        }
        .main-btn:hover {
            background-color: #2980b9;
        }
        .panel {
            background-color: white;
            padding: 20px;
            border-radius: 10px;
            box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
            margin-top: 20px;
        }
        .panel h2 {
            color: #2c3e50;
            font-size: 20px;
            margin-bottom: 10px;
        }
        .panel input[type="text"] {
            width: 100%;
            padding: 10px;
            margin: 10px 0;
            border-radius: 5px;
            border: 1px solid #ddd;
        }
    </style>
</head>
<body>
    <div class="header">
        <h1>Admin Dashboard</h1>
    </div>
    <div id="form1">
        <div class="panel">
            <h2>Add New User</h2>
            <form>
                <label>First Name:</label>
                <input type="text" />
                <label>Last Name:</label>
                <input type="text" />
                <label>Email:</label>
                <input type="text" />
                <label>Password:</label>
                <input type="password" />
                <label>Confirm Password:</label>
                <input type="password" />
                <label>Role:</label>
                <select>
                    <option>Administrator</option>
                    <option>User</option>
                </select>
                <br/>
                <input type="button" value="Submit" class="main-btn" />
            </form>
        </div>
    </div>
</body>
</html>
```

```

        }
        .panel button {
            background-color: #27ae60;
            color: white;
            padding: 10px 20px;
            border: none;
            border-radius: 5px;
            cursor: pointer;
            font-size: 16px;
        }
        .panel button:hover {
            background-color: #2ecc71;
        }
        .gridview-container {
            margin-top: 20px;
        }
        .gridview-container table {
            width: 100%;
            border-collapse: collapse;
        }
        .gridview-container th, .gridview-container td {
            padding: 12px;
            text-align: left;
            border-bottom: 1px solid #ddd;
        }
        .gridview-container th {
            background-color: #ecf0f1;
        }
    
```

</style>

```

</head>
<body>
    <form id="form1" runat="server">
        <div class="header">
            <h1>Admin Dashboard</h1>
        </div>

        <!-- Main Control Buttons -->
        <div>
            <asp:Button ID="btnManageCategory" runat="server" Text="Manage Categories" OnClick="btnManageCategory_Click" CssClass="main-btn" />
            <asp:Button ID="btnManageSuppliers" runat="server" Text="Manage Suppliers" OnClick="btnManageSuppliers_Click" CssClass="main-btn" />
            <asp:Button ID="btnManageCustomers" runat="server" Text="Manage Customers" OnClick="btnManageCustomers_Click" CssClass="main-btn" />
            <asp:Button ID="btnManageProducts" runat="server" Text="Manage Products" OnClick="btnManageProducts_Click" CssClass="main-btn" />
            <asp:Button ID="btnTrackOrders" runat="server" Text="Track Orders" OnClick="btnTrackOrders_Click" CssClass="main-btn" />
        </div>

        <hr />

        <!-- Categories Panel -->
        <asp:Panel ID="pnlCategory" runat="server" Visible="False" CssClass="panel">
            <h3>Manage Categories</h3>
            <h2>Add Category</h2>
            <asp:TextBox ID="txtCatName" runat="server" Placeholder="Category Name" CssClass="panel-input"></asp:TextBox>
        </asp:Panel>
    </form>

```

```

        <asp:TextBox ID="txtCatDesc" runat="server" Placeholder="Category Description" CssClass="panel-input"></asp:TextBox>
        <asp:TextBox ID="txtAdminID" runat="server" Placeholder="Admin ID" CssClass="panel-input"></asp:TextBox>
        <asp:Button ID="btnAddCategory" runat="server" Text="Add Category" OnClick="btnAddCategory_Click" CssClass="panel-btn" />
        <h2>View Categories</h2>
        <asp:Button ID="btnViewCategory" runat="server" Text="View Categories" OnClick="btnViewCategory_Click" CssClass="panel-btn" />
        <div class="gridview-container">
            <asp:GridView ID="gvCategories" runat="server" AutoGenerateColumns="True" />
        </div>
        <h2>Delete Category</h2>
        <asp:TextBox ID="txtCatID" runat="server" Placeholder="Category ID" CssClass="panel-input"></asp:TextBox>
        <asp:Button ID="btnDeleteCategory" runat="server" Text="Delete Category" OnClick="btnDeleteCategory_Click" CssClass="panel-btn" />
        <asp:Label ID="lblCategoryMessage" runat="server"></asp:Label>
    </asp:Panel>

    <!-- Suppliers Panel -->
    <asp:Panel ID="pnlSuppliers" runat="server" Visible="False" CssClass="panel">
        <h3>Manage Suppliers</h3>
        <h2>Add Supplier</h2>
        <asp:TextBox ID="txtSuppUser" runat="server" Placeholder="Supplier Username" CssClass="panel-input"></asp:TextBox>
        <asp:TextBox ID="txtSuppName" runat="server" Placeholder="Supplier Name" CssClass="panel-input"></asp:TextBox>
        <asp:TextBox ID="txtSuppAddress" runat="server" Placeholder="Supplier Address" CssClass="panel-input"></asp:TextBox>
        <asp:Button ID="btnAddSupplier" runat="server" Text="Add Supplier" OnClick="btnAddSuppliers_Click" CssClass="panel-btn" />
        <h2>View Suppliers</h2>
        <asp:Button ID="btnViewSuppliers" runat="server" Text="View Suppliers" OnClick="ViewSuppliers" CssClass="panel-btn" />
        <div class="gridview-container">
            <asp:GridView ID="gvSuppliers" runat="server" AutoGenerateColumns="True" />
        </div>
        <h2>Delete Supplier</h2>
        <asp:TextBox ID="txtSuppID" runat="server" Placeholder="Supplier ID" CssClass="panel-input"></asp:TextBox>
        <asp:Button ID="btnDeleteSupplier" runat="server" Text="Delete Supplier" OnClick="btnDeleteSupplier_Click" CssClass="panel-btn" />
    </asp:Panel>

    <!-- Customers Panel -->
    <asp:Panel ID="pnlCustomers" runat="server" Visible="False" CssClass="panel">
        <h3>Manage Customers</h3>
        <h2>View Customers</h2>
        <asp:Button ID="btnViewCustomers" runat="server" Text="View Customers" OnClick="ViewCustomers" CssClass="panel-btn" />
        <div class="gridview-container">
            <asp:GridView ID="gvCustomers" runat="server" AutoGenerateColumns="True" />
        </div>
    </asp:Panel>

```

```

        <!-- Products Panel -->
        <asp:Panel ID="pnlProducts" runat="server" Visible="False"
CssClass="panel">
            <h3>Manage Products</h3>
            <h2>View Products</h2>
            <asp:Button ID="btnViewProducts" runat="server" Text="View Products"
OnClick="ViewProducts" CssClass="panel-btn" />
            <div class="gridview-container">
                <asp:GridView ID="gvProducts" runat="server"
AutoGenerateColumns="True" />
            </div>
        </asp:Panel>

        <!-- Orders Panel -->
        <asp:Panel ID="pnlOrders" runat="server" Visible="False" CssClass="panel">
            <h3>Track Orders</h3>
            <asp:Button ID="btnViewOrders" runat="server" Text="View Orders"
OnClick="ViewOrders" CssClass="panel-btn" />
            <div class="gridview-container">
                <asp:GridView ID="gvOrders" runat="server"
AutoGenerateColumns="True" />
            </div>
        </asp:Panel>

    </form>
</body>
</html>

```

### VB code:

```

Imports System.Data
Imports System.Data.SqlClient

Partial Class Admin_Dash
    Inherits System.Web.UI.Page
    Private connectionString As String = "Data Source=DESKTOP-1T8MKP4;Initial
Catalog=Grocery_store;Integrated Security=True"
    Private admin_id As Integer

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Me.Load
        If Session("Admin_ID") Is Nothing Then
            Response.Redirect("Admin_Login.aspx")
            Return
        End If

        admin_id = Convert.ToInt32(Session("Admin_ID"))

        If Not IsPostBack Then
            ' Optionally load something here on first page load
        End If
    End Sub
    Protected Sub btnManageCategory_Click(sender As Object, e As EventArgs)
        pnlCategory.Visible = True
        pnlSuppliers.Visible = False
    End Sub

```

```

        pnlCustomers.Visible = False
        pnlProducts.Visible = False
        pnlOrders.Visible = False
    End Sub
    Protected Sub btnManageSuppliers_Click(sender As Object, e As EventArgs)
        pnlCategory.Visible = False
        pnlSuppliers.Visible = True
        pnlCustomers.Visible = False
        pnlProducts.Visible = False
        pnlOrders.Visible = False
    End Sub

    Protected Sub btnManageCustomers_Click(sender As Object, e As EventArgs)
        pnlCategory.Visible = False
        pnlSuppliers.Visible = False
        pnlCustomers.Visible = True
        pnlProducts.Visible = False
        pnlOrders.Visible = False
    End Sub

    Protected Sub btnManageProducts_Click(sender As Object, e As EventArgs)
        pnlCategory.Visible = False
        pnlSuppliers.Visible = False
        pnlCustomers.Visible = False
        pnlProducts.Visible = True
        pnlOrders.Visible = False
    End Sub

    Protected Sub btnTrackOrders_Click(sender As Object, e As EventArgs)
        pnlCategory.Visible = False
        pnlSuppliers.Visible = False
        pnlCustomers.Visible = False
        pnlProducts.Visible = False
        pnlOrders.Visible = True
    End Sub

    End Sub

    Protected Sub btnViewCategory_Click(sender As Object, e As EventArgs)
        ViewCategories()
    End Sub

    ' Category Management
    Protected Sub btnAddCategory_Click(sender As Object, e As EventArgs)
        AddCategory(txtCatName.Text, txtCatDesc.Text, txtAdminID.Text)
    End Sub

    Protected Sub AddCategory(name As String, desc As String, adminId As String)
        ' Fix: Retrieve max category_ID correctly
        Dim maxID As Integer = 0
        Using conn As New SqlConnection(connectionString)
            Dim cmd As New SqlCommand("SELECT ISNULL(MAX(Category_ID), 0) FROM Category", conn)
            conn.Open()
            maxID = Convert.ToInt32(cmd.ExecuteScalar())
        End Using
        Dim category_id As Integer = maxID + 1

        Using conn As New SqlConnection(connectionString)
            Dim cmd As New SqlCommand("INSERT INTO Category (Category_ID, Cat_Name, Cat_description, Admin_ID) VALUES (@category_id, @Name, @Description, @AdminID)", conn)

```

```

        cmd.Parameters.AddWithValue("@category_id", category_id)
        cmd.Parameters.AddWithValue("@Name", name)
        cmd.Parameters.AddWithValue("@Description", desc)
        cmd.Parameters.AddWithValue("@AdminID", adminId)
        conn.Open()
        cmd.ExecuteNonQuery()
    End Using
    ViewCategories()
End Sub

' View Categories
Protected Sub ViewCategories()
    Using conn As New SqlConnection(connectionString)
        Dim cmd As New SqlCommand("SELECT * FROM Category", conn)
        Dim da As New SqlDataAdapter(cmd)
        Dim dt As New DataTable()
        da.Fill(dt)
        gvCategories.DataSource = dt
        gvCategories.DataBind()
    End Using
End Sub

' Supplier Management
Protected Sub btnAddSuppliers_Click(sender As Object, e As EventArgs)
    AddSupplier(txtSuppUser.Text, txtSuppName.Text, txtSuppAddress.Text)
End Sub

Protected Sub AddSupplier(username As String, name As String, address As String)
    ' Fix: Retrieve max supplier_ID correctly
    Dim maxID As Integer = 0
    Using conn As New SqlConnection(connectionString)
        Dim cmd As New SqlCommand("SELECT ISNULL(MAX(Supplier_ID), 0) FROM Supplier", conn)
        conn.Open()
        maxID = Convert.ToInt32(cmd.ExecuteScalar())
    End Using
    Dim supplierID As Integer = maxID + 1

    Using conn As New SqlConnection(connectionString)
        Dim cmd As New SqlCommand("INSERT INTO Supplier (Supplier_ID, Supp_Username, Supp_Name, Supp_Address) VALUES (@supplierid, @Username, @Name, @Address)", conn)
        cmd.Parameters.AddWithValue("@supplierid", supplierID)
        cmd.Parameters.AddWithValue("@Username", username)
        cmd.Parameters.AddWithValue("@Name", name)
        cmd.Parameters.AddWithValue("@Address", address)
        conn.Open()
        cmd.ExecuteNonQuery()
    End Using
    ViewSuppliers()
End Sub

' View Suppliers
Protected Sub ViewSuppliers()
    Using conn As New SqlConnection(connectionString)
        Dim cmd As New SqlCommand("SELECT * FROM Supplier", conn)
        Dim da As New SqlDataAdapter(cmd)
        Dim dt As New DataTable()
        da.Fill(dt)
        gvSuppliers.DataSource = dt
    End Using
End Sub

```

```

        gvSuppliers.DataBind()
    End Using
End Sub

' Delete Category
Protected Sub btnDeleteCategory_Click(sender As Object, e As EventArgs)
    Dim categoryId As Integer
    If Integer.TryParse(txtCatID.Text, categoryId) Then
        Dim conn As New SqlConnection("YourConnectionString")
        Dim cmdProducts As New SqlCommand("DELETE FROM Products WHERE
CategoryID = @CategoryID", conn)
        Dim cmdCategory As New SqlCommand("DELETE FROM Categories WHERE
CategoryID = @CategoryID", conn)

        cmdProducts.Parameters.AddWithValue("@CategoryID", categoryId)
        cmdCategory.Parameters.AddWithValue("@CategoryID", categoryId)

        conn.Open()
        cmdProducts.ExecuteNonQuery()
        cmdCategory.ExecuteNonQuery()
        conn.Close()
    End If
End Sub

' Delete Supplier
Protected Sub btnDeleteSupplier_Click(sender As Object, e As EventArgs)
    DeleteSupplier(txtSuppID.Text)
End Sub

Protected Sub DeleteSupplier(supplierId As String)
    Using conn As New SqlConnection(connectionString)
        Dim cmd As New SqlCommand("DELETE FROM Supplier WHERE Supplier_ID =
@ID", conn)
        cmd.Parameters.AddWithValue("@ID", supplierId)
        conn.Open()
        cmd.ExecuteNonQuery()
    End Using
    ViewSuppliers()
End Sub

' Product Management
Protected Sub ViewProducts()
    Using conn As New SqlConnection(connectionString)
        Dim cmd As New SqlCommand("SELECT * FROM Product", conn)
        Dim da As New SqlDataAdapter(cmd)
        Dim dt As New DataTable()
        da.Fill(dt)
        gvProducts.DataSource = dt
        gvProducts.DataBind()
    End Using
End Sub

' Customer Management
Protected Sub ViewCustomers()
    Using conn As New SqlConnection(connectionString)
        Dim cmd As New SqlCommand("SELECT * FROM Customer", conn)
        Dim da As New SqlDataAdapter(cmd)
        Dim dt As New DataTable()
        da.Fill(dt)
        gvCustomers.DataSource = dt
    End Using
End Sub

```

```

        gvCustomers.DataBind()
    End Using
End Sub

' Order Tracking
Protected Sub ViewOrders()
    Using conn As New SqlConnection(connectionString)
        Dim cmd As New SqlCommand("SELECT * FROM Customer_Order", conn)
        Dim da As New SqlDataAdapter(cmd)
        Dim dt As New DataTable()
        da.Fill(dt)
        gvOrders.DataSource = dt
        gvOrders.DataBind()
    End Using
End Sub
End Class

```

## User\_login:

```

<%@ Page Language="VB" AutoEventWireup="false" CodeFile="User_Login.aspx.vb"
Inherits="User_Login" %>

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>User Login</title>
    <link
    href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;500;600&display=swa
p" rel="stylesheet" />
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css" />
    <style>
        body {
            font-family: 'Poppins', sans-serif;
            display: flex;
            justify-content: center;
            align-items: center;
            min-height: 100vh;
            margin: 0;
            background-image:
url('https://staticvecteezy.com/system/resources/thumbnails/045/661/918/small_2x/e
mpty-wooden-table-with-a-beautiful-grocery-store-background-featuring-aisles-of-
colorful-products-ideal-for-food-marketing-and-promotions-photo.jpeg');
            background-size: cover;
            background-position: center;
            background-attachment: fixed;
        }

        .login-box {
            background-color: rgba(255, 255, 255, 0.95);
            padding: 25px 30px;
            border-radius: 10px;
            box-shadow: 0 8px 25px rgba(0, 0, 0, 0.12);
            width: 150px;
            max-width: 600px;
            text-align: center;
            backdrop-filter: blur(3px);
            border: 1px solid rgba(255, 255, 255, 0.3);
            transition: transform 0.3s ease;
        }
    </style>

```

```
.login-box:hover {
    transform: translateY(-3px);
    box-shadow: 0 12px 30px rgba(0, 0, 0, 0.15);
}

.login-box h2 {
    color: #2c3e50;
    font-size: 24px;
    font-weight: 600;
    margin-bottom: 20px;
    position: relative;
    display: inline-block;
}

.login-box h2:after {
    content: '';
    position: absolute;
    width: 40px;
    height: 3px;
    background: linear-gradient(90deg, #00796b, #004d40);
    bottom: -8px;
    left: 50%;
    transform: translateX(-50%);
    border-radius: 3px;
}

.input-group {
    margin-bottom: 16px;
    text-align: left;
}

label {
    display: block;
    margin-bottom: 6px;
    font-weight: 500;
    color: #00796b;
    font-size: 14px;
}

.input-field {
    width: 100%;
    padding: 12px 14px;
    border: 2px solid #e0e0e0;
    border-radius: 6px;
    box-sizing: border-box;
    font-size: 14px;
    transition: all 0.3s;
    background-color: #f8f9fa;
}

.input-field:focus {
    border-color: #00796b;
    background-color: #fff;
    outline: none;
    box-shadow: 0 0 0 3px rgba(0, 121, 107, 0.1);
}

.password-container {
    position: relative;
}
```

```

.login-btn {
    width: 100%;
    padding: 12px;
    background: linear-gradient(135deg, #00796b, #004d40);
    border: none;
    color: white;
    font-weight: 600;
    font-size: 15px;
    cursor: pointer;
    border-radius: 6px;
    transition: all 0.3s;
    margin-top: 8px;
    letter-spacing: 0.5px;
}

.login-btn:hover {
    background: linear-gradient(135deg, #00695c, #003d33);
    transform: translateY(-1px);
    box-shadow: 0 4px 12px rgba(0, 77, 64, 0.25);
}

.message {
    text-align: center;
    color: #e53935;
    margin: 12px 0;
    font-size: 13px;
    min-height: 18px;
}

.register-link {
    margin-top: 16px;
    font-size: 13px;
    color: #616161;
}

.register-link a {
    color: #00796b;
    text-decoration: none;
    font-weight: 500;
    transition: color 0.3s;
}

.register-link a:hover {
    color: #004d40;
    text-decoration: underline;
}

</style>
</head>
<body>
    <form id="form1" runat="server">
        <div class="login-box">
            <h2>User Login</h2>

            <div class="input-group">
                <label for="txtUser">Username</label>
                <asp:TextBox ID="txtUser" runat="server" CssClass="input-field"
placeholder="Enter your username" />

```

```

        </div>

        <div class="input-group password-container">
            <label for="txtPass">Password</label>
            <asp:TextBox ID="txtPass" runat="server" TextMode="Password"
CssClass="input-field" placeholder="Enter your password" />
        </div>

        <asp:Button ID="btnLogin" runat="server" Text="Login"
OnClick="btnLogin_Click" CssClass="login-btn" />

        <asp:Label ID="lblMessage" runat="server"
CssClass="message"></asp:Label>

        <div class="register-link">
            <asp:HyperLink ID="lnkRegister" runat="server"
NavigateUrl="Registration.aspx?role=Customer">
                Not registered? <span>Create an account</span>
            </asp:HyperLink>
        </div>
    </div>
</form>

</body>
</html>

```

### VB code:

```

Imports System.Data.SqlClient
Imports System.Security.Cryptography
Imports System.Text

Partial Class User_Login
    Inherits System.Web.UI.Page

    Protected Sub btnLogin_Click(sender As Object, e As EventArgs)
        Dim connectionString As String = "Data Source=DESKTOP-1T8MKP4;Initial Catalog=Grocery_store;Integrated Security=True"

        Using con As New SqlConnection(connectionString)
            Try
                con.Open()

                Dim cmd As New SqlCommand("SELECT Cust_password, Salt, Customer_ID
FROM Customer WHERE Cust_UserName = @username", con)
                cmd.Parameters.AddWithValue("@username", txtUser.Text.Trim())

                Using reader As SqlDataReader = cmd.ExecuteReader()
                    If reader.Read() Then
                        Dim storedHashedPassword As String =
reader("Cust_password").ToString()
                        Dim storedSalt As String = reader("Salt").ToString()

                        Dim enteredHashedPassword As String =
HashPassword(txtPass.Text.Trim(), storedSalt)

                        If enteredHashedPassword = storedHashedPassword Then
                            Session("UserID") = reader("Customer_ID").ToString()
                            Response.Redirect("User_Dash.aspx")
                        End If
                    End If
                End Using
            End Try
        End Using
    End Sub
End Class

```

```

        Else
            lblMessage.Text = "Invalid username or password."
        End If
    Else
        lblMessage.Text = "Invalid username or password."
    End If
End Using
Catch ex As Exception
    lblMessage.Text = "Error: " & ex.Message
End Try
End Using
End Sub

Private Function HashPassword(password As String, salt As String) As String
    Using sha256 As SHA256 = SHA256.Create()
        Dim saltedPassword As String = password & salt
        Dim hashBytes As Byte() =
    sha256.ComputeHash(Encoding.UTF8.GetBytes(saltedPassword))
        Return Convert.ToString(hashBytes)
    End Using
End Function
End Class

```

## User\_Dashboard:

```

<%@ Page Language="VB" AutoEventWireup="false" CodeFile="User_Dash.aspx.vb"
Inherits="User_Dash" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Grocery Store - User Dashboard</title>
    <style type="text/css">
        body {
            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
            margin: 0;
            padding: 0;
            background-color: #f8f9fa;
        }
        .header {
            background-color: #28a745;
            color: white;
            padding: 15px 0;
            box-shadow: 0 2px 5px rgba(0,0,0,0.1);
        }
        .container {
            width: 90%;
            max-width: 1200px;
            margin: 0 auto;
        }
        .header-content {
            display: flex;
            justify-content: space-between;
            align-items: center;
        }
        .logo {
            font-size: 24px;

```

```
        font-weight: bold;
    }
.user-nav {
    display: flex;
    align-items: center;
    gap: 15px;
}
.btn-nav {
    padding: 8px 15px;
    border: none;
    border-radius: 4px;
    cursor: pointer;
    font-size: 14px;
    transition: all 0.3s;
}
.btn-cart {
    background-color: #ffc107;
    color: #212529;
}
.btn-orders {
    background-color: #17a2b8;
    color: white;
}
.btn-logout {
    background-color: #dc3545;
    color: white;
}
.btn-nav:hover {
    opacity: 0.9;
    transform: translateY(-2px);
}
.user-profile {
    background-color: white;
    padding: 20px;
    border-radius: 5px;
    margin: 20px 0;
    box-shadow: 0 2px 5px rgba(0,0,0,0.1);
}
.profile-info {
    margin-bottom: 10px;
}
.search-container {
    margin: 20px 0;
    display: flex;
    gap: 10px;
}
.search-box {
    flex-grow: 1;
    padding: 10px;
    border: 1px solid #ced4da;
    border-radius: 4px;
    font-size: 16px;
}
.btn-search {
    padding: 10px 20px;
    background-color: #28a745;
    color: white;
    border: none;
    border-radius: 4px;
    cursor: pointer;
    font-size: 16px;
}
```

```
}

.products-container {
  display: grid;
  grid-template-columns: repeat(auto-fill, minmax(250px, 1fr));
  gap: 20px;
  margin: 20px 0;
}
.product-card {
  background-color: white;
  border-radius: 5px;
  overflow: hidden;
  box-shadow: 0 2px 5px rgba(0,0,0,0.1);
  transition: all 0.3s;
}
.product-card:hover {
  transform: translateY(-5px);
  box-shadow: 0 5px 15px rgba(0,0,0,0.1);
}
.product-card img {
  width: 100%;
  height: 180px;
  object-fit: cover;
}
.product-info {
  padding: 15px;
}
.product-info h3 {
  margin: 0 0 10px 0;
  color: #333;
  font-size: 18px;
}
.price {
  font-weight: bold;
  color: #28a745;
  font-size: 16px;
  margin: 5px 0;
}
.stock {
  color: #6c757d;
  font-size: 14px;
  margin: 5px 0;
}
.add-to-cart {
  width: 100%;
  padding: 10px;
  background-color: #28a745;
  color: white;
  border: none;
  cursor: pointer;
  font-size: 16px;
  transition: background-color 0.3s;
}
.add-to-cart:hover {
  background-color: #218838;
}
.out-of-stock {
  width: 100%;
  padding: 10px;
  background-color: #6c757d;
  color: white;
  border: none;
}
```

```

        cursor: not-allowed;
        font-size: 16px;
    }
.no-products, .error-message {
    grid-column: 1 / -1;
    text-align: center;
    padding: 20px;
    background-color: white;
    border-radius: 5px;
    box-shadow: 0 2px 5px rgba(0,0,0,0.1);
}
.error-message {
    color: #dc3545;
}

```

</style>

</head>

<body>

<form id="form1" runat="server">

<div class="header">

<div class="container header-content">

<div class="logo">Grocery Store</div>

<div class="user-nav">

<asp:Button ID="btnViewCart" runat="server" Text="View Cart" CssClass="btn-nav btn-cart" OnClick="btnViewCart\_Click" />

<asp:Button ID="btnViewOrders" runat="server" Text="My Orders" CssClass="btn-nav btn-orders" OnClick="btnViewOrders\_Click" />

<asp:Button ID="btnLogout" runat="server" Text="Logout" CssClass="btn-nav btn-logout" OnClick="btnLogout\_Click" />

</div>

</div>

</div>

<div class="container">

<div class="user-profile">

<h2><asp:Label ID="lblWelcome" runat="server" Text="Welcome"></asp:Label></h2>

<div class="profile-info">

<asp:Label ID="lblUsername" runat="server" Text=""></asp:Label>

</div>

<div class="profile-info">

<asp:Label ID="lblAddress" runat="server" Text=""></asp:Label>

</div>

</div>

<div class="search-container">

<asp:TextBox ID="txtSearch" runat="server" CssClass="search-box" placeholder="Search products..."/></asp:TextBox>

<asp:Button ID="btnSearch" runat="server" Text="Search" CssClass="btn-search" OnClick="btnSearch\_Click" />

</div>

<div id="ProductPanel" runat="server" class="products-container">

<!-- Products will be loaded here dynamically -->

</div>

</div>

</form>

</body>

</html>

**VB code:**

```

Imports System.Data.SqlClient

Partial Class User_Dash
    Inherits System.Web.UI.Page

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Me.Load
        If Session("UserID") Is Nothing Then
            Response.Redirect("User_Login.aspx")
            Exit Sub
        End If

        If Not IsPostBack Then
            LoadProducts("") ' Load all products on first visit
            LoadUserProfile()
        End If
    End Sub

    Protected Sub btnSearch_Click(ByVal sender As Object, ByVal e As EventArgs)
Handles btnSearch.Click
        Dim searchTerm As String = txtSearch.Text.Trim()
        LoadProducts(searchTerm)
    End Sub

    Private Sub LoadProducts(ByVal keyword As String)
        Dim connStr As String = "Data Source=DESKTOP-1T8MKP4;Initial Catalog=Grocery_store;Integrated Security=True"
        Dim html As String = ""

        Using conn As New SqlConnection(connStr)
            Using cmd As New SqlCommand()
                cmd.Connection = conn

                If String.IsNullOrEmpty(keyword) Then
                    cmd.CommandText = "SELECT Product_ID, Prod_Name, Price, Image, Quantity FROM Product WHERE Quantity > 0"
                Else
                    cmd.CommandText = "SELECT Product_ID, Prod_Name, Price, Image, Quantity FROM Product WHERE Prod_Name LIKE @Keyword AND Quantity > 0"
                    cmd.Parameters.AddWithValue("@Keyword", "%" & keyword & "%")
                End If

                Try
                    conn.Open()
                    Using reader As SqlDataReader = cmd.ExecuteReader()
                        If Not reader.HasRows Then
                            html = "<div class='no-products'>No products found matching your search.</div>"
                        Else
                            While reader.Read()
                                Dim proID As Integer =
Convert.ToInt32(reader("Product_ID"))
                                Dim prodName As String =
reader("Prod_Name").ToString()
                                Dim price As Decimal =
Convert.ToDecimal(reader("Price"))
                                Dim imageUrl As String = If(reader("Image") Is DBNull.Value, "", reader("Image").ToString())
                                Dim quantity As Integer =
Convert.ToInt32(reader("Quantity"))

```

```

        html &= "<div class='product-card'>

            ' Product Image
            If String.IsNullOrEmpty(imageUrl) Then
                html &= "<img
src='https://via.placeholder.com/200x120?text=No+Image' alt='"
& prodName & "' />
            Else
                html &= "<img src='"
& imageUrl & "' alt='"
& prodName & "' />
            End If

            ' Product Info
            html &= "<div class='product-info'>
            html &= "<h3>" & prodName & "</h3>
            html &= "<p class='price'>Rs. " &
            price.ToString("N2") & "</p>
            html &= "</div>

            ' Add to Cart Button
            If quantity > 0 Then
                html &= "<button class='add-to-cart'
onclick=""location.href='AddToCart.aspx?ProductID=" & prodID & "''>Add to
Cart</button>
            Else
                html &= "<button class='out-of-stock'
disabled>Out of Stock</button>
            End If

            html &= "</div>
        End While
    End If
    End Using
    Catch ex As Exception
        html = "<div class='error-message'>Error loading products: " &
ex.Message & "</div>"
    End Try
    End Using
End Using

        ProductPanel.InnerHtml = html
End Sub

Private Sub LoadUserProfile()
    Dim userId As Integer = Convert.ToInt32(Session("UserID"))

    Using conn As New SqlConnection("Data Source=DESKTOP-1T8MKP4;Initial
Catalog=Grocery_store;Integrated Security=True")
        Dim cmd As New SqlCommand("SELECT First_Name, Last_Name, Cust_UserName,
Cust_address FROM Customer WHERE Customer_ID = @UserID", conn)
        cmd.Parameters.AddWithValue("@UserID", userId)

        Try
            conn.Open()
            Dim reader As SqlDataReader = cmd.ExecuteReader()

            If reader.Read() Then
                lblWelcome.Text = "Welcome, " & reader("First_Name").ToString()
& " " & reader("Last_Name").ToString()
            End If
        End Try
    End Using
End Sub

```

```

        lblUsername.Text = "Username: " &
reader("Cust_UserName").ToString()
        lblAddress.Text = "Address: " &
reader("Cust_address").ToString()
    End If
    Catch ex As Exception
        ' Handle error if needed
    End Try
End Using
End Sub

Protected Sub btnViewCart_Click(sender As Object, e As EventArgs) Handles
btnViewCart.Click
    Response.Redirect("Cart.aspx")
End Sub

Protected Sub btnViewOrders_Click(sender As Object, e As EventArgs) Handles
btnViewOrders.Click
    Response.Redirect("OrderHistory.aspx")
End Sub

Protected Sub btnLogout_Click(sender As Object, e As EventArgs) Handles
btnLogout.Click
    Session.Clear()
    Response.Redirect("User_Login.aspx")
End Sub
End Class

```

## Supplier\_Login:

```

<%@ Page Language="VB" AutoEventWireup="false" CodeFile="Supplier_Login.aspx.vb"
Inherits="Supplier_Login" %>

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Supplier Login</title>
    <link
    href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;500;600&display=swa
p" rel="stylesheet" />
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css" />
    <style>
        body {
            font-family: 'Poppins', sans-serif;
            display: flex;
            justify-content: center;
            align-items: center;
            min-height: 100vh;
            margin: 0;
            background-image:
url('https://staticvecteezy.com/system/resources/thumbnails/045/661/918/small_2x/e
mpty-wooden-table-with-a-beautiful-grocery-store-background-featuring-aisles-of-
colorful-products-ideal-for-food-marketing-and-promotions-photo.jpeg');
            background-size: cover;
            background-position: center;
            background-attachment: fixed;
        }
        .login-box {

```

```
background-color: rgba(255, 255, 255, 0.95);
padding: 25px 30px;
border-radius: 10px;
box-shadow: 0 8px 25px rgba(0, 0, 0, 0.12);
width: 150%;
max-width: 600px;
text-align: center;
backdrop-filter: blur(3px);
border: 1px solid rgba(255, 255, 255, 0.3);
transition: transform 0.3s ease;
}

.login-box:hover {
    transform: translateY(-3px);
    box-shadow: 0 12px 30px rgba(0, 0, 0, 0.15);
}

.login-box h2 {
    color: #2c3e50;
    font-size: 24px;
    font-weight: 600;
    margin-bottom: 20px;
    position: relative;
    display: inline-block;
}

.login-box h2:after {
    content: '';
    position: absolute;
    width: 40px;
    height: 3px;
    background: linear-gradient(90deg, #00796b, #004d40);
    bottom: -8px;
    left: 50%;
    transform: translateX(-50%);
    border-radius: 3px;
}

.input-group {
    margin-bottom: 16px;
    text-align: left;
}

label {
    display: block;
    margin-bottom: 6px;
    font-weight: 500;
    color: #00796b;
    font-size: 14px;
}

.input-field {
    width: 100%;
    padding: 12px 14px;
    border: 2px solid #e0e0e0;
    border-radius: 6px;
    box-sizing: border-box;
    font-size: 14px;
    transition: all 0.3s;
    background-color: #f8f9fa;
}
```

```

.input-field:focus {
    border-color: #00796b;
    background-color: #fff;
    outline: none;
    box-shadow: 0 0 0 3px rgba(0, 121, 107, 0.1);
}

.password-container {
    position: relative;
}

.login-btn {
    width: 100%;
    padding: 12px;
    background: linear-gradient(135deg, #00796b, #004d40);
    border: none;
    color: white;
    font-weight: 600;
    font-size: 15px;
    cursor: pointer;
    border-radius: 6px;
    transition: all 0.3s;
    margin-top: 8px;
    letter-spacing: 0.5px;
}

.login-btn:hover {
    background: linear-gradient(135deg, #00695c, #003d33);
    transform: translateY(-1px);
    box-shadow: 0 4px 12px rgba(0, 77, 64, 0.25);
}

.message {
    text-align: center;
    color: #e53935;
    margin: 12px 0;
    font-size: 13px;
    min-height: 18px;
}


```

</style>

</head>

<body>

<form id="form1" runat="server">

<div class="login-box">

<h2>Supplier Login</h2>

<div class="input-group">

<label for="Name">Name</label>

<asp:TextBox ID="Name" runat="server" CssClass="input-field" placeholder="Enter supplier username" />

</div>

<div class="input-group">

<label for="txtUsername">Username</label>

```

        <asp:TextBox ID="txtUsername" runat="server" CssClass="input-field"
placeholder="Enter supplier username" />
    </div>

    <div class="input-group password-container">
        <label for="txtPassword">Password</label>
        <asp:TextBox ID="txtPassword" runat="server" TextMode="Password"
CssClass="input-field" placeholder="Enter your password" />
    </div>

    <asp:Button ID="btnLogin" runat="server" Text="Login"
OnClick="btnLogin_Click" CssClass="login-btn" />

    <asp:Label ID="lblMessage" runat="server"
CssClass="message"></asp:Label>
    </div>
</form>

</body>
</html>

```

### VB code:

```

Imports System.Data.SqlClient

Partial Class Supplier_Login
    Inherits System.Web.UI.Page

    Protected Sub btnLogin_Click(sender As Object, e As EventArgs)
        Dim connectionString As String = "Data Source=DESKTOP-1T8MKP4;Initial
Catalog=Grocery_store;Integrated Security=True"

        Using con As New SqlConnection(connectionString)
            Try
                con.Open()
                Dim cmd As New SqlCommand("SELECT Supplier_ID FROM Supplier WHERE
Supp_Username = @username AND Supp_Name = @name", con)
                cmd.Parameters.AddWithValue("@username", txtUsername.Text.Trim())
                cmd.Parameters.AddWithValue("@name", Name.Text.Trim())

                Dim supplierIdObj As Object = cmd.ExecuteScalar()

                If supplierIdObj IsNot Nothing Then
                    Session("SupplierID") = supplierIdObj.ToString()
                    Response.Redirect("Supplier_Dash.aspx")
                Else
                    lblMessage.Text = "Invalid supplier login."
                End If

                Catch ex As Exception
                    lblMessage.Text = "Error: " & ex.Message
                End Try
            End Using
        End Sub
    End Class

```

## Supplier\_Dashboard:

```
<%@ Page Language="VB" AutoEventWireup="false" CodeFile="Supplier_Dash.aspx.vb"
Inherits="Supplier_Dash" %>

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Supplier Dashboard</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f9;
            margin: 0;
            padding: 0;
        }
        #form1 {
            max-width: 1200px;
            margin: 0 auto;
            padding: 20px;
        }
        .header {
            background-color: #2c3e50;
            color: white;
            padding: 20px;
            text-align: center;
            border-radius: 10px 10px 0 0;
        }
        .header h1 {
            margin: 0;
        }
        .main-btn {
            background-color: #3498db;
            color: white;
            padding: 10px 20px;
            border: none;
            border-radius: 5px;
            margin: 5px;
            cursor: pointer;
            font-size: 16px;
            transition: background-color 0.3s ease;
        }
        .main-btn:hover {
            background-color: #2980b9;
        }
        .panel {
            background-color: white;
            padding: 20px;
            border-radius: 10px;
            box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
            margin-top: 20px;
        }
        .panel h2 {
            color: #2c3e50;
            font-size: 20px;
            margin-bottom: 10px;
        }
        .panel input[type="text"], .panel input[type="number"] {
            width: 100%;
            padding: 10px;
        }
    </style>
</head>
<body>
    <form id="form1" runat="server">
        <div class="header">
            <h1>Supplier Dashboard</h1>
        </div>
        <div class="panel">
            <h2>Main Content Area</h2>
            <p>This is the main content area of the dashboard.</p>
            <input type="text" placeholder="Search..." />
            <input type="button" value="Search" class="main-btn" />
            <table border="1">
                <thead>
                    <tr>
                        <th>Category</th>
                        <th>Product Name</th>
                        <th>Quantity</th>
                    </tr>
                </thead>
                <tbody>
                    <tr>
                        <td>Electronics</td>
                        <td>Smartphone</td>
                        <td>1000</td>
                    </tr>
                    <tr>
                        <td>Clothing</td>
                        <td>T-Shirt</td>
                        <td>500</td>
                    </tr>
                    <tr>
                        <td>Home Goods</td>
                        <td>Dishwasher</td>
                        <td>300</td>
                    </tr>
                </tbody>
            </table>
        </div>
    </form>
</body>
</html>
```

```

        margin: 10px 0;
        border-radius: 5px;
        border: 1px solid #ddd;
    }
    .panel button {
        background-color: #27ae60;
        color: white;
        padding: 10px 20px;
        border: none;
        border-radius: 5px;
        cursor: pointer;
        font-size: 16px;
        transition: background-color 0.3s ease;
    }
    .panel button:hover {
        background-color: #2ecc71;
    }
    .gridview-container {
        margin-top: 20px;
        overflow-x: auto;
    }
    .gridview-container table {
        width: 100%;
        border-collapse: collapse;
    }
    .gridview-container th, .gridview-container td {
        padding: 12px;
        text-align: left;
        border-bottom: 1px solid #ddd;
    }
    .gridview-container th {
        background-color: #ecf0f1;
    }
    .message {
        margin: 10px 0;
        padding: 10px;
        border-radius: 5px;
    }
    .success {
        background-color: #d4edda;
        color: #155724;
    }
    .error {
        background-color: #f8d7da;
        color: #721c24;
    }

```

</style>

</head>

<body>

<form id="form1" runat="server">

<div class="header">

<h1>Supplier Dashboard</h1>

<asp:Label ID="lblWelcome" runat="server" Text=""></asp:Label>

</div>

<!-- Main Control Buttons -->

<div>

<asp:Button ID="btnAddProduct" runat="server" Text="Add Product" OnClick="btnAddProduct\_Click" CssClass="main-btn" />

<asp:Button ID="btnDeleteProduct" runat="server" Text="Delete Product" OnClick="btnDeleteProduct\_Click" CssClass="main-btn" />

```

        <asp:Button ID="btnUpdateProduct" runat="server" Text="Update Product"
        OnClick="btnUpdateProduct_Click" CssClass="main-btn" />
        <asp:Button ID="btnViewProducts" runat="server" Text="Your Products"
        OnClick="btnViewProducts_Click" CssClass="main-btn" />
        <asp:Button ID="btnViewOrders" runat="server" Text="Product Orders"
        OnClick="btnViewOrders_Click" CssClass="main-btn" />
    </div>

    <hr />

    <!-- Add Product Panel -->
    <asp:Panel ID="pnlAddProduct" runat="server" Visible="False"
    CssClass="panel">
        <h2>Add New Product</h2>
        <asp:TextBox ID="txtProdName" runat="server" Placeholder="Product Name"
        CssClass="panel-input" /></asp:TextBox>
        <asp:TextBox ID="txtProdDesc" runat="server" Placeholder="Product
        Description" CssClass="panel-input" TextMode="MultiLine" Rows="3" /></asp:TextBox>
        <asp:TextBox ID="txtCategoryID" runat="server" Placeholder="Category
        ID" CssClass="panel-input" /></asp:TextBox>
        <asp:TextBox ID="txtPrice" runat="server" Placeholder="Price"
        CssClass="panel-input" TextMode="Number" step="0.01" /></asp:TextBox>
        <asp:TextBox ID="txtQuantity" runat="server" Placeholder="Quantity"
        CssClass="panel-input" TextMode="Number" /></asp:TextBox>
        <asp:Button ID="btnSubmitProduct" runat="server" Text="Add Product"
        OnClick="btnSubmitProduct_Click" CssClass="panel-btn" />
        <asp:Label ID="lblAddProductMessage" runat="server"
        CssClass="message" /></asp:Label>
        <div class="gridview-container">
            <asp:GridView ID="gvCategories" runat="server"
            AutoGenerateColumns="True" />
        </div>
    </asp:Panel>

    <!-- Delete Product Panel -->
    <asp:Panel ID="pnlDeleteProduct" runat="server" Visible="False"
    CssClass="panel">
        <h2>Delete Product</h2>
        <asp:TextBox ID="txtDeleteProdID" runat="server" Placeholder="Product
        ID to Delete" CssClass="panel-input" /></asp:TextBox>
        <asp:Button ID="btnConfirmDelete" runat="server" Text="Delete Product"
        OnClick="btnConfirmDelete_Click" CssClass="panel-btn" />
        <asp:Label ID="lblDeleteMessage" runat="server"
        CssClass="message" /></asp:Label>
        <h3>Your Products (for reference)</h3>
        <div class="gridview-container">
            <asp:GridView ID="gvSupplierProducts" runat="server"
            AutoGenerateColumns="True" />
        </div>
    </asp:Panel>

    <!-- Update Product Panel -->
    <asp:Panel ID="pnlUpdateProduct" runat="server" Visible="False"
    CssClass="panel">
        <h2>Update Product</h2>
        <asp:TextBox ID="txtUpdateProdID" runat="server" Placeholder="Product
        ID to Update" CssClass="panel-input" /></asp:TextBox>
        <asp:Button ID="btnLoadProduct" runat="server" Text="Load Product
        Details" OnClick="btnLoadProduct_Click" CssClass="panel-btn" />
        <asp:Panel ID="pnlProductDetails" runat="server" Visible="False">

```

```

        <asp:TextBox ID="txtUpdateName" runat="server" Placeholder="Product
Name" CssClass="panel-input"></asp:TextBox>
        <asp:TextBox ID="txtUpdateDesc" runat="server" Placeholder="Product
Description" CssClass="panel-input" TextMode="MultiLine" Rows="3"></asp:TextBox>
        <asp:TextBox ID="txtUpdateCategory" runat="server"
Placeholder="Category" ID" CssClass="panel-input"></asp:TextBox>
        <asp:TextBox ID="txtUpdatePrice" runat="server" Placeholder="Price"
CssClass="panel-input" TextMode="Number" step="0.01"></asp:TextBox>
        <asp:TextBox ID="txtUpdateQuantity" runat="server"
Placeholder="Quantity" CssClass="panel-input" TextMode="Number"></asp:TextBox>
        <asp:Button ID="btnUpdateProductDetails" runat="server"
Text="Update Product" OnClick="btnUpdateProductDetails_Click" CssClass="panel-btn"
/>
    </asp:Panel>
    <asp:Label ID="lblUpdateMessage" runat="server"
CssClass="message"></asp:Label>
    <h3>Your Products</h3>
    <div class="gridview-container">
        <asp:GridView ID="gvProductsForUpdate" runat="server"
AutoGenerateColumns="True" />
    </div>
</asp:Panel>

<!-- View Products Panel -->
<asp:Panel ID="pnlViewProducts" runat="server" Visible="False"
CssClass="panel">
    <h2>Your Products</h2>
    <div class="gridview-container">
        <asp:GridView ID="gvYourProducts" runat="server"
AutoGenerateColumns="True" />
    </div>
</asp:Panel>

<!-- View Orders Panel -->
<asp:Panel ID="pnlViewOrders" runat="server" Visible="False"
CssClass="panel">
    <h2>Orders for Your Products</h2>
    <div class="gridview-container">
        <asp:GridView ID="gvProductOrders" runat="server"
AutoGenerateColumns="True" />
    </div>
</asp:Panel>

</form>
</body>
</html>

```

### VB code:

```

Imports System.Data
Imports System.Data.SqlClient

Partial Class Supplier_Dash
    Inherits System.Web.UI.Page

    Public connectionString As String = "Data Source=DESKTOP-1T8MKP4;Initial
Catalog=Grocery_store;Integrated Security=True"
    Private supplierID As Integer

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

```

```

        If Session("SupplierID") Is Nothing Then
            Response.Redirect("Supplier_Login.aspx")
            Return
        End If

        supplierID = Convert.ToInt32(Session("SupplierID"))

        If Not IsPostBack Then
            ' Optionally load something here on first page load
        End If
    End Sub

    ' Utility to show one panel at a time
    Private Sub ShowOnlyPanel(panelToShow As Panel)
        pnlAddProduct.Visible = False
        pnlDeleteProduct.Visible = False
        pnlUpdateProduct.Visible = False
        pnlViewProducts.Visible = False
        pnlViewOrders.Visible = False

        panelToShow.Visible = True
    End Sub

    ' Event handlers for navigation
    Protected Sub btnAddProduct_Click(sender As Object, e As EventArgs) Handles
btnAddProduct.Click
        ShowOnlyPanel(pnlAddProduct)
        LoadCategories()
    End Sub

    Protected Sub btnDeleteProduct_Click(sender As Object, e As EventArgs) Handles
btnDeleteProduct.Click
        ShowOnlyPanel(pnlDeleteProduct)
        ViewSupplierProducts(gvSupplierProducts)
    End Sub

    Protected Sub btnUpdateProduct_Click(sender As Object, e As EventArgs) Handles
btnUpdateProduct.Click
        ShowOnlyPanel(pnlUpdateProduct)
        ViewSupplierProducts(gvProductsForUpdate)
    End Sub

    Protected Sub btnViewProducts_Click(sender As Object, e As EventArgs) Handles
btnViewProducts.Click
        ShowOnlyPanel(pnlViewProducts)
        ViewSupplierProducts(gvYourProducts)
    End Sub

    Protected Sub btnViewOrders_Click(sender As Object, e As EventArgs) Handles
btnViewOrders.Click
        ShowOnlyPanel(pnlViewOrders)
        ViewProductOrders()
    End Sub

    ' Load Categories (to dropdown or grid)
    Private Sub LoadCategories()
        Using conn As New SqlConnection(connectionString)
            Dim cmd As New SqlCommand("SELECT Category_ID, Cat_Name FROM Category",
conn)
                Dim da As New SqlDataAdapter(cmd)
                Dim dt As New DataTable()

```

```

        da.Fill(dt)
        gvCategories.DataSource = dt
        gvCategories.DataBind()
    End Using
End Sub

' Add Product
Protected Sub btnSubmitProduct_Click(sender As Object, e As EventArgs) Handles btnSubmitProduct.Click
    Try
        Dim productID As Integer

        Using conn As New SqlConnection(connectionString)
            Dim cmd As New SqlCommand("SELECT ISNULL(MAX(Product_ID), 0) + 1
FROM Product", conn)
            conn.Open()
            productID = Convert.ToInt32(cmd.ExecuteScalar())
        End Using

        Using conn As New SqlConnection(connectionString)
            Dim cmd As New SqlCommand("INSERT INTO Product (Product_ID,
Prod_Name, Prod_Description, Category_ID, Supplier_ID, Price, Quantity) " &
                "VALUES (@ProductID, @Name, @Description,
@CategoryID, @SupplierID, @Price, @Quantity)", conn)

            cmd.Parameters.AddWithValue("@ProductID", productID)
            cmd.Parameters.AddWithValue("@Name", txtProdName.Text)
            cmd.Parameters.AddWithValue("@Description", txtProdDesc.Text)
            cmd.Parameters.AddWithValue("@CategoryID",
Convert.ToInt32(txtCategoryID.Text))
            cmd.Parameters.AddWithValue("@SupplierID", supplierID)
            cmd.Parameters.AddWithValue("@Price",
Convert.ToDecimal(txtPrice.Text))
            cmd.Parameters.AddWithValue("@Quantity",
Convert.ToInt32(txtQuantity.Text))

            conn.Open()
            cmd.ExecuteNonQuery()
        End Using

        lblAddProductMessage.Text = "Product added successfully!"
        lblAddProductMessage.CssClass = "message success"

    Catch ex As Exception
        lblAddProductMessage.Text = "Error: " & ex.Message
        lblAddProductMessage.CssClass = "message error"
    End Try
End Sub

' Delete Product
Protected Sub btnConfirmDelete_Click(sender As Object, e As EventArgs) Handles btnConfirmDelete.Click
    Dim productID As Integer
    If Not Integer.TryParse(txtDeleteProdID.Text, productID) Then
        lblDeleteMessage.Text = "Invalid Product ID"
        Return
    End If

    Using conn As New SqlConnection(connectionString)
        conn.Open()

```

```

        Dim checkCmd As New SqlCommand("SELECT COUNT(*) FROM Product WHERE
Product_ID = @ProductID AND Supplier_ID = @SupplierID", conn)
checkCmd.Parameters.AddWithValue("@ProductID", productID)
checkCmd.Parameters.AddWithValue("@SupplierID", supplierID)

        If Convert.ToInt32(checkCmd.ExecuteScalar()) = 0 Then
            lblDeleteMessage.Text = "Product not found or not yours"
            Return
        End If

        Dim delCmd As New SqlCommand("DELETE FROM Product WHERE Product_ID =
@ProductID", conn)
delCmd.Parameters.AddWithValue("@ProductID", productID)
delCmd.ExecuteNonQuery()

        lblDeleteMessage.Text = "Product deleted!"
        ViewSupplierProducts(gvSupplierProducts)
    End Using
End Sub

' Load Product for Update
Protected Sub btnLoadProduct_Click(sender As Object, e As EventArgs) Handles
btnLoadProduct.Click
    Dim productID As Integer
    If Not Integer.TryParse(txtUpdateProdID.Text, productID) Then
        lblUpdateMessage.Text = "Invalid Product ID"
        Return
    End If

    Using conn As New SqlConnection(connectionString)
        Dim cmd As New SqlCommand("SELECT * FROM Product WHERE Product_ID =
@ProductID AND Supplier_ID = @SupplierID", conn)
        cmd.Parameters.AddWithValue("@ProductID", productID)
        cmd.Parameters.AddWithValue("@SupplierID", supplierID)

        conn.Open()
        Dim reader As SqlDataReader = cmd.ExecuteReader()

        If reader.Read() Then
            txtUpdateName.Text = reader("Prod_Name").ToString()
            txtUpdateDesc.Text = reader("Prod_Description").ToString()
            txtUpdateCategory.Text = reader("Category_ID").ToString()
            txtUpdatePrice.Text = reader("Price").ToString()
            txtUpdateQuantity.Text = reader("Quantity").ToString()
            pnlProductDetails.Visible = True
        Else
            lblUpdateMessage.Text = "Product not found"
            pnlProductDetails.Visible = False
        End If
    End Using
End Sub

' Update Product
Protected Sub btnUpdateProductDetails_Click(sender As Object, e As EventArgs)
Handles btnUpdateProductDetails.Click
    Dim productID As Integer
    If Not Integer.TryParse(txtUpdateProdID.Text, productID) Then
        lblUpdateMessage.Text = "Invalid Product ID"
        Return
    End If

```

```

        Using conn As New SqlConnection(connectionString)
            Dim cmd As New SqlCommand("UPDATE Product SET Prod_Name=@Name,
Prod_Description=@Description, Category_ID=@CategoryID, Price=@Price,
Quantity=@Quantity WHERE Product_ID=@ProductID AND Supplier_ID=@SupplierID", conn)

            cmd.Parameters.AddWithValue("@ProductID", productID)
            cmd.Parameters.AddWithValue("@SupplierID", supplierID)
            cmd.Parameters.AddWithValue("@Name", txtUpdateName.Text)
            cmd.Parameters.AddWithValue("@Description", txtUpdateDesc.Text)
            cmd.Parameters.AddWithValue("@CategoryID",
Convert.ToInt32(txtUpdateCategory.Text))
            cmd.Parameters.AddWithValue("@Price",
Convert.ToDecimal(txtUpdatePrice.Text))
            cmd.Parameters.AddWithValue("@Quantity",
Convert.ToInt32(txtUpdateQuantity.Text))

            conn.Open()
            Dim rowsAffected As Integer = cmd.ExecuteNonQuery()

            If rowsAffected > 0 Then
                lblUpdateMessage.Text = "Product updated!"
                pnlProductDetails.Visible = False
                ViewSupplierProducts(gvProductsForUpdate)
            Else
                lblUpdateMessage.Text = "Update failed!"
            End If
        End Using
    End Sub

    ' View Products
    Private Sub ViewSupplierProducts(gridView As GridView)
        Try
            Using conn As New SqlConnection(connectionString)
                Dim cmd As New SqlCommand("SELECT p.Product_ID, p.Prod_Name,
p.Prod_Description, c.Cat_Name AS Category, p.Price, p.Quantity FROM Product p JOIN
Category c ON p.Category_ID = c.Category_ID WHERE p.Supplier_ID = @SupplierID",
conn)
                    cmd.Parameters.AddWithValue("@SupplierID", supplierID)

                    Dim da As New SqlDataAdapter(cmd)
                    Dim dt As New DataTable()
                    da.Fill(dt)

                    gridView.DataSource = dt
                    gridView.DataBind()
                End Using
            Catch ex As Exception
                ' Handle or log error if needed
            End Try
        End Sub

        ' View Orders (Dummy)
        Private Sub ViewProductOrders()
            ' Add logic here to fetch and bind order data
        End Sub
    End Class

```

**Cart:** <%@ Page Language="VB" AutoEventWireup="false" CodeFile="Cart.aspx.vb"  
Inherits="Cart" %>

```
<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Shopping Cart</title>
    <style type="text/css">
        body {
            font-family: Arial, sans-serif;
            margin: 0;
            padding: 0;
            background-color: #f5f5f5;
        }
        .container {
            width: 80%;
            margin: 20px auto;
            background-color: white;
            padding: 20px;
            border-radius: 5px;
            box-shadow: 0 0 10px rgba(0,0,0,0.1);
        }
        .cart-header {
            font-size: 24px;
            color: #333;
            margin-bottom: 20px;
            padding-bottom: 10px;
            border-bottom: 1px solid #eee;
        }
        .gridview {
            width: 100%;
            border-collapse: collapse;
            margin-bottom: 20px;
        }
        .gridview th {
            background-color: #4CAF50;
            color: white;
            padding: 10px;
            text-align: left;
        }
        .gridview td {
            padding: 10px;
            border-bottom: 1px solid #ddd;
        }
        .gridview tr:nth-child(even) {
            background-color: #f9f9f9;
        }
        .total-section {
            text-align: right;
            font-size: 18px;
            font-weight: bold;
            margin: 20px 0;
        }
        .btn {
            padding: 10px 20px;
            border: none;
            border-radius: 3px;
            cursor: pointer;
            font-size: 16px;
        }
    </style>
</head>
<body>
    <div class="container">
        <h1>Cart</h1>
        <table border="1" class="gridview">
            <thead>
                <tr>
                    <th>Product Name</th>
                    <th>Quantity</th>
                    <th>Price</th>
                </tr>
            </thead>
            <tbody>
                <tr>
                    <td>Product A</td>
                    <td>2</td>
                    <td>$10.00</td>
                </tr>
                <tr>
                    <td>Product B</td>
                    <td>1</td>
                    <td>$5.00</td>
                </tr>
                <tr>
                    <td>Product C</td>
                    <td>3</td>
                    <td>$15.00</td>
                </tr>
                <tr>
                    <td>Product D</td>
                    <td>1</td>
                    <td>$20.00</td>
                </tr>
            </tbody>
        </table>
        <div class="total-section">
            Total: $50.00
        </div>
        <div class="btn">Proceed to Checkout</div>
    </div>
</body>
</html>
```

```

        }
        .btn-checkout {
            background-color: #4CAF50;
            color: white;
        }
        .btn-continue {
            background-color: #2196F3;
            color: white;
        }
        .btn-remove {
            background-color: #f44336;
            color: white;
            padding: 5px 10px;
            font-size: 14px;
        }
        .btn-update {
            background-color: #FFC107;
            color: black;
            padding: 5px 10px;
            font-size: 14px;
        }
        .quantity-input {
            width: 50px;
            padding: 5px;
            text-align: center;
        }
        .message {
            padding: 10px;
            margin-bottom: 15px;
            border-radius: 4px;
        }
        .error-message {
            background-color: #ffdddd;
            color: #d8000c;
        }
        .success-message {
            background-color: #ddffdd;
            color: #4F8A10;
        }
        .empty-cart {
            text-align: center;
            padding: 20px;
            font-size: 18px;
        }
        .action-buttons {
            text-align: right;
            margin-top: 20px;
        }
    
```

</style>

</head>

<body>

<form id="form1" runat="server">

<div class="container">

<div class="cart-header">Shopping Cart</div>

<asp:Label ID="lblMessage" runat="server" CssClass="message" Visible="false"></asp:Label>

<asp:Panel ID="pnlCartItems" runat="server">

<asp:GridView ID="gvCart" runat="server" AutoGenerateColumns="False" CssClass="gridview"

```

        OnRowCommand="gvCart_RowCommand"
OnRowEditing="gvCart_RowEditing"
        OnRowCancelingEdit="gvCart_RowCancelingEdit"
DataKeyNames="Cart_ID">
    <Columns>
        <asp:BoundField DataField="Prod_Name" HeaderText="Product"
/>
        <asp:BoundField DataField="Price" HeaderText="Price"
DataFormatString="Rs. {0:N2}" />
        <asp:TemplateField HeaderText="Quantity">
            <ItemTemplate>
                <asp:TextBox ID="txtQuantity" runat="server"
Text='<%# Eval("Quantity") %>' CssClass="quantity-input" ></asp:TextBox>
            </ItemTemplate>
        </asp:TemplateField>
        <asp:BoundField DataField="Subtotal" HeaderText="Subtotal"
DataFormatString="Rs. {0:N2}" />
        <asp:TemplateField HeaderText="Actions">
            <ItemTemplate>
                <asp:Button ID="btnRemove" runat="server"
Text="Remove" CommandName="Remove" CommandArgument='<%# Eval("Cart_ID") %>' 
CssClass="btn-remove" />
            </ItemTemplate>
        </asp:TemplateField>
    </Columns>
</asp:GridView>

    <div class="total-section">
        <asp:Label ID="lblTotal" runat="server" Text="Total: Rs.
0.00"></asp:Label>
    </div>

    <div class="action-buttons">
        <asp:Button ID="btnContinueShopping" runat="server"
Text="Continue Shopping" CssClass="btn btn-continue"
OnClick="btnContinueShopping_Click" />
        <asp:Button ID="btnCheckout" runat="server" Text="Proceed to
Checkout" CssClass="btn btn-checkout" OnClick="btnCheckout_Click" />
    </div>
</asp:Panel>

<asp:Panel ID="pnlEmptyCart" runat="server" Visible="false"
CssClass="empty-cart">
    Your shopping cart is empty.
    <br /><br />
    <asp:Button ID="btnStartShopping" runat="server" Text="Start
Shopping" CssClass="btn btn-continue" OnClick="btnContinueShopping_Click" />
</asp:Panel>
</div>
</form>
</body>
</html>

```

### VB code:

```

Imports System.Data
Imports System.Data.SqlClient

```

```

Partial Class Cart
    Inherits System.Web.UI.Page

    Private connectionString As String = "Data Source=DESKTOP-1T8MKP4;Initial Catalog=Grocery_store;Integrated Security=True"

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As System.EventArgs)
Handles Me.Load
        If Session("UserID") Is Nothing Then
            Response.Redirect("User_Login.aspx")
            Return
        End If

        If Not IsPostBack Then
            LoadCartItems()
            CalculateTotal()
        End If
    End Sub

    Private Sub LoadCartItems()
        Dim userId As Integer = Convert.ToInt32(Session("UserID"))

        Using conn As New SqlConnection(connectionString)
            Dim cmd As New SqlCommand("SELECT c.Cart_ID, p.Product_ID, p.Prod_Name,
p.Price, c.Quantity, (p.Price * c.Quantity) AS Subtotal " &
                                     "FROM Cart c INNER JOIN Product p ON
c.Product_ID = p.Product_ID " &
                                     "WHERE c.Customer_ID = @CustomerID", conn)
                cmd.Parameters.AddWithValue("@CustomerID", userId)

                Dim da As New SqlDataAdapter(cmd)
                Dim dt As New DataTable()
                da.Fill(dt)

                If dt.Rows.Count > 0 Then
                    gvCart.DataSource = dt
                    gvCart.DataBind()
                    pnlEmptyCart.Visible = False
                    pnlCartItems.Visible = True
                Else
                    pnlEmptyCart.Visible = True
                    pnlCartItems.Visible = False
                End If
            End Using
        End Sub

        Private Sub CalculateTotal()
            Dim userId As Integer = Convert.ToInt32(Session("UserID"))
            Dim total As Decimal = 0

            Using conn As New SqlConnection(connectionString)
                Dim cmd As New SqlCommand("SELECT SUM(p.Price * c.Quantity) AS Total "
&
                                     "FROM Cart c INNER JOIN Product p ON
c.Product_ID = p.Product_ID " &
                                     "WHERE c.Customer_ID = @CustomerID", conn)
                cmd.Parameters.AddWithValue("@CustomerID", userId)
                conn.Open()

                Dim result = cmd.ExecuteScalar()
            End Using
        End Sub
    End Class

```

```

        If result IsNot DBNull.Value Then
            total = Convert.ToDecimal(result)
        End If
    End Using

    lblTotal.Text = String.Format("Total: Rs. {0:N2}", total)
End Sub

Protected Sub gvCart_RowEditing(sender As Object, e As GridViewEditEventArgs)
    ' Cancel the edit mode since we're handling updates manually
    gvCart.EditIndex = -1
    LoadCartItems()
End Sub

Protected Sub gvCart_RowCancelingEdit(sender As Object, e As
GridViewCancelEventArgs)
    ' Cancel the edit mode
    gvCart.EditIndex = -1
    LoadCartItems()
End Sub

Protected Sub gvCart_RowCommand(sender As Object, e As
GridViewCommandEventArgs)
    If e.CommandName = "Remove" Then
        Dim cartId As Integer = Convert.ToInt32(e.CommandArgument)
        RemoveFromCart(cartId)
        LoadCartItems()
        CalculateTotal()
    ElseIf e.CommandName = "Update" Then
        Dim rowIndex As Integer = Convert.ToInt32(e.CommandArgument)
        Dim row As GridViewRow = gvCart.Rows(rowIndex)
        Dim txtQuantity As TextBox = CType(row.FindControl("txtQuantity"),
TextBox)
        Dim cartId As Integer =
Convert.ToInt32(gvCart.DataKeys(row.RowIndex).Value)

        Dim newQuantity As Integer
        If Integer.TryParse(txtQuantity.Text, newQuantity) AndAlso newQuantity
> 0 Then
            UpdateCartItem(cartId, newQuantity)
            lblMessage.Text = "Quantity updated successfully."
            lblMessage.CssClass = "success-message"
            lblMessage.Visible = True
        Else
            lblMessage.Text = "Quantity must be a number greater than 0."
            lblMessage.CssClass = "error-message"
            lblMessage.Visible = True
            Return
        End If
        LoadCartItems()
        CalculateTotal()
    End If
End Sub

Private Sub RemoveFromCart(cartId As Integer)
    Using conn As New SqlConnection(connectionString)
        Dim cmd As New SqlCommand("DELETE FROM Cart WHERE Cart_ID = @CartID",
conn)
        cmd.Parameters.AddWithValue("@CartID", cartId)
        conn.Open()
    End Using
End Sub

```

```

        cmd.ExecuteNonQuery()
    End Using
End Sub

Private Sub UpdateCartItem(cartId As Integer, newQuantity As Integer)
    Using conn As New SqlConnection(connectionString)
        Dim cmd As New SqlCommand("UPDATE Cart SET Quantity = @Quantity WHERE
Cart_ID = @CartID", conn)
        cmd.Parameters.AddWithValue("@Quantity", newQuantity)
        cmd.Parameters.AddWithValue("@CartID", cartId)
        conn.Open()
        cmd.ExecuteNonQuery()
    End Using
End Sub

Protected Sub btnCheckout_Click(sender As Object, e As EventArgs)
    If gvCart.Rows.Count = 0 Then
        lblMessage.Text = "Your cart is empty. Please add items before
checkout."
        lblMessage.CssClass = "error-message"
        lblMessage.Visible = True
        Return
    End If

    Response.Redirect("Checkout.aspx")
End Sub

Protected Sub btnContinueShopping_Click(sender As Object, e As EventArgs)
    Response.Redirect("User_Dash.aspx")
End Sub
End Class

```

## Checkout:

```

<%@ Page Language="VB" AutoEventWireup="false" CodeFile="Checkout.aspx.vb"
Inherits="Checkout" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Checkout</title>
    <style type="text/css">
        body {
            font-family: Arial, sans-serif;
            margin: 0;
            padding: 0;
            background-color: #f5f5f5;
        }
        .container {
            width: 80%;
            margin: 20px auto;
            background-color: white;
            padding: 20px;
            border-radius: 5px;
            box-shadow: 0 0 10px rgba(0,0,0,0.1);
        }
    </style>
</head>
<body>
    <div class="container">
        <h1>Checkout</h1>
        <p>This is the checkout page. Please enter your payment information below.</p>
        <form>
            <label>Name:</label>
            <input type="text" name="name" />
            <label>Address:</label>
            <input type="text" name="address" />
            <label>City:</label>
            <input type="text" name="city" />
            <label>State/Zip:</label>
            <input type="text" name="statezip" />
            <label>Credit Card Number:</label>
            <input type="text" name="ccn" />
            <label>Expiration Date:</label>
            <input type="text" name="expdate" />
            <label>CVV:</label>
            <input type="text" name="cvv" />
            <input type="submit" value="Place Order" />
        </form>
    </div>
</body>
</html>

```

```
.checkout-header {
    font-size: 24px;
    color: #333;
    margin-bottom: 20px;
    padding-bottom: 10px;
    border-bottom: 1px solid #eee;
}
.section {
    margin-bottom: 30px;
    padding-bottom: 20px;
    border-bottom: 1px solid #eee;
}
.section-title {
    font-size: 18px;
    color: #4CAF50;
    margin-bottom: 15px;
}
.gridView {
    width: 100%;
    border-collapse: collapse;
    margin-bottom: 20px;
}
.gridView th {
    background-color: #4CAF50;
    color: white;
    padding: 10px;
    text-align: left;
}
.gridView td {
    padding: 10px;
    border-bottom: 1px solid #ddd;
}
.gridView tr:nth-child(even) {
    background-color: #f9f9f9;
}
.total-section {
    text-align: right;
    font-size: 18px;
    font-weight: bold;
    margin: 20px 0;
}
.form-group {
    margin-bottom: 15px;
}
.form-group label {
    display: block;
    margin-bottom: 5px;
    font-weight: bold;
}
.form-control {
    width: 100%;
    padding: 8px;
    border: 1px solid #ddd;
    border-radius: 4px;
    box-sizing: border-box;
}
.btn {
    padding: 10px 20px;
    border: none;
    border-radius: 3px;
    cursor: pointer;
}
```

```

        font-size: 16px;
        background-color: #4CAF50;
        color: white;
    }
    .message {
        padding: 10px;
        margin-bottom: 15px;
        border-radius: 4px;
    }
    .error-message {
        background-color: #ffdddd;
        color: #d8000c;
    }
    .success-message {
        background-color: #ddffdd;
        color: #4F8A10;
    }

```

</style>

</head>

<body>

<form id="form1" runat="server">

<div class="container">

<div class="checkout-header">Checkout</div>

<asp:Label ID="lblMessage" runat="server" CssClass="message" Visible="false"></asp:Label>

<div class="section">

<div class="section-title">Order Summary</div>

<asp:GridView ID="gvOrderItems" runat="server" AutoGenerateColumns="False" CssClass="gridview">

<Columns>

<asp:BoundField DataField="Prod\_Name" HeaderText="Product" />

<asp:BoundField DataField="Price" HeaderText="Price" DataFormatString="Rs. {0:N2}" />

<asp:BoundField DataField="Quantity" HeaderText="Qty" />

<asp:BoundField DataField="Subtotal" HeaderText="Subtotal" DataFormatString="Rs. {0:N2}" />

</Columns>

</asp:GridView>

<div class="total-section">

<div>Subtotal: <asp:Label ID="lblSubtotal" runat="server" Text="Rs. 0.00"></asp:Label></div>

<div>Total: <asp:Label ID="lblTotal" runat="server" Text="Rs. 0.00"></asp:Label></div>

</div>

<div class="section">

<div class="section-title">Shipping Information</div>

<div class="form-group">

<label for="txtShippingAddress">Shipping Address</label>

<asp:TextBox ID="txtShippingAddress" runat="server" CssClass="form-control" TextMode="MultiLine" Rows="3" required="true"></asp:TextBox>

<asp:RequiredFieldValidator ID="rfvShippingAddress" runat="server" />

```

        ControlToValidate="txtShippingAddress"
ErrorMessage="Shipping address is required"
Display="Dynamic" CssClass="error-
message">></asp:RequiredFieldValidator>
</div>

<div class="form-group">
    <label for="txtPhone">Phone Number</label>
    <asp:TextBox ID="txtPhone" runat="server" CssClass="form-
control" required="true"></asp:TextBox>
    <asp:RequiredFieldValidator ID="rfvPhone" runat="server"
        ControlToValidate="txtPhone" ErrorMessage="Phone number is
required"
        Display="Dynamic" CssClass="error-
message">></asp:RequiredFieldValidator>
    <asp:RegularExpressionValidator ID="revPhone" runat="server"
        ControlToValidate="txtPhone" ErrorMessage="Enter a valid
phone number"
        ValidationExpression="^[\d]{10,15}$" Display="Dynamic"
        CssClass="error-message">></asp:RegularExpressionValidator>
</div>

<div class="form-group">
    <label for="ddlShippingMethod">Shipping Method</label>
    <asp:DropDownList ID="ddlShippingMethod" runat="server"
        CssClass="form-control">
        <asp:ListItem Text="Standard Shipping (3-5 business days)"
        Value="Standard"></asp:ListItem>
        <asp:ListItem Text="Express Shipping (1-2 business days)"
        Value="Express"></asp:ListItem>
    </asp:DropDownList>
</div>
</div>

<div class="form-group" style="text-align: right;">
    <asp:Button ID="btnPlaceOrder" runat="server" Text="Place Order"
        CssClass="btn" OnClick="btnPlaceOrder_Click" />
</div>
</div>
</form>
</body>
</html>

```

### VB code:

```

Imports System.Data
Imports System.Data.SqlClient

Partial Class Checkout
    Inherits System.Web.UI.Page

    Private connectionString As String = "Data Source=DESKTOP-1T8MKP4;Initial
Catalog=Grocery_store;Integrated Security=True"

    Protected Sub Page_Load(ByVal sender As Object, ByVal e As EventArgs) Handles
Me.Load
        If Session("UserID") Is Nothing Then
            Response.Redirect("User_Login.aspx")
            Return
        End If

```

```

If Not IsPostBack Then
    LoadCartItems()
    CalculateTotal()
    LoadUserAddress()
End If
End Sub

Private Sub LoadUserAddress()
    Dim userId As Integer = Convert.ToInt32(Session("UserID"))

    Using conn As New SqlConnection(connectionString)
        Dim cmd As New SqlCommand("SELECT Cust_Address FROM Customer WHERE
Customer_ID = @CustomerID", conn)
        cmd.Parameters.AddWithValue("@CustomerID", userId)

        conn.Open()
        Dim result = cmd.ExecuteScalar()
        If result IsNot Nothing AndAlso Not IsDBNull(result) Then
            txtShippingAddress.Text = result.ToString()
        End If
    End Using
End Sub

Private Sub LoadCartItems()
    Dim userId As Integer = Convert.ToInt32(Session("UserID"))

    Using conn As New SqlConnection(connectionString)
        Dim cmd As New SqlCommand("SELECT p.Prod_Name, p.Price, c.Quantity,
(p.Price * c.Quantity) AS Subtotal " &
                           "FROM Cart c INNER JOIN Product p ON c.Product_ID
= p.Product_ID " &
                           "WHERE c.Customer_ID = @CustomerID", conn)
        cmd.Parameters.AddWithValue("@CustomerID", userId)

        Dim da As New SqlDataAdapter(cmd)
        Dim dt As New DataTable()
        da.Fill(dt)

        gvOrderItems.DataSource = dt
        gvOrderItems.DataBind()
    End Using
End Sub

Private Sub CalculateTotal()
    Dim userId As Integer = Convert.ToInt32(Session("UserID"))
    Dim subtotal As Decimal = 0

    Using conn As New SqlConnection(connectionString)
        Dim cmd As New SqlCommand("SELECT SUM(p.Price * c.Quantity) AS Total "
&
                           "FROM Cart c INNER JOIN Product p ON c.Product_ID
= p.Product_ID " &
                           "WHERE c.Customer_ID = @CustomerID", conn)
        cmd.Parameters.AddWithValue("@CustomerID", userId)

        conn.Open()
        Dim result = cmd.ExecuteScalar()
        If result IsNot Nothing AndAlso Not IsDBNull(result) Then
            subtotal = Convert.ToDecimal(result)
        End If
    End Using

```

```

        lblSubtotal.Text = String.Format("Rs. {0:N2}", subtotal)
        lblTotal.Text = lblSubtotal.Text
    End Sub

    Protected Sub btnPlaceOrder_Click(sender As Object, e As EventArgs) Handles
btnPlaceOrder.Click
        If Not Page.IsValid Then Exit Sub

        Dim userId As Integer = Convert.ToInt32(Session("UserID"))

        Using conn As New SqlConnection(connectionString)
            conn.Open()
            Dim transaction As SqlTransaction = conn.BeginTransaction()

            Try
                ' 1. Create Billing Info
                Dim billingId As Integer = Convert.ToInt32(New SqlCommand("SELECT
ISNULL(MAX(Billing_ID), 0) + 1 FROM Billing_info", conn,
transaction).ExecuteScalar())

                Dim billingCmd As New SqlCommand("INSERT INTO Billing_info
(Billing_ID, Bill_Date, Customer_ID) VALUES (@BillingID, GETDATE(), @CustomerID)", conn,
transaction)
                billingCmd.Parameters.AddWithValue("@BillingID", billingId)
                billingCmd.Parameters.AddWithValue("@CustomerID", userId)
                billingCmd.ExecuteNonQuery()

                ' 2. Create Address record if different from customer address
                Dim customerAddress As String = ""
                Dim getAddressCmd As New SqlCommand("SELECT Cust_Address FROM
Customer WHERE Customer_ID = @CustomerID", conn, transaction)
                getAddressCmd.Parameters.AddWithValue("@CustomerID", userId)
                customerAddress = getAddressCmd.ExecuteScalar().ToString()

                If Not
txtShippingAddress.Text.Trim().Equals(customerAddress.Trim(),
StringComparison.OrdinalIgnoreCase) Then
                    Dim addressId As Integer = Convert.ToInt32(New
SqlCommand("SELECT ISNULL(MAX(Address_ID), 0) + 1 FROM Address", conn,
transaction).ExecuteScalar())

                    Dim addressCmd As New SqlCommand("INSERT INTO Address
(Address_ID, Customer_ID, Bill_Address) VALUES (@AddressID, @CustomerID,
@Address)", conn, transaction)
                    addressCmd.Parameters.AddWithValue("@AddressID", addressId)
                    addressCmd.Parameters.AddWithValue("@CustomerID", userId)
                    addressCmd.Parameters.AddWithValue("@Address",
txtShippingAddress.Text.Trim())
                    addressCmd.ExecuteNonQuery()
                End If

                ' 3. Create Order
                Dim orderId As Integer = Convert.ToInt32(New SqlCommand("SELECT
ISNULL(MAX(Order_ID), 0) + 1 FROM [Customer_Order]", conn,
transaction).ExecuteScalar())

                Dim orderCmd As New SqlCommand("INSERT INTO [Customer_Order]
(Order_ID, Order_Date, Customer_ID, Shipping_Method, Billing_ID) VALUES (@OrderID,
GETDATE(), @CustomerID, @ShippingMethod, @BillingID)", conn, transaction)
                orderCmd.Parameters.AddWithValue("@OrderID", orderId)
            End Try
        End Using
    End Sub

```

```

        orderCmd.Parameters.AddWithValue("@CustomerID", userId)
        orderCmd.Parameters.AddWithValue("@ShippingMethod",
ddlShippingMethod.SelectedValue)
        orderCmd.Parameters.AddWithValue("@BillingID", billingId)
        orderCmd.ExecuteNonQuery()

        ' 4. Process Order Details
        Dim cartItems As New DataTable()
        Dim getCartCmd As New SqlCommand("SELECT c.Product_ID, c.Quantity,
p.Price, p.Quantity AS AvailableQty FROM Cart c INNER JOIN Product p ON
c.Product_ID = p.Product_ID WHERE c.Customer_ID = @CustomerID", conn, transaction)
        getCartCmd.Parameters.AddWithValue("@CustomerID", userId)

        Using da As New SqlDataAdapter(getCartCmd)
            da.Fill(cartItems)
        End Using

        For Each row As DataRow In cartItems.Rows
            Dim productId As Integer = Convert.ToInt32(row("Product_ID"))
            Dim quantity As Integer = Convert.ToInt32(row("Quantity"))
            Dim availableQty As Integer =
Convert.ToInt32(row("AvailableQty"))
            Dim price As Decimal = Convert.ToDecimal(row("Price"))

            If quantity > availableQty Then
                Throw New Exception(String.Format("Not enough stock for
product ID {0}. Available: {1}, Requested: {2}", productId, availableQty,
quantity))
            End If

            ' Insert Order Details
            Dim orderDetailsId As Integer = Convert.ToInt32(New
SqlCommand("SELECT ISNULL(MAX(Order_Details_ID), 0) + 1 FROM Order_Details", conn,
transaction).ExecuteScalar())

            Dim orderDetailsCmd As New SqlCommand("INSERT INTO
Order_Details (Order_Details_ID, Product_ID, Order_ID, Quantity) VALUES
(@OrderDetailsID, @ProductID, @OrderID, @Quantity)", conn, transaction)
            orderDetailsCmd.Parameters.AddWithValue("@OrderDetailsID",
orderDetailsId)
            orderDetailsCmd.Parameters.AddWithValue("@ProductID",
productId)
            orderDetailsCmd.Parameters.AddWithValue("@OrderID", orderId)
            orderDetailsCmd.Parameters.AddWithValue("@Quantity", quantity)
            orderDetailsCmd.ExecuteNonQuery()

            ' Update Product Inventory
            Dim updateProductCmd As New SqlCommand("UPDATE Product SET
Quantity = Quantity - @Quantity WHERE Product_ID = @ProductID", conn, transaction)
            updateProductCmd.Parameters.AddWithValue("@Quantity", quantity)
            updateProductCmd.Parameters.AddWithValue("@ProductID",
productId)
            updateProductCmd.ExecuteNonQuery()
        Next

        ' 5. Clear the cart
        Dim clearCartCmd As New SqlCommand("DELETE FROM Cart WHERE
Customer_ID = @CustomerID", conn, transaction)
        clearCartCmd.Parameters.AddWithValue("@CustomerID", userId)
        clearCartCmd.ExecuteNonQuery()
    
```

```

        transaction.Commit()

        ' Display success message
        lblMessage.CssClass = "success-message message"
        lblMessage.Text = String.Format("Order placed successfully! Order
ID: {0}", orderId)
        lblMessage.Visible = True

        ' Clear the UI
        gvOrderItems.DataSource = Nothing
        gvOrderItems.DataBind()
        lblSubtotal.Text = "Rs. 0.00"
        lblTotal.Text = "Rs. 0.00"
        btnPlaceOrder.Enabled = False

    Catch ex As Exception
        transaction.Rollback()
        lblMessage.CssClass = "error-message message"
        lblMessage.Text = "An error occurred while placing the order: " &
ex.Message
        lblMessage.Visible = True
    End Try
End Using
End Sub
End Class

```

## Order\_confirmation:

```

<%@ Page Language="VB" AutoEventWireup="false" CodeFile="OrderConfirmation.aspx.vb"
Inherits="OrderConfirmation" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Order Confirmation</title>
    <style type="text/css">
        body {
            font-family: Arial, sans-serif;
            margin: 0;
            padding: 0;
            background-color: #f5f5f5;
        }
        .container {
            width: 80%;
            margin: 20px auto;
            background-color: white;
            padding: 20px;
            border-radius: 5px;
            box-shadow: 0 0 10px rgba(0,0,0,0.1);
        }
        .confirmation-header {
            font-size: 24px;
            color: #4CAF50;
            margin-bottom: 20px;
            padding-bottom: 10px;
            border-bottom: 1px solid #eee;
        }
        .thank-you {
            font-size: 18px;

```

```
        margin-bottom: 20px;
    }
    .order-info {
        margin-bottom: 30px;
    }
    .info-item {
        margin-bottom: 10px;
    }
    .info-label {
        font-weight: bold;
        display: inline-block;
        width: 150px;
    }
}
\gridview {
    width: 100%;
    border-collapse: collapse;
    margin-bottom: 20px;
}
\gridview th {
    background-color: #4CAF50;
    color: white;
    padding: 10px;
    text-align: left;
}
\gridview td {
    padding: 10px;
    border-bottom: 1px solid #ddd;
}
\gridview tr:nth-child(even) {
    background-color: #f9f9f9;
}
.total-section {
    text-align: right;
    font-size: 18px;
    font-weight: bold;
    margin: 20px 0;
}
.action-buttons {
    text-align: center;
    margin-top: 30px;
}
.btn {
    padding: 10px 20px;
    border: none;
    border-radius: 3px;
    cursor: pointer;
    font-size: 16px;
    background-color: #4CAF50;
    color: white;
}

```

</style>

</head>

<body>

<form id="form1" runat="server">

<div class="container">

<div class="confirmation-header">Order Confirmation</div>

<div class="thank-you">

Thank you for your order! Your purchase has been confirmed.

</div>

```

<div class="order-info">
    <div class="info-item">
        <span class="info-label">Order Number:</span>
        <asp:Label ID="lblOrderId" runat="server"></asp:Label>
    </div>
    <div class="info-item">
        <span class="info-label">Order Date:</span>
        <asp:Label ID="lblOrderDate" runat="server"></asp:Label>
    </div>
    <div class="info-item">
        <span class="info-label">Shipping Address:</span>
        <asp:Label ID="lblShippingAddress" runat="server"></asp:Label>
    </div>
    <div class="info-item">
        <span class="info-label">Shipping Method:</span>
        <asp:Label ID="lblShippingMethod" runat="server"></asp:Label>
    </div>
</div>

<div>
    <asp:GridView ID="gvOrderItems" runat="server"
AutoGenerateColumns="False" CssClass="gridview">
        <Columns>
            <asp:BoundField DataField="Prod_Name" HeaderText="Product"
/>
            <asp:BoundField DataField="Price" HeaderText="Price"
DataFormatString="Rs. {0:N2}" />
            <asp:BoundField DataField="Quantity" HeaderText="Qty" />
            <asp:BoundField DataField="Subtotal" HeaderText="Subtotal"
DataFormatString="Rs. {0:N2}" />
        </Columns>
    </asp:GridView>

    <div class="total-section">
        <div>Total: <asp:Label ID="lblTotal" runat="server" Text="Rs.
0.00"></asp:Label></div>
    </div>
</div>

<div class="action-buttons">
    <asp:Button ID="btnContinueShopping" runat="server" Text="Continue
Shopping" CssClass="btn" OnClick="btnContinueShopping_Click" />
</div>
</div>
</form>
</body>
</html>

```

### VB code:

```

Imports System.Data
Imports System.Data.SqlClient

Partial Class OrderConfirmation
    Inherits System.Web.UI.Page

    Private connectionString As String = "Data Source=DESKTOP-1T8MKP4;Initial
Catalog=Grocery_store;Integrated Security=True"

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

```

```

If Session("UserID") Is Nothing Then
    Response.Redirect("User_Login.aspx")
    Return
End If

If Not IsPostBack Then
    If Session("OrderID") Is Nothing Then
        Response.Redirect("User_Dash.aspx")
    Else
        DisplayOrderConfirmation(Convert.ToInt32(Session("OrderID")))
        Session.Remove("OrderID")
    End If
End If
End Sub

Private Sub DisplayOrderConfirmation(orderId As Integer)
    Dim userId As Integer = Convert.ToInt32(Session("UserID"))

    Using conn As New SqlConnection(connectionString)
        ' Get order summary
        Dim cmd As New SqlCommand("SELECT o.Order_ID, o.Order_Date,
o.Shipping_Method, a.Bill_Address " &
                               "FROM [Customer_Order] o " &
                               "JOIN Billing_info b ON o.Billing_ID =
b.Billing_ID " &
                               "JOIN Address a ON b.Customer_ID =
a.Customer_ID " &
                               "WHERE o.Order_ID = @OrderID AND o.Customer_ID =
= @CustomerID", conn)
        cmd.Parameters.AddWithValue("@OrderID", orderId)
        cmd.Parameters.AddWithValue("@CustomerID", userId)

        conn.Open()
        Dim reader As SqlDataReader = cmd.ExecuteReader()

        If reader.Read() Then
            lblOrderId.Text = reader("Order_ID").ToString()
            lblOrderDate.Text =
Convert.ToDateTime(reader("Order_Date")).ToString("MMMM dd, yyyy")
            lblShippingAddress.Text = reader("Bill_Address").ToString()
            lblShippingMethod.Text = reader("Shipping_Method").ToString()
        End If
        reader.Close()

        ' Get order items
        Dim itemsCmd As New SqlCommand("SELECT p.Prod_Name, od.Price,
od.Quantity, (od.Price * od.Quantity) AS Subtotal " &
                                       "FROM Order_Details od " &
                                       "JOIN Product p ON od.Product_ID =
p.Product_ID " &
                                       "WHERE od.Order_ID = @OrderID", conn)
        itemsCmd.Parameters.AddWithValue("@OrderID", orderId)

        Dim da As New SqlDataAdapter(itemsCmd)
        Dim dt As New DataTable()
        da.Fill(dt)

        gvOrderItems.DataSource = dt
        gvOrderItems.DataBind()

        ' Calculate total
    End Using
End Sub

```

```

        Dim totalCmd As New SqlCommand("SELECT SUM(Price * Quantity) AS Total
FROM Order_Details WHERE Order_ID = @OrderID", conn)
        totalCmd.Parameters.AddWithValue("@OrderID", orderId)

        Dim total As Decimal = Convert.ToDecimal(totalCmd.ExecuteScalar())
        lblTotal.Text = String.Format("Rs. {0:N2}", total)
    End Using
End Sub

Protected Sub btnContinueShopping_Click(sender As Object, e As EventArgs)
    Response.Redirect("User_Dash.aspx")
End Sub
End Class

```

## Order\_History:

```

<%@ Page Language="VB" AutoEventWireup="false" CodeFile="OrderHistory.aspx.vb"
Inherits="OrderHistory" %>

<!DOCTYPE html>

<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
    <title>Grocery Store - My Orders</title>
    <style type="text/css">
        body {
            font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
            margin: 0;
            padding: 0;
            background-color: #f8f9fa;
        }
        .header {
            background-color: #28a745;
            color: white;
            padding: 15px 0;
            box-shadow: 0 2px 5px rgba(0,0,0,0.1);
        }
        .container {
            width: 90%;
            max-width: 1200px;
            margin: 0 auto;
        }
        .header-content {
            display: flex;
            justify-content: space-between;
            align-items: center;
        }
        .logo {
            font-size: 24px;
            font-weight: bold;
        }
        .user-nav {
            display: flex;
            align-items: center;
            gap: 15px;
        }
        .btn-nav {
            padding: 8px 15px;
            border: none;
            border-radius: 4px;
        }
    </style>
</head>
<body>
    <div class="header">
        <div class="header-content">
            <img alt="Grocery Store Logo" class="logo" />
            <div>My Orders</div>
            <div><a href="#">Logout</a></div>
        </div>
    </div>
    <div class="container">
        <div>
            <table border="1">
                <thead>
                    <tr>
                        <th>Order ID</th>
                        <th>Product Name</th>
                        <th>Quantity</th>
                        <th>Unit Price</th>
                        <th>Subtotal</th>
                    </tr>
                </thead>
                <tbody>
                    <tr>
                        <td>OD-001</td>
                        <td>Apples</td>
                        <td>5</td>
                        <td>₹10.00</td>
                        <td>₹50.00</td>
                    </tr>
                    <tr>
                        <td>OD-002</td>
                        <td>Bananas</td>
                        <td>3</td>
                        <td>₹8.00</td>
                        <td>₹24.00</td>
                    </tr>
                    <tr>
                        <td>OD-003</td>
                        <td>Oranges</td>
                        <td>4</td>
                        <td>₹12.00</td>
                        <td>₹48.00</td>
                    </tr>
                    <tr>
                        <td>OD-004</td>
                        <td>Grapes</td>
                        <td>2</td>
                        <td>₹15.00</td>
                        <td>₹30.00</td>
                    </tr>
                    <tr>
                        <td>OD-005</td>
                        <td>Milk</td>
                        <td>1</td>
                        <td>₹20.00</td>
                        <td>₹20.00</td>
                    </tr>
                </tbody>
            </table>
        </div>
        <div>
            <div>Subtotal:</div>
            <div>₹150.00</div>
            <div><a href="#">Proceed to Payment</a></div>
        </div>
    </div>
</body>

```

```
        cursor: pointer;
        font-size: 14px;
        transition: all 0.3s;
    }
    .btn-cart {
        background-color: #ffc107;
        color: #212529;
    }
    .btn-dashboard {
        background-color: #17a2b8;
        color: white;
    }
    .btn-logout {
        background-color: #dc3545;
        color: white;
    }
    .btn-nav:hover {
        opacity: 0.9;
        transform: translateY(-2px);
    }
    .orders-container {
        margin: 20px 0;
    }
    .order-card {
        background-color: white;
        padding: 20px;
        border-radius: 5px;
        margin-bottom: 20px;
        box-shadow: 0 2px 5px rgba(0,0,0,0.1);
    }
    .order-header {
        display: flex;
        justify-content: space-between;
        margin-bottom: 10px;
        padding-bottom: 10px;
        border-bottom: 1px solid #eee;
    }
    .order-id {
        font-weight: bold;
        color: #333;
    }
    .order-date {
        color: #6c757d;
    }
    .order-status {
        padding: 5px 10px;
        border-radius: 4px;
        font-weight: bold;
    }
    .status-pending {
        background-color: #fff3cd;
        color: #856404;
    }
    .status-completed {
        background-color: #d4edda;
        color: #155724;
    }
    .status-cancelled {
        background-color: #f8d7da;
        color: #721c24;
    }
}
```

```

.order-items {
    width: 100%;
    border-collapse: collapse;
    margin-top: 10px;
}
.order-items th {
    text-align: left;
    padding: 8px;
    background-color: #f2f2f2;
}
.order-items td {
    padding: 8px;
    border-bottom: 1px solid #ddd;
}
.order-total {
    text-align: right;
    margin-top: 10px;
    font-weight: bold;
    font-size: 18px;
}
.no-orders {
    text-align: center;
    padding: 20px;
    background-color: white;
    border-radius: 5px;
    box-shadow: 0 2px 5px rgba(0,0,0,0.1);
}

```

</style>

</head>

<body>

<form id="form1" runat="server">

<div class="header">

<div class="container header-content">

<div class="logo">Grocery Store</div>

<div class="user-nav">

<asp:Button ID="btnViewCart" runat="server" Text="View Cart" CssClass="btn-nav btn-cart" OnClick="btnViewCart\_Click" />

<asp:Button ID="btnDashboard" runat="server" Text="Dashboard" CssClass="btn-nav btn-dashboard" OnClick="btnDashboard\_Click" />

<asp:Button ID="btnLogout" runat="server" Text="Logout" CssClass="btn-nav btn-logout" OnClick="btnLogout\_Click" />

</div>

</div>

</div>

<div class="container">

<h1>My Orders</h1>

<div class="orders-container">

<asp:Panel ID="pnlOrders" runat="server">

<asp:Literal ID="litOrders" runat="server"></asp:Literal>

</asp:Panel>

</div>

</div>

</form>

</body>

</html>

**VB code:**



```

        Using itemsConn As New SqlConnection(conn.ConnectionString)
            Dim itemsCmd As New SqlCommand("SELECT p.Prod_Name,
oi.Quantity, p.Price " &
                "FROM Order_Details oi " &
                "INNER JOIN Product p ON oi.Product_ID = p.Product_ID
" &
                "WHERE oi.Order_ID = @OrderID", itemsConn)

            itemsCmd.Parameters.AddWithValue("@OrderID", orderId)

            itemsConn.Open()
            Dim itemsReader As SqlDataReader =
itemsCmd.ExecuteReader()

            If itemsReader.HasRows Then
                itemsHtml.Append("<div class='order-items-
container'>")
                itemsHtml.Append("<table class='order-items'>")

                itemsHtml.Append("<thead><tr><th>Product</th><th>Quantity</th><th>Unit
Price</th><th>Total</th></tr></thead>")
                itemsHtml.Append("<tbody>")

                While itemsReader.Read()
                    Dim productName As String =
itemsReader("Prod_Name").ToString()
                    Dim quantity As Integer =
Convert.ToInt32(itemsReader("Quantity"))
                    Dim price As Decimal =
Convert.ToDecimal(itemsReader("Price"))
                    Dim itemTotal As Decimal = quantity * price
                    totalAmount += itemTotal

                    itemsHtml.Append("<tr>")
                    itemsHtml.AppendFormat("<td>{0}</td>",
productName)
                    itemsHtml.AppendFormat("<td>{0}</td>",
quantity)
                    itemsHtml.AppendFormat("<td>Rs. {0}</td>",
price.ToString("N2"))
                    itemsHtml.AppendFormat("<td>Rs. {0}</td>",
itemTotal.ToString("N2"))
                    itemsHtml.Append("</tr>")
                End While

                itemsHtml.Append("</tbody>")
                itemsHtml.Append("</table>")

                ' Order summary
                itemsHtml.Append("<div class='order-summary'>")
                itemsHtml.AppendFormat("<div class='order-
total'>Total Amount: <span>Rs. {0}</span></div>", totalAmount.ToString("N2"))
                itemsHtml.Append("</div>") ' Close order-summary

                itemsHtml.Append("</div>") ' Close order-items-
container
            Else
                itemsHtml.Append("<div class='no-items'>No items
found in this order.</div>")
            End If

```

```

        End Using

        html.Append(itemsHtml.ToString())
        html.Append("</div>") ' Close order-card
    End While
End If
Catch ex As Exception
    html.Append("<div class='error-message'>Error loading orders: " &
Server.HtmlEncode(ex.Message) & "</div>")
End Try
End Using

litOrders.Text = html.ToString()
End Sub

Protected Sub btnViewCart_Click(sender As Object, e As EventArgs) Handles
btnViewCart.Click
    Response.Redirect("Cart.aspx")
End Sub

Protected Sub btnDashboard_Click(sender As Object, e As EventArgs) Handles
btnDashboard.Click
    Response.Redirect("User_Dash.aspx")
End Sub

Protected Sub btnLogout_Click(sender As Object, e As EventArgs) Handles
btnLogout.Click
    Session.Clear()
    Response.Redirect("User_Login.aspx")
End Sub
End Class

```

## Software requirements specification:

The system will:

- Allow customers to browse, order, and pay for groceries online
- Enable suppliers to manage their product listings
- Provide administrators with tools to manage users, products, and orders

### 1.1 Product Perspective

**Web-based application** with three main user roles:

- **Customers:** Browse and purchase products
- **Suppliers:** Manage product listings
- **Administrators:** Oversee system operations

## 2.2 User Classes

User Role	Description
Customer	Shops online, tracks orders
Supplier	Manages product listings, views orders
Administrator	Manages users, suppliers, products, and orders

## 2.3 Operating Environment

- **Frontend:** ASP.NET Web Forms
- **Backend:** SQL Server Database
- **Hosting:** Windows Server with IIS

# System Features & Requirements

## 3.1 Functional Requirements

### 3.1.1 Customer Requirements

ID	Requirement	Priority
FR1	Register a new account	High
FR2	Login to the system	High
FR3	Browse available products	High
FR4	Add products to shopping cart	High
FR5	View and modify cart	Medium
FR6	Checkout and place order	High
FR7	View order history	Medium

### 3.1.2 Supplier Requirements

ID	Requirement	Priority
FR8	Add new products	High
FR9	Update product details (price, stock)	High
FR10	Delete discontinued products	Medium
FR11	View orders for their products	Medium

### 3.1.3 Administrator Requirements

ID	Requirement	Priority
FR12	Manage customer accounts	High
FR13	Approve/remove suppliers	High
FR14	Manage product categories	Medium
FR15	Track and update order statuses	High

## 3.2 Non-Functional Requirements

Category	Requirement
Performance	Support 100+ concurrent users
Security	Password encryption, SQL injection prevention
Usability	Mobile-friendly interface

<b>Reliability</b>	99.9% uptime
<b>Scalability</b>	Support future feature additions

## External Interface Requirements

### 4.1 User Interfaces

- **Customer:** Product catalog, shopping cart, checkout
- **Supplier:** Product management dashboard
- **Administrator:** Admin panel for user and order management

### 4.2 Hardware Interfaces

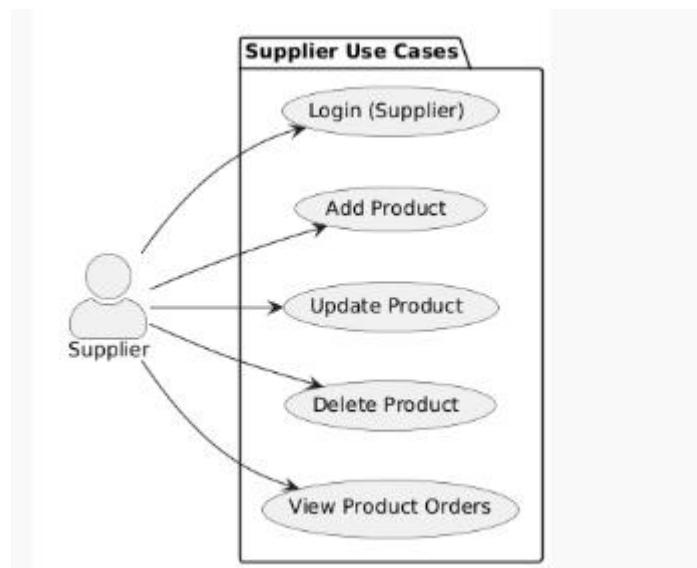
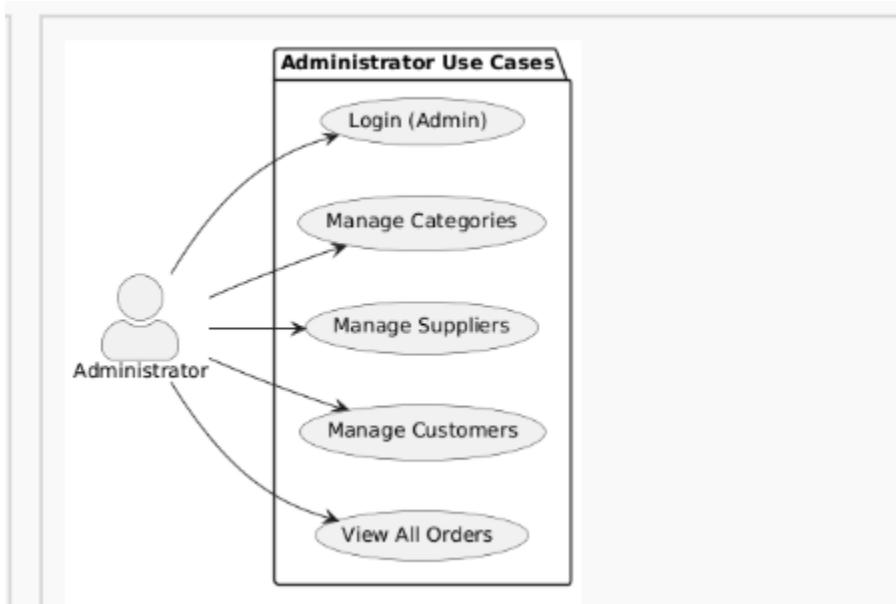
- **Server:** Windows Server, SQL Database
- **Client:** Web browsers (Chrome, Firefox, Edge)

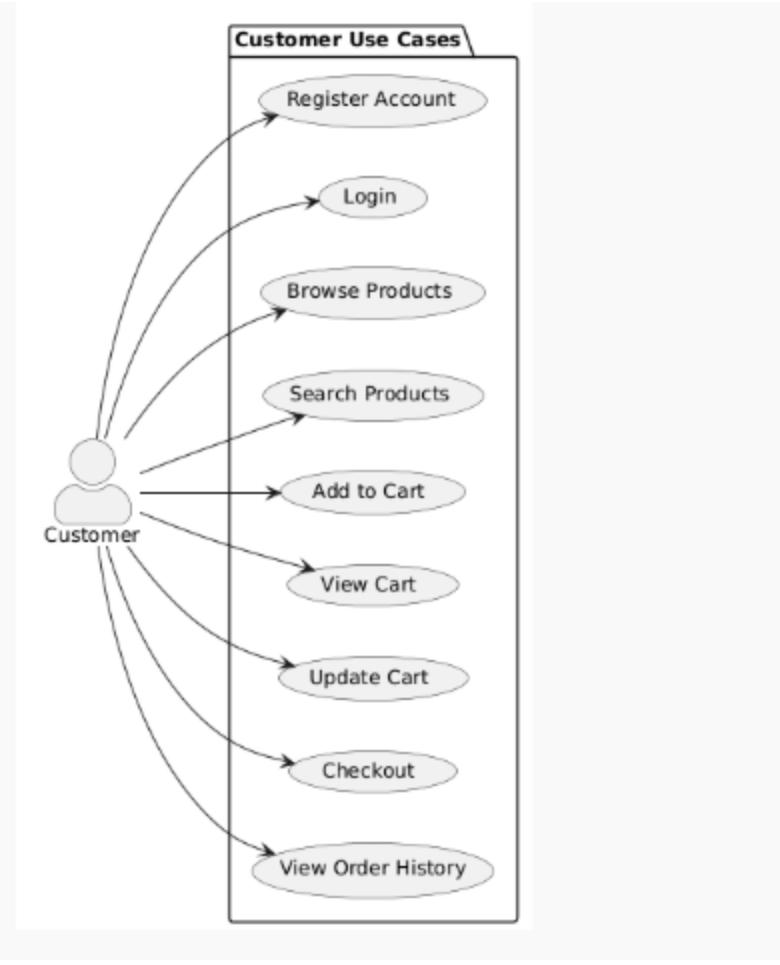
### 4.4 Communication Interfaces

- HTTPS for secure transaction

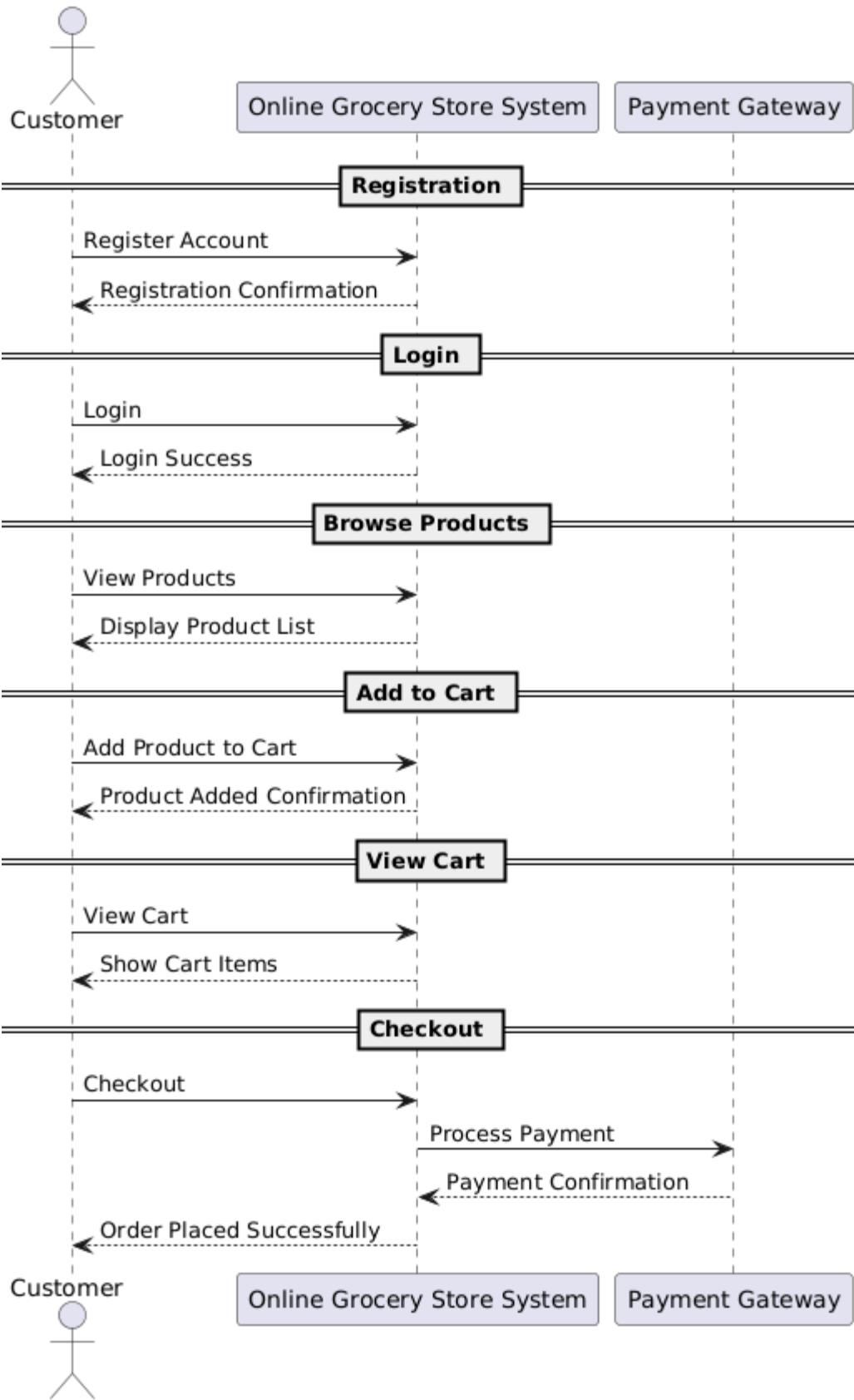
## To develop the analysis and design model of web application

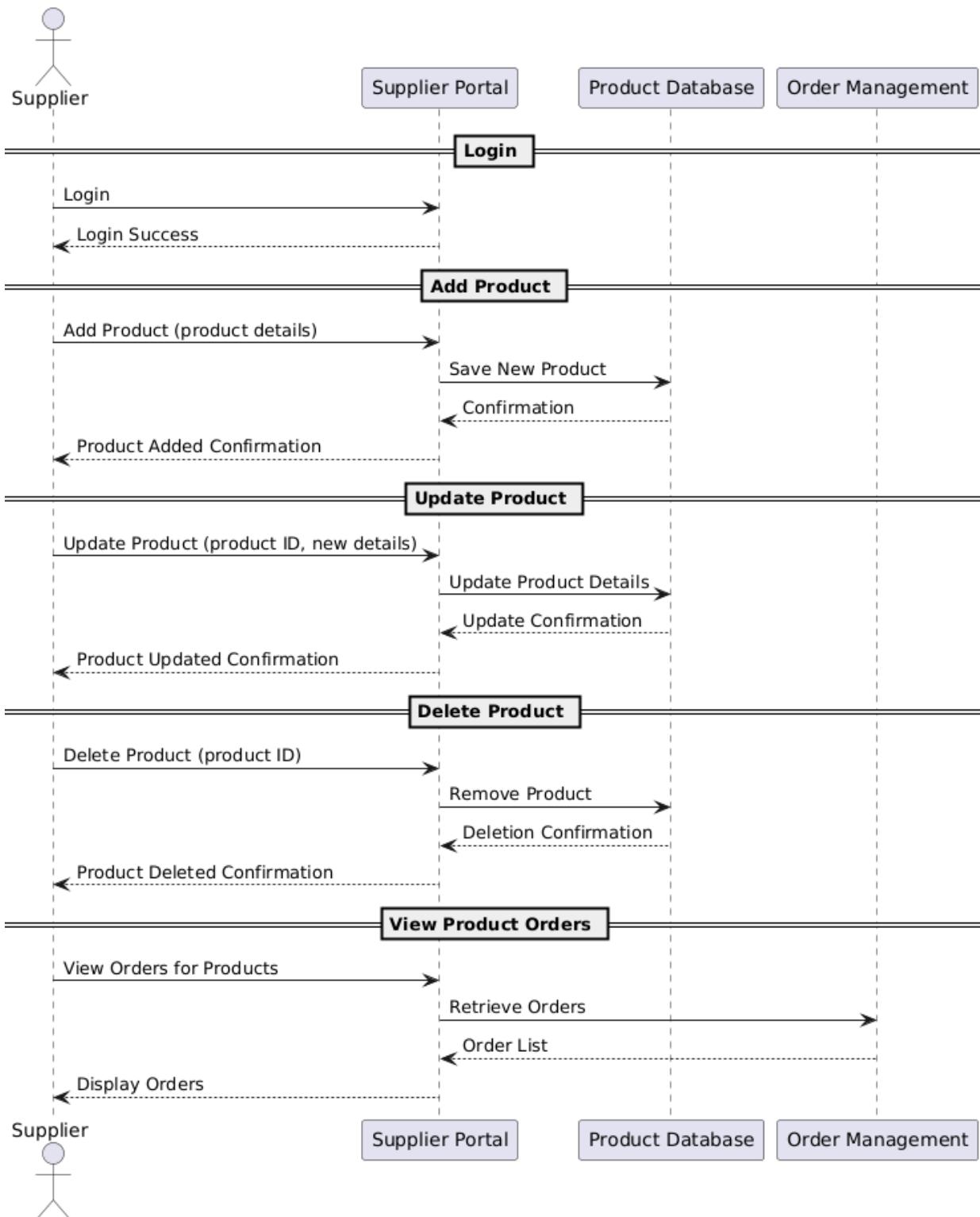
- Use case models

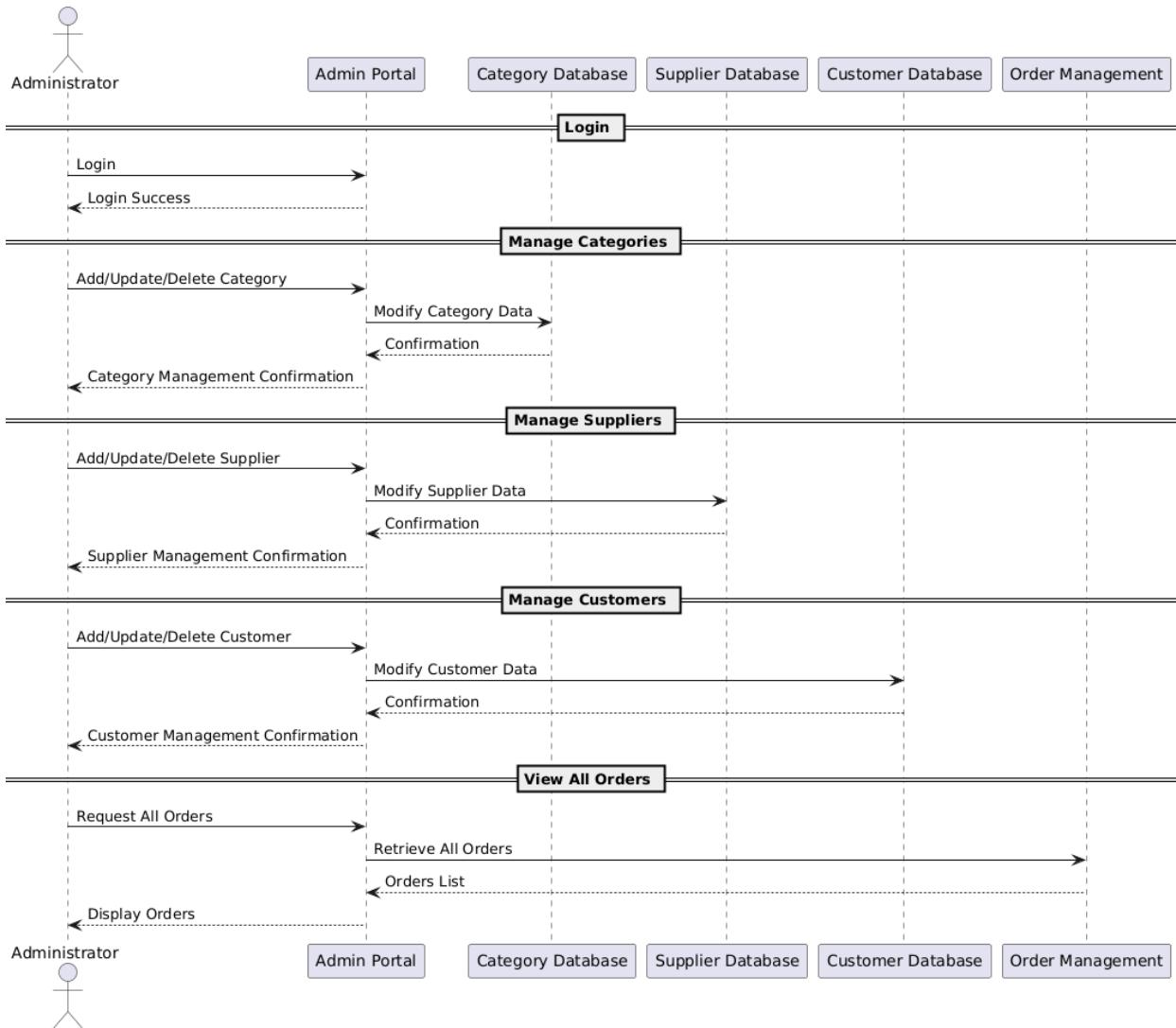




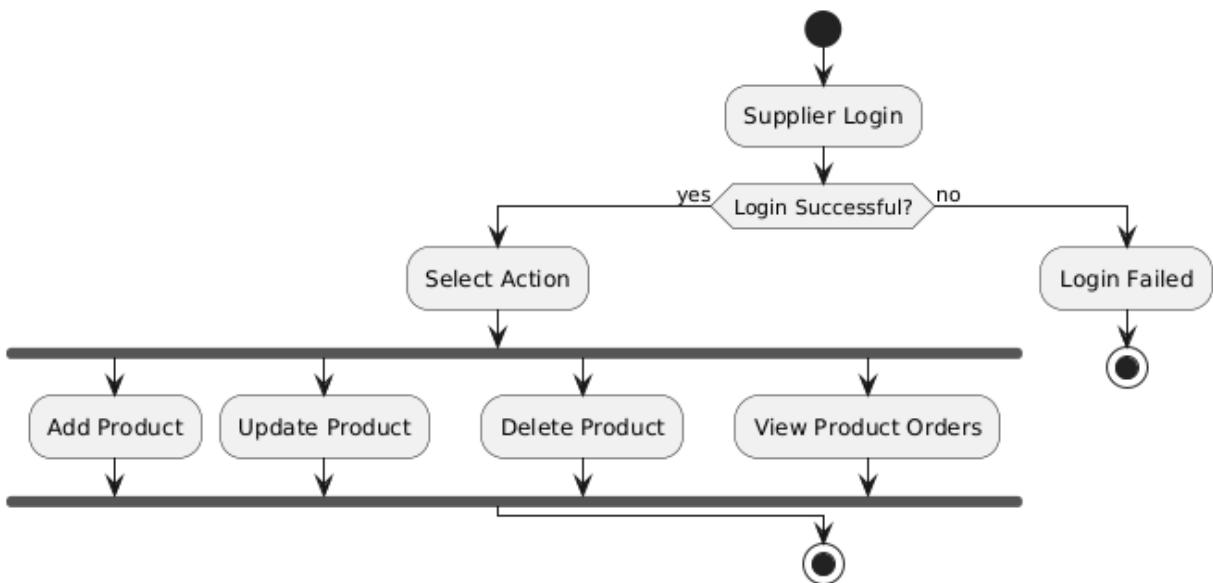
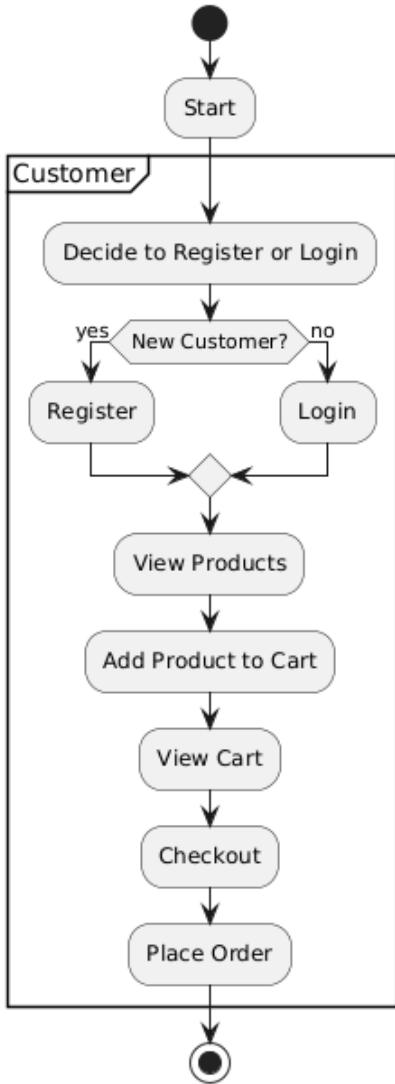
**Sequence diagrams:**

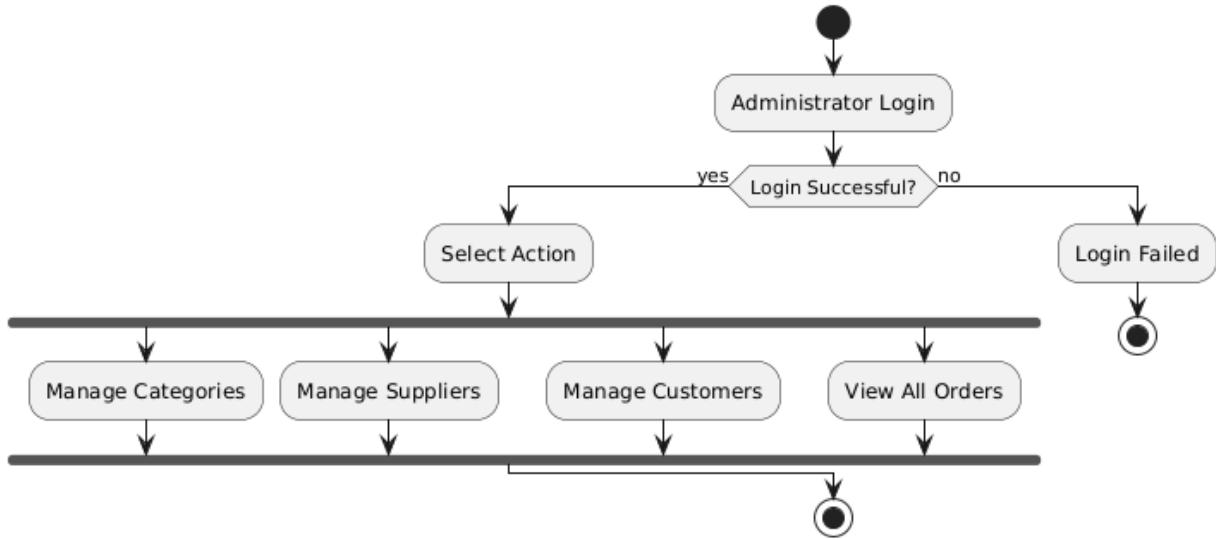




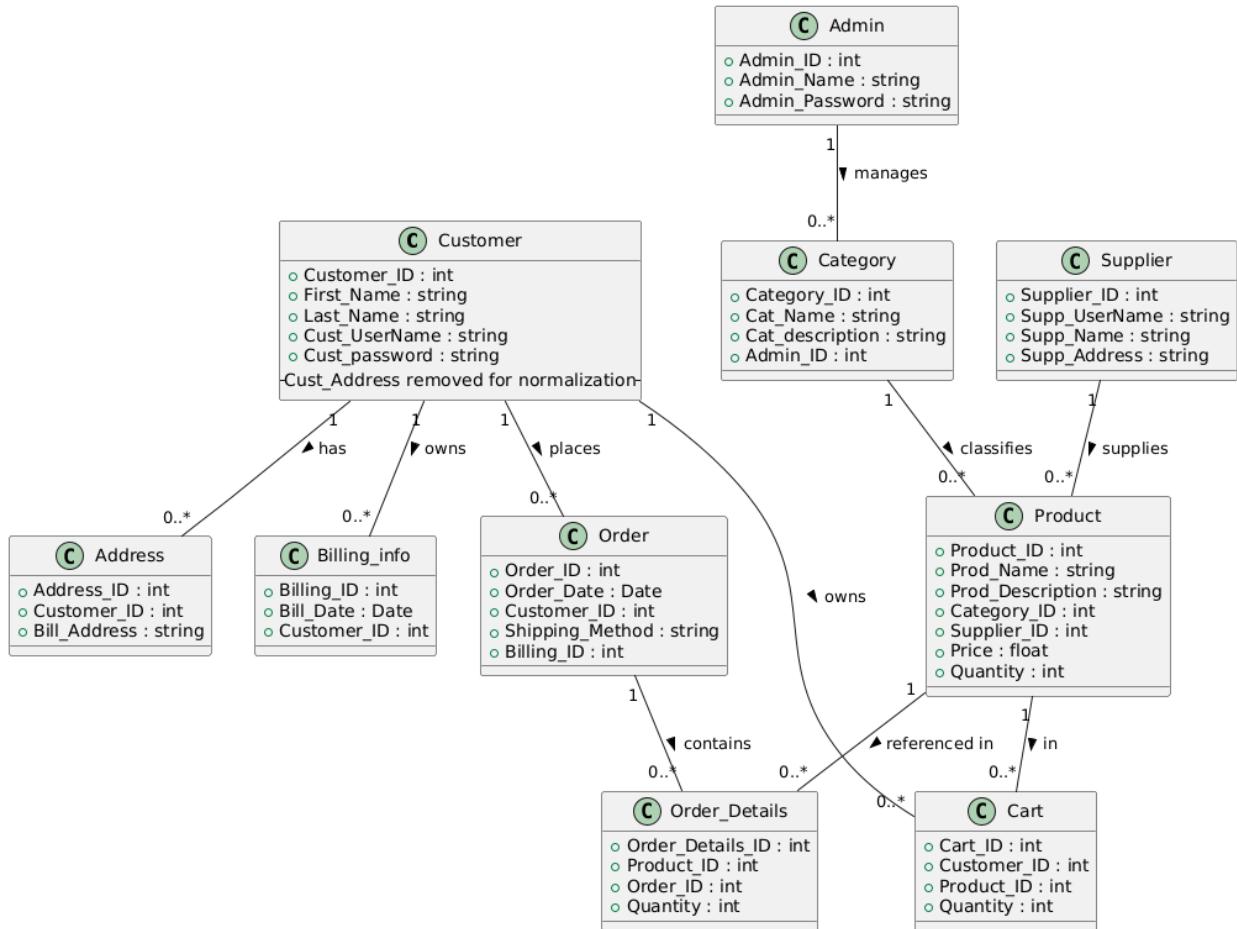


- **Activity diagrams**

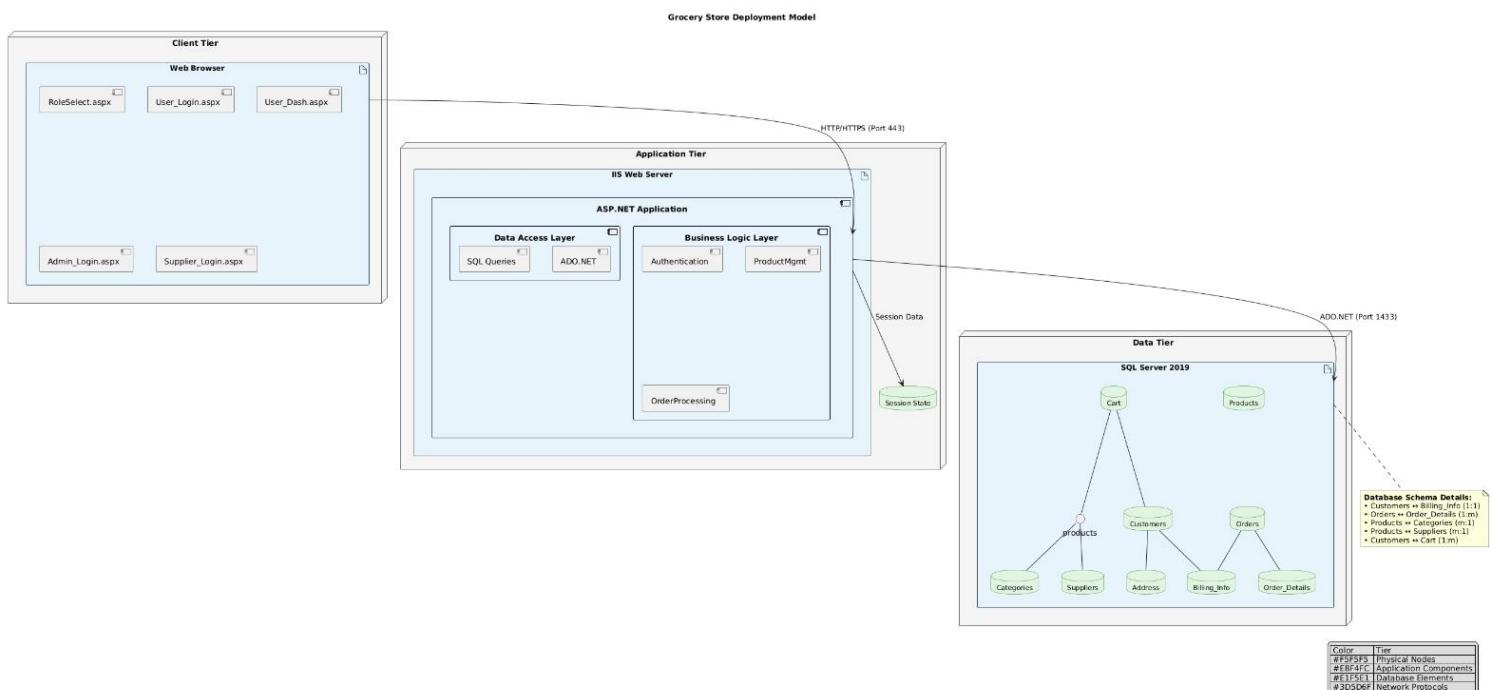
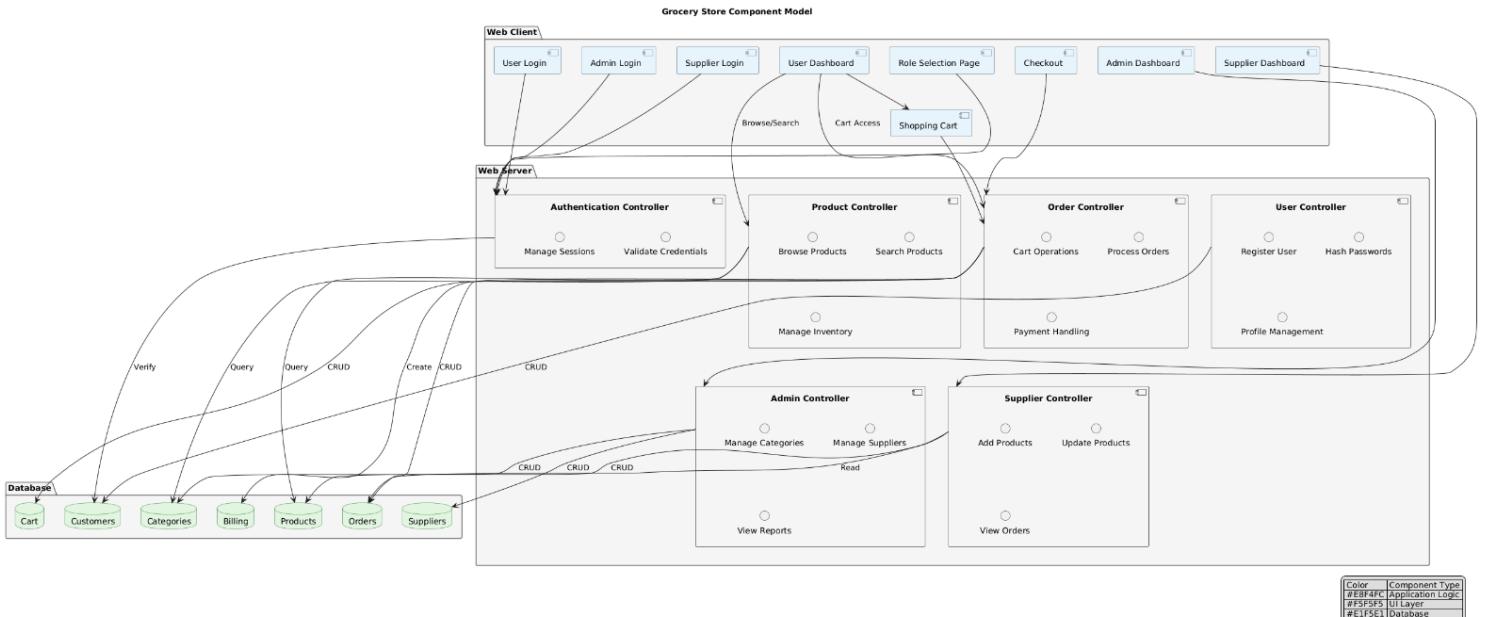




- Class diagram



## Component model and deployment model



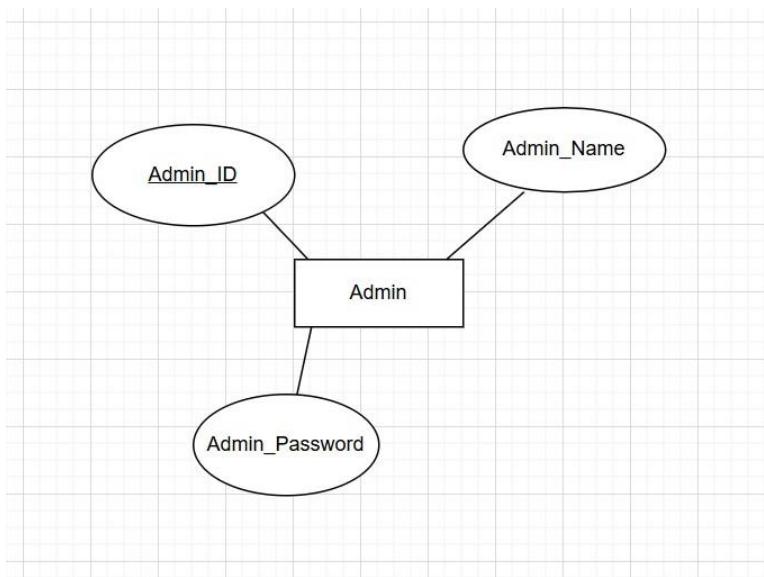
## ER Diagram Overview:

### Entities Name:

1. Customer
2. Billing\_info
3. Order
4. Order\_Details
5. Product
6. Category
7. Cart
8. Supplier
9. Admin

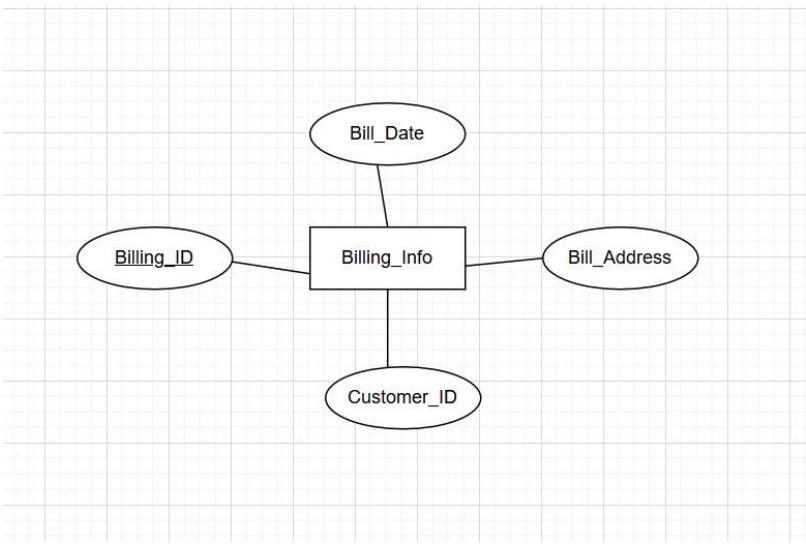
### Entities and its Attributes:

#### Admin:



PK Admin\_ID

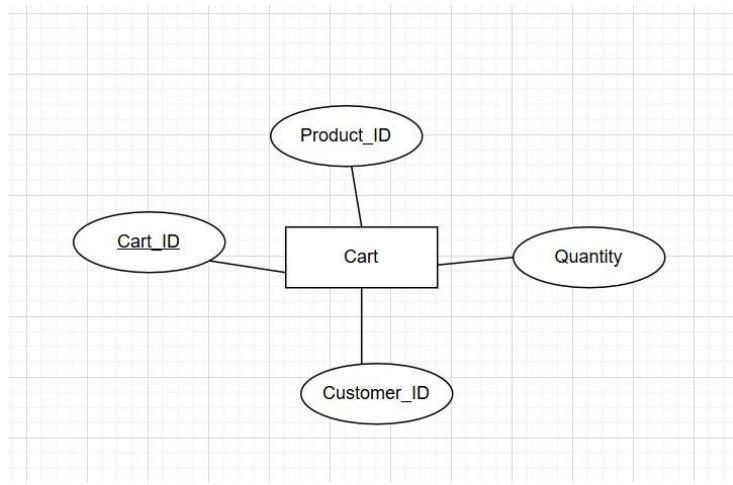
#### Billing\_Info:



PK **Billing\_ID**

FK Customer\_ID

### Cart:

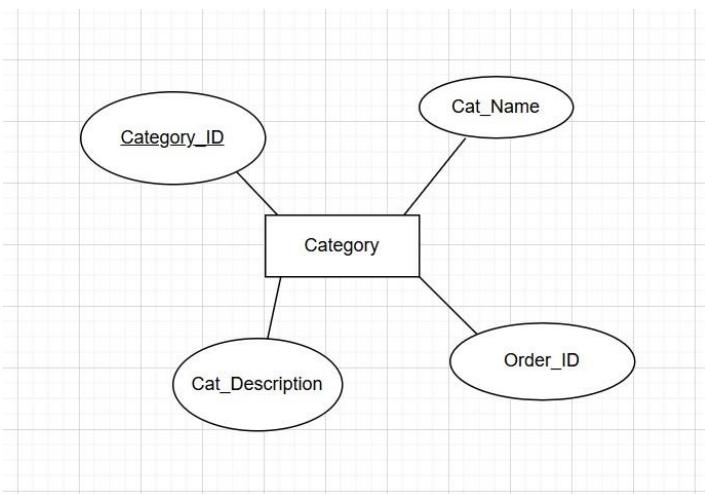


PK **Cart\_ID**

FK Customer\_ID

FK Product\_ID

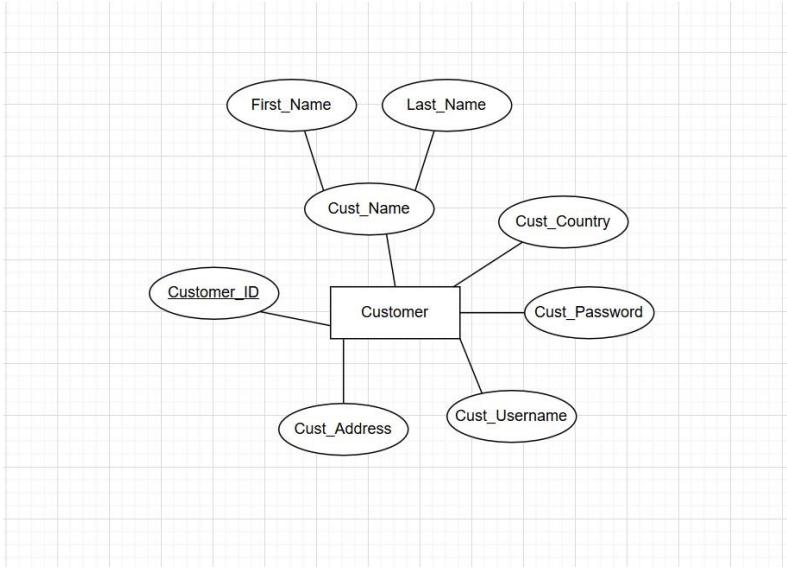
### Category:



PK **Category\_ID**

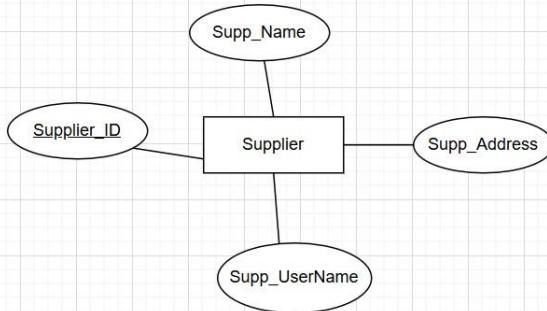
FK Order\_ID

### Customer:



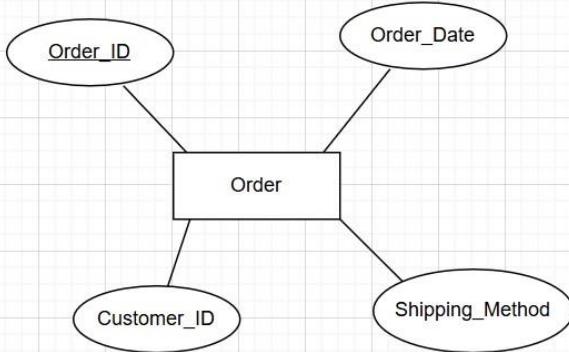
**PK Customer\_ID**

### Supplier:



**PK Supplier\_ID**

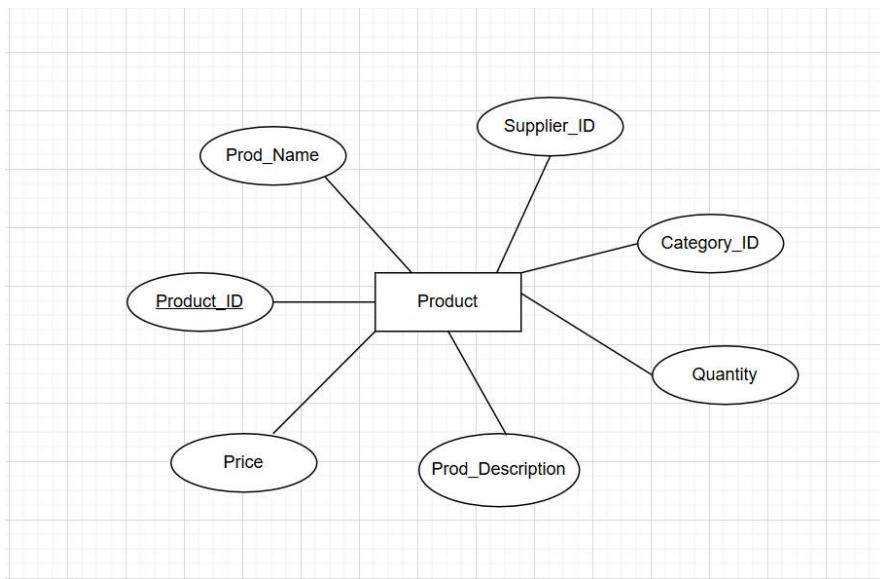
### Order:



**PK Order\_ID**

**FK Customer\_ID**

### Product:

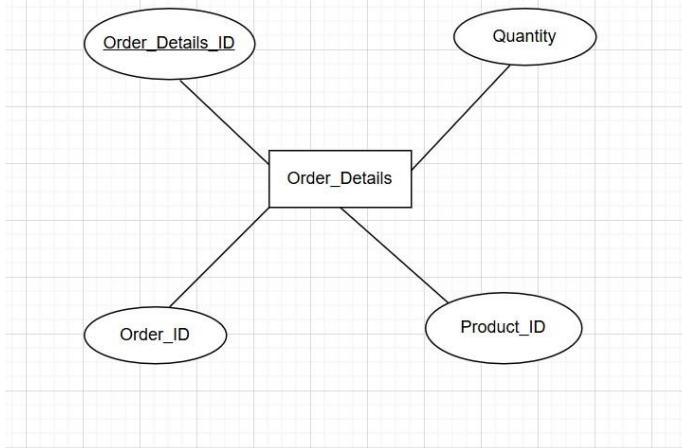


PK Product\_ID

FK Category\_ID

FK Supplier\_ID

### Order\_Details:



PK Order\_Details\_ID

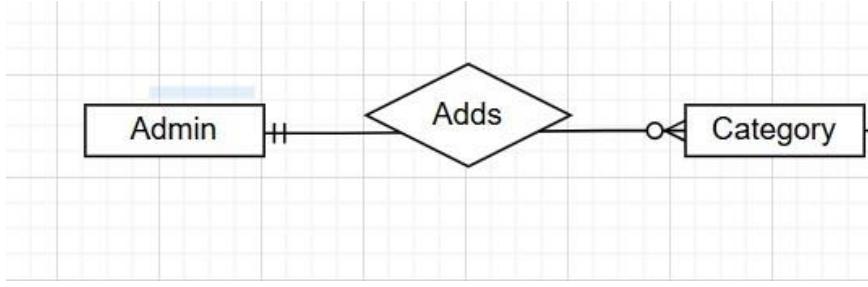
FK Product\_ID

FK Order\_ID

### Showing Cardinalities and Relationships:

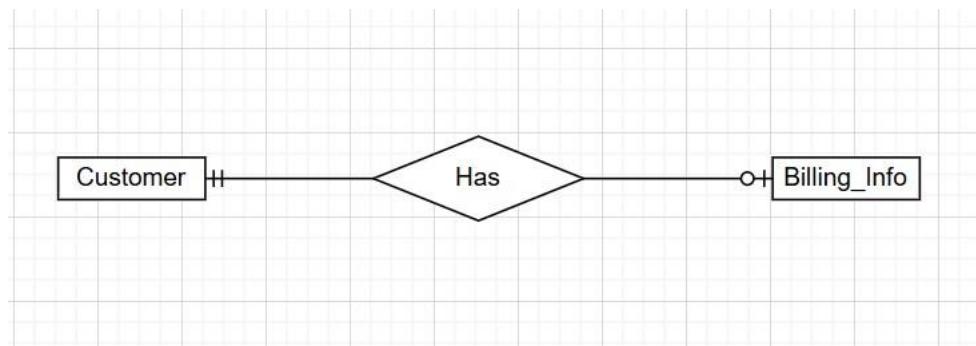
#### Admin → Category (One-to-Many)

- **Forward (Admin to Category):**
  - Min: **0** (An admin may not have added any categories).
  - Max: **Many** (An admin can add multiple categories).
- **Reverse (Category to Admin):**
  - Min: **1** (Each category must be added by an admin). ○ Max: **1** (A category is added by only one admin).



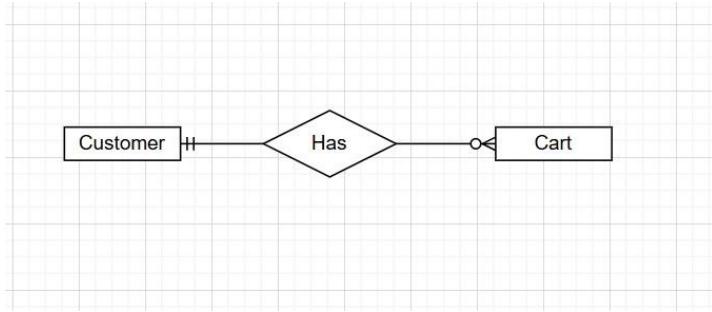
### **Customer → BillingInfo (One-to-One)**

- **Forward (Customer to BillingInfo):**
  - Min: **0** (A customer might not have billing information initially).
  - Max: **1** (A customer can have only one billing information).
- **Reverse (BillingInfo to Customer):**
  - Min: **1** (Billing information must be tied to a customer).
  - Max: **1** (Each billing information is associated with exactly one customer).



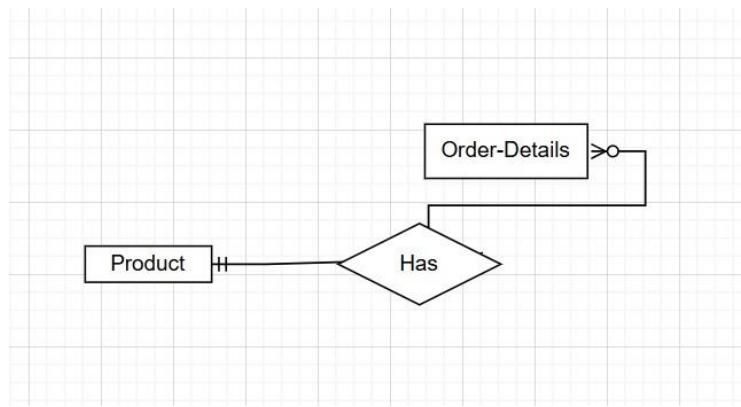
### **Customer → Cart (One-to-Many)**

- **Forward (Customer to Cart):**
  - Min: **0** (A customer may not have items in the cart).
  - Max: **Many** (A customer can have multiple carts).
- **Reverse (Cart to Customer):**
  - Min: **1** (Each cart must belong to one customer).
  - Max: **1** (Each cart belongs to only one customer).



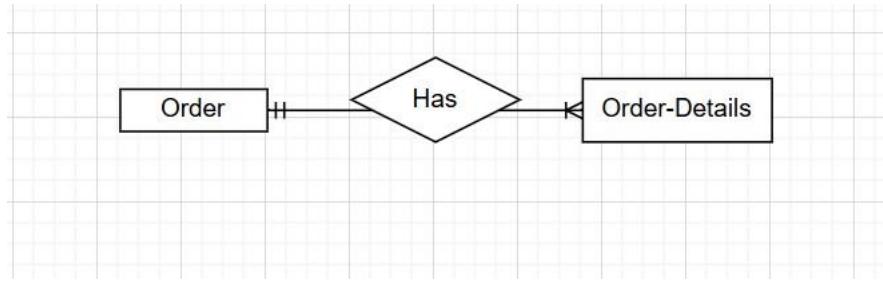
### **OrderDetails → Product (Many-to-One)**

- **Forward (OrderDetails to Product):**
  - Min: **1** (Each order detail must refer to one product).
  - Max: **1** (One order detail item points to only one product).
- **Reverse (Product to OrderDetails):**
  - Min: **0** (A product might not have been ordered yet).
  - Max: **Many** (A product can appear in multiple order details).



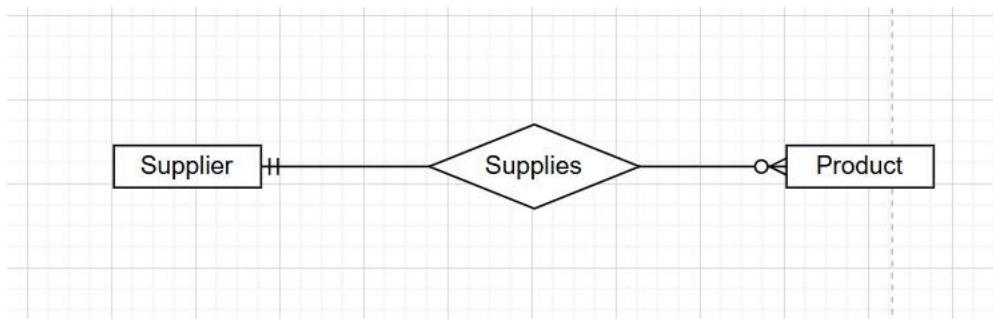
### **Order → Order\_Details (One-to-Many)**

- **Forward (Order to Order\_Details):**
  - Min: **1** (An order must have at least one product in order details).
  - Max: **Many** (An order can contain multiple products).
- **Reverse (Order\_Details to Order):**
  - Min: **1** (Order details must belong to one order).
  - Max: **1** (Each order detail item is tied to a single order).



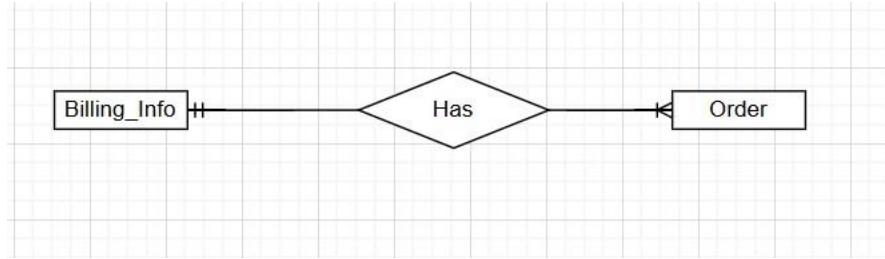
### **Product → Supplier (Many-to-One)**

- **Forward (Product to Supplier):**
  - Min: **1** (A product must have one supplier).
  - Max: **1** (A product can only be supplied by one supplier).
- **Reverse (Supplier to Product):**
  - Min: **0** (A supplier might not supply any products).
  - Max: **Many** (A supplier can supply multiple products).



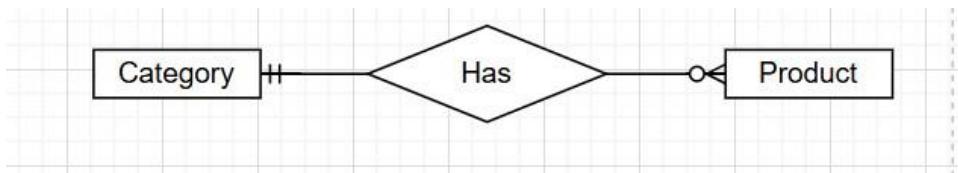
### **Billing\_Info → Order (One-to-Many)**

- **Forward (Billing\_Info to Order):**
  - Min: 1 (Each Billing\_Info must be associated with at least one Order).
  - Max: Many (One Billing\_Info can be associated with many Orders).
- **Reverse (Order to Billing\_Info):**
  - Min: 1 (Each Order must have one associated Billing\_Info).
  - Max: 1 (Each Order is associated with exactly one Billing\_Info).



### Product → Category (Many-to-One)

- **Forward (Product to Category):**
  - Min: **1** (Each product must belong to one category).
  - Max: **1** (A product belongs to only one category).
- **Reverse (Category to Product):**
  - Min: **0** (A category might not have products yet).
  - Max: **Many** (A category can have multiple products).

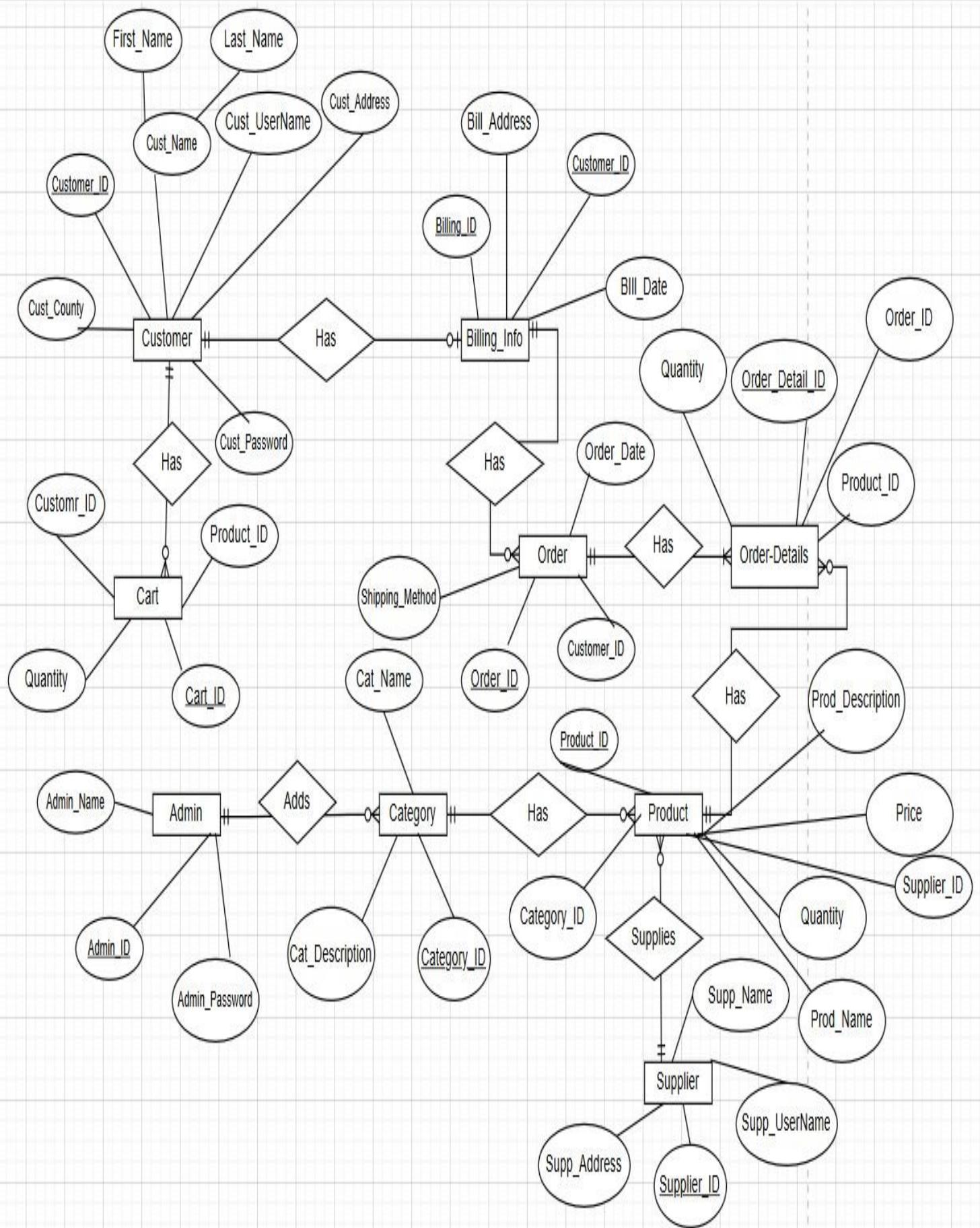


### Business Rules:

- An admin can add multiple categories but each category must be added by one admin.
- A customer might not have billing information initially but each customer can only have one billing information entry and each billing info must be linked to one and only one customer.
- A customer might not have any carts and also a customer can have multiple carts but each cart belongs to only one customer.
- Each order detail must refer to exactly one product but a product can appear in multiple order details.
- Every order must contain at least one order detail but each order detail is part of one order.
- Each product must be supplied by exactly one supplier but also a supplier can supply multiple products.

- Each billing info entry must be linked to at least one order but each order must have exactly one billing info.
- Each product must belong to one category but a category can have multiple products.

**Complete ER Diagram:**



## Relational Models Overview:

### Mapping:

#### Customer:

Customer_ID	First_Name	Last_Name	Cust_UserName	Cust_password	Cust_Address
-------------	------------	-----------	---------------	---------------	--------------

#### Billing\_info:

Billing_ID	Bill_Date	Bill_Adress	Customer_ID
------------	-----------	-------------	-------------

#### Order:

Order_ID	Order_Date	Customer_ID	Shipping_Method	Billing_ID
----------	------------	-------------	-----------------	------------

#### Order\_Details:

Order_Details_ID	Product_ID	Order_ID	Quantity
------------------	------------	----------	----------

#### Product:

Product_ID	Prod_Name	Prod_Description	Category_ID	Supplier_ID	Price	Quantity
------------	-----------	------------------	-------------	-------------	-------	----------

#### Category:

Category_ID	Cat_Name	Cat_description	Admin_ID
-------------	----------	-----------------	----------

#### Admin:

Admin_ID	Admin_Name	Admin_Password
----------	------------	----------------

**Supplier:**

Supplier_ID	Supp_UserName	Supp_Name	Supp_Address
-------------	---------------	-----------	--------------

**Product:**

Product_ID	Prod_Name	Prod_Description	Category_ID	Supplier_ID	Price	Quantity
------------	-----------	------------------	-------------	-------------	-------	----------

**Customer:**

Customer_ID	First_Name	Last_Name	Cust_UserName	Cust_password	Cust_Address
-------------	------------	-----------	---------------	---------------	--------------

**Cart**

Cart_ID	Customer_ID	Product_ID	Quantity
---------	-------------	------------	----------

**Normalization:****1 Normal-Form:**

Each column contains only atomic values.

**Customer:**

Customer_ID	First_Name	Last_Name	Cust_UserName	Cust_password	Cust_Address
-------------	------------	-----------	---------------	---------------	--------------

**Billing\_info:**

Billing_ID	Bill_Date	Bill_Adress	Customer_ID
------------	-----------	-------------	-------------

**Order:**

Order_ID	Order_Date	Customer_ID	Shipping_Method	Billing_ID
----------	------------	-------------	-----------------	------------

**Order\_Details:**

Order_Details_ID	Product_ID	Order_ID	Quantity
------------------	------------	----------	----------

**Product:**

Product_ID	Prod_Name	Prod_Description	Category_ID	Supplier_ID	Price	Quantity
------------	-----------	------------------	-------------	-------------	-------	----------

**Category:**

Category_ID	Cat_Name	Cat_description	Admin_ID
-------------	----------	-----------------	----------

**Admin:**

Admin_ID	Admin_Name	Admin_Password
----------	------------	----------------

**Supplier:**

Supplier_ID	Supp_UserName	Supp_Name	Supp_Address
-------------	---------------	-----------	--------------

**Product:**

Product_ID	Prod_Name	Prod_Description	Category_ID	Supplier_ID	Price	Quantity
------------	-----------	------------------	-------------	-------------	-------	----------

**Customer:**

Customer_ID	First_Name	Last_Name	Cust_UserName	Cust_password	Cust_Address
-------------	------------	-----------	---------------	---------------	--------------

**Cart**

Cart_ID	Customer_ID	Product_ID	Quantity
---------	-------------	------------	----------

**2 Normal-Form:**

It is in 1NF.

It has no partial dependency.

**Customer:**

Customer_ID	First_Name	Last_Name	Cust_UserName	Cust_password	Cust_Address
-------------	------------	-----------	---------------	---------------	--------------

**Billing\_info:**

Billing_ID	Bill_Date	Bill_Adress	Customer_ID
------------	-----------	-------------	-------------

**Order:**

Order_ID	Order_Date	Customer_ID	Shipping_Method	Billing_ID
----------	------------	-------------	-----------------	------------

**Order\_Details:**

Order_Details_ID	Product_ID	Order_ID	Quantity
------------------	------------	----------	----------

**Product:**

Product_ID	Prod_Name	Prod_Description	Category_ID	Supplier_ID	Price	Quantity
------------	-----------	------------------	-------------	-------------	-------	----------

**Category:**

Category_ID	Cat_Name	Cat_description	Admin_ID
-------------	----------	-----------------	----------

**Admin:**

Admin_ID	Admin_Name	Admin_Password
----------	------------	----------------

**Supplier:**

Supplier_ID	Supp_UserName	Supp_Name	Supp_Address
-------------	---------------	-----------	--------------

**Product:**

Product_ID	Prod_Name	Prod_Description	Category_ID	Supplier_ID	Price	Quantity
------------	-----------	------------------	-------------	-------------	-------	----------

**Customer:**

Customer_ID	First_Name	Last_Name	Cust_UserName	Cust_password	Cust_Address
-------------	------------	-----------	---------------	---------------	--------------

**Cart**

Cart_ID	Customer_ID	Product_ID	Quantity
---------	-------------	------------	----------

**3 Normal-Form:**

It is in 2NF.

It has no transitive dependency.

**Customer:**

Customer_ID	First_Name	Last_Name	Cust_UserName	Cust_password	Cust_Address
-------------	------------	-----------	---------------	---------------	--------------

**Billing\_info:**

Transitive Dependency: The Bill\_Address is a non-prime attribute and may depend on the CustomerID indirectly. To remove this, we should separate the address into a different table.

Solution: Create a separate table for Address and link it with CustomerID.

Billing_ID	Bill_Date	Customer_ID
------------	-----------	-------------

**Address:**

Address_ID	Customer_ID	Bill_Address
------------	-------------	--------------

**Order:**

Order_ID	Order_Date	Customer_ID	Shipping_Method	Billing_ID
----------	------------	-------------	-----------------	------------

**Order\_Details:**

Order_Details_ID	Product_ID	Order_ID	Quantity
------------------	------------	----------	----------

**Product:**

Product_ID	Prod_Name	Prod_Description	Category_ID	Supplier_ID	Price	Quantity
------------	-----------	------------------	-------------	-------------	-------	----------

**Category:**

Category_ID	Cat_Name	Cat_description	Admin_ID
-------------	----------	-----------------	----------

**Admin:**

Admin_ID	Admin_Name	Admin_Password
----------	------------	----------------

**Supplier:**

Supplier_ID	Supp_UserName	Supp_Name	Supp_Address
-------------	---------------	-----------	--------------

**Product:**

Product_ID	Prod_Name	Prod_Description	Category_ID	Supplier_ID	Price	Quantity
------------	-----------	------------------	-------------	-------------	-------	----------

**Customer:**

Customer_ID	First_Name	Last_Name	Cust_UserName	Cust_password	Cust_Address
-------------	------------	-----------	---------------	---------------	--------------

**Cart**

Cart_ID	Customer_ID	Product_ID	Quantity
---------	-------------	------------	----------

**Create Database On Oracle Overview:**

Creating  
New  
Database  
User:

User: SYSTEM

Home > Administration > Manage Database Users > **Create Database User**

Create Database User

\* Username:   
\* Password:   
\* Confirm Password:   
Expire Password   
Account Status:

Default Tablespace: **USERS**  
Temporary Tablespace: **TEMP**

User Privileges

Roles:  
 CONNECT  RESOURCE  DBA

Direct Grant System Privileges:  
 CREATE DATABASE LINK  CREATE MATERIALIZED VIEW  CREATE PROCEDURE  
 CREATE PUBLIC SYNONYM  CREATE ROLE  CREATE SEQUENCE  
 CREATE SYNONYM  CREATE TABLE  CREATE TRIGGER  
 CREATE TYPE  CREATE VIEW

[Check All](#) [Uncheck All](#)

Login:

Creating Tables:

**Database Login**

Enter your database username and password.

Username:   
Password:

[Click here to learn how to get started](#)

User: GROCERY\_STORE

Home > SQL > SQL Commands

Autocommit Display 10 ▾

```
FOREIGN KEY (Customer_ID) REFERENCES CUSTOMER(Customer_ID),
FOREIGN KEY (Billing_ID) REFERENCES Billing_Info(Billing_ID)
);

CREATE TABLE Order_Details (
    Order_Details_ID NUMBER PRIMARY KEY,
    Product_ID NUMBER,
    Order_ID NUMBER,
    Quantity NUMBER NOT NULL,
    FOREIGN KEY (Product_ID) REFERENCES Product(Product_ID),
    FOREIGN KEY (Order_ID) REFERENCES Customer_Order(Order_ID)
);
```

Results Explain Describe Saved SQL History

ADDRESS
ADMIN
BILLING_INFO
CART
CATEGORY
CUSTOMER
CUSTOMER_ORDER
ORDER_DETAILS
PRODUCT
SUPPLIER

Table created.

0.03 seconds

Inserting Sample Data:

```

2 VALUES (24, 'Victor', 'Morris', 'Victorm', 'Victor001', '2105 Oak Court');
1 row created.

SQL>
SQL> INSERT INTO Customer (Customer_ID, First_Name, Last_Name, Cust_UserName, Cust_Password, Cust_Address)
  2 VALUES (25, 'Wendy', 'Bell', 'wendyb', 'wendy456', '2204 Pine Park');

1 row created.

SQL>
SQL> INSERT INTO Customer (Customer_ID, First_Name, Last_Name, Cust_UserName, Cust_Password, Cust_Address)
  2 VALUES (26, 'Xander', 'Lee', 'xander1', 'xander999', '2305 Maple Meadow');

1 row created.

SQL>
SQL> INSERT INTO Customer (Customer_ID, First_Name, Last_Name, Cust_UserName, Cust_Password, Cust_Address)
  2 VALUES (27, 'Yara', 'Hill', 'yarah', 'yarapass', '2406 Birch Grove');

1 row created.

SQL>
SQL> INSERT INTO Customer (Customer_ID, First_Name, Last_Name, Cust_UserName, Cust_Password, Cust_Address)
  2 VALUES (28, 'Zane', 'Ward', 'zanew', 'zane123', '2507 Walnut Circle');

1 row created.

SQL>
SQL> INSERT INTO Customer (Customer_ID, First_Name, Last_Name, Cust_UserName, Cust_Password, Cust_Address)
  2 VALUES (29, 'Adam', 'Knight', 'adamk', 'adamsecure', '2608 Cherry Woods');

1 row created.

SQL>
SQL> INSERT INTO Customer (Customer_ID, First_Name, Last_Name, Cust_UserName, Cust_Password, Cust_Address)
  2 VALUES (30, 'Bella', 'Pierce', 'bellap', 'bella123', '2709 Elm Haven');

1 row created.

SQL>
2 VALUES (24, TO_DATE('2024-01-24', 'YYYY-MM-DD'), 24);

1 row created.

SQL>
SQL> INSERT INTO Billing_info (Billing_ID, Bill_Date, Customer_ID)
  2 VALUES (25, TO_DATE('2024-01-25', 'YYYY-MM-DD'), 25);

1 row created.

SQL>
SQL> INSERT INTO Billing_info (Billing_ID, Bill_Date, Customer_ID)
  2 VALUES (26, TO_DATE('2024-01-26', 'YYYY-MM-DD'), 26);

1 row created.

SQL>
SQL> INSERT INTO Billing_info (Billing_ID, Bill_Date, Customer_ID)
  2 VALUES (27, TO_DATE('2024-01-27', 'YYYY-MM-DD'), 27);

1 row created.

SQL>
SQL> INSERT INTO Billing_info (Billing_ID, Bill_Date, Customer_ID)
  2 VALUES (28, TO_DATE('2024-01-28', 'YYYY-MM-DD'), 28);

1 row created.

SQL>
SQL> INSERT INTO Billing_info (Billing_ID, Bill_Date, Customer_ID)
  2 VALUES (29, TO_DATE('2024-01-29', 'YYYY-MM-DD'), 29);

1 row created.

SQL>
SQL> INSERT INTO Billing_info (Billing_ID, Bill_Date, Customer_ID)
  2 VALUES (30, TO_DATE('2024-01-30', 'YYYY-MM-DD'), 30);

1 row created.

```

```
2 VALUES (24, TO_DATE('2024-02-24', 'YYYY-MM-DD'), 24, 'Express', 24);
1 row created.

SQL>
SQL> INSERT INTO Customer_Order (Order_ID, Order_Date, Customer_ID, Shipping_Method, Billing_ID)
  2 VALUES (25, TO_DATE('2024-02-25', 'YYYY-MM-DD'), 25, 'Standard', 25);

1 row created.

SQL>
SQL> INSERT INTO Customer_Order (Order_ID, Order_Date, Customer_ID, Shipping_Method, Billing_ID)
  2 VALUES (26, TO_DATE('2024-02-26', 'YYYY-MM-DD'), 26, 'Express', 26);

1 row created.

SQL>
SQL> INSERT INTO Customer_Order (Order_ID, Order_Date, Customer_ID, Shipping_Method, Billing_ID)
  2 VALUES (27, TO_DATE('2024-02-27', 'YYYY-MM-DD'), 27, 'Standard', 27);

1 row created.

SQL>
SQL> INSERT INTO Customer_Order (Order_ID, Order_Date, Customer_ID, Shipping_Method, Billing_ID)
  2 VALUES (28, TO_DATE('2024-02-28', 'YYYY-MM-DD'), 28, 'Express', 28);

1 row created.

SQL>
SQL> INSERT INTO Customer_Order (Order_ID, Order_Date, Customer_ID, Shipping_Method, Billing_ID)
  2 VALUES (29, TO_DATE('2024-02-29', 'YYYY-MM-DD'), 29, 'Standard', 29);

1 row created.

SQL>
SQL> INSERT INTO Customer_Order (Order_ID, Order_Date, Customer_ID, Shipping_Method, Billing_ID)
  2 VALUES (30, TO_DATE('2024-03-01', 'YYYY-MM-DD'), 30, 'Express', 30);
```

```
2 VALUES (2, 'Wearables', 'Smartwatches, fitness trackers, and other wearable devices.');
1 row created.

SQL>
SQL> INSERT INTO Category (Category_ID, Cat_Name, Cat_Description)
  2 VALUES (3, 'Accessories', 'Phone cases, chargers, screen protectors, and other accessories.');

1 row created.

SQL>
SQL> INSERT INTO Category (Category_ID, Cat_Name, Cat_Description)
  2 VALUES (4, 'Gaming', 'Products related to gaming including consoles, games, and accessories.');

1 row created.

SQL>
SQL> INSERT INTO Category (Category_ID, Cat_Name, Cat_Description)
  2 VALUES (5, 'Home Appliances', 'Appliances for home use like vacuum cleaners, air purifiers, etc.');

1 row created.
```

```
SQL> INSERT INTO Supplier (Supplier_ID, Supp_UserName, Supp_Name, Supp_Address)
  2  VALUES (6, 'appliancehub', 'Appliance Hub LLC', '303 Appliance Pkwy, Chicago, IL');

1 row created.

SQL>
SQL> INSERT INTO Supplier (Supplier_ID, Supp_UserName, Supp_Name, Supp_Address)
  2  VALUES (7, 'gaminggear', 'Gaming Gear Supplies', '404 Gaming Ln, Las Vegas, NV');

1 row created.

SQL>
SQL> INSERT INTO Supplier (Supplier_ID, Supp_UserName, Supp_Name, Supp_Address)
  2  VALUES (8, 'phototech', 'Photo Tech Supplies', '505 Photo Dr, Denver, CO');

1 row created.

SQL>
SQL> INSERT INTO Supplier (Supplier_ID, Supp_UserName, Supp_Name, Supp_Address)
  2  VALUES (9, 'homeplus', 'Home Plus Electronics', '606 Home Blvd, San Francisco, CA');

1 row created.

SQL>
SQL> INSERT INTO Supplier (Supplier_ID, Supp_UserName, Supp_Name, Supp_Address)
  2  VALUES (10, 'healthtech', 'Health Tech Solutions', '707 Health St, Boston, MA');
```

```
1 row created.

SQL>
SQL> INSERT INTO Order_Details (Order_Details_ID, Product_ID, Order_ID, Quantity)
  2  VALUES (26, 26, 13, 6);

1 row created.

SQL>
SQL> INSERT INTO Order_Details (Order_Details_ID, Product_ID, Order_ID, Quantity)
  2  VALUES (27, 27, 14, 2);

1 row created.

SQL>
SQL> INSERT INTO Order_Details (Order_Details_ID, Product_ID, Order_ID, Quantity)
  2  VALUES (28, 28, 14, 1);

1 row created.

SQL>
SQL> INSERT INTO Order_Details (Order_Details_ID, Product_ID, Order_ID, Quantity)
  2  VALUES (29, 29, 15, 5);

1 row created.

SQL>
SQL> INSERT INTO Order_Details (Order_Details_ID, Product_ID, Order_ID, Quantity)
  2  VALUES (30, 30, 15, 3);

1 row created.
```

```
1 row created.

SQL>
SQL> INSERT INTO Cart (Cart_ID, Customer_ID, Product_ID, Quantity)
  2  VALUES (25, 13, 25, 1);

1 row created.

SQL>
SQL> INSERT INTO Cart (Cart_ID, Customer_ID, Product_ID, Quantity)
  2  VALUES (26, 13, 26, 1);

1 row created.

SQL>
SQL> INSERT INTO Cart (Cart_ID, Customer_ID, Product_ID, Quantity)
  2  VALUES (27, 14, 27, 2);

1 row created.

SQL>
SQL> INSERT INTO Cart (Cart_ID, Customer_ID, Product_ID, Quantity)
  2  VALUES (28, 14, 28, 1);

1 row created.

SQL>
SQL> INSERT INTO Cart (Cart_ID, Customer_ID, Product_ID, Quantity)
  2  VALUES (29, 15, 29, 1);

1 row created.

SQL>
SQL> INSERT INTO Cart (Cart_ID, Customer_ID, Product_ID, Quantity)
  2  VALUES (30, 15, 30, 2);

1 row created.
```

```
SQL> INSERT INTO Admin (Admin_ID, Admin_Name, Admin_Password)
  2  VALUES (1, 'AdminName', 'AdminPassword');

1 row created.

SQL>
```

```
1 row created.

SQL>
SQL> INSERT INTO Product (Product_ID, Prod_Name, Prod_Description, Category_ID, Supplier_ID, Price, Quantity)
  2  VALUES (26, 'Router Extender', 'Wi-Fi extender for boosting signal', 3, 6, 49.99, 250);

1 row created.

SQL>
SQL> INSERT INTO Product (Product_ID, Prod_Name, Prod_Description, Category_ID, Supplier_ID, Price, Quantity)
  2  VALUES (27, 'Ethernet Cable', 'High-speed 10ft Ethernet cable', 3, 7, 9.99, 350);

1 row created.

SQL>
SQL> INSERT INTO Product (Product_ID, Prod_Name, Prod_Description, Category_ID, Supplier_ID, Price, Quantity)
  2  VALUES (28, 'USB Flash Drive', '64GB USB 3.0 flash drive', 3, 8, 15.99, 500);

1 row created.

SQL>
SQL> INSERT INTO Product (Product_ID, Prod_Name, Prod_Description, Category_ID, Supplier_ID, Price, Quantity)
  2  VALUES (29, 'Bluetooth Earbuds', 'True wireless earbuds with noise-canceling', 3, 9, 99.99, 60);

1 row created.

SQL>
SQL> INSERT INTO Product (Product_ID, Prod_Name, Prod_Description, Category_ID, Supplier_ID, Price, Quantity)
  2  VALUES (30, 'Webcam', '1080p HD webcam for streaming', 3, 10, 49.99, 200);
```

```
1 row created.

SQL>
SQL> INSERT INTO Address (Address_ID, Customer_ID, Bill_Address)
  2  VALUES (25, 25, '2204 Pine Park');

1 row created.

SQL>
SQL> INSERT INTO Address (Address_ID, Customer_ID, Bill_Address)
  2  VALUES (26, 26, '2305 Maple Meadow');

1 row created.

SQL>
SQL> INSERT INTO Address (Address_ID, Customer_ID, Bill_Address)
  2  VALUES (27, 27, '2406 Birch Grove');

1 row created.

SQL>
SQL> INSERT INTO Address (Address_ID, Customer_ID, Bill_Address)
  2  VALUES (28, 28, '2507 Walnut Circle');

1 row created.

SQL>
SQL> INSERT INTO Address (Address_ID, Customer_ID, Bill_Address)
  2  VALUES (29, 29, '2608 Cherry Woods');

1 row created.

SQL>
SQL> INSERT INTO Address (Address_ID, Customer_ID, Bill_Address)
  2  VALUES (30, 30, '2709 Elm Haven');

1 row created.
```

```
SQL> commit;  
Commit complete.
```

Checking Inserted Sample Data:

```
SQL> select *from Billing_Info;  
  
BILLING_ID BILL_DATE CUSTOMER_ID  
===== ====== =====  
1 01-JAN-24 1  
2 02-JAN-24 2  
3 03-JAN-24 3  
4 04-JAN-24 4  
5 05-JAN-24 5  
6 06-JAN-24 6  
7 07-JAN-24 7  
8 08-JAN-24 8  
9 09-JAN-24 9  
10 10-JAN-24 10  
11 11-JAN-24 11  
12 12-JAN-24 12  
13 13-JAN-24 13  
14 14-JAN-24 14  
15 15-JAN-24 15  
16 16-JAN-24 16  
17 17-JAN-24 17  
18 18-JAN-24 18  
19 19-JAN-24 19  
20 20-JAN-24 20  
21 21-JAN-24 21  
22 22-JAN-24 22  
23 23-JAN-24 23  
24 24-JAN-24 24  
25 25-JAN-24 25  
26 26-JAN-24 26  
27 27-JAN-24 27  
28 28-JAN-24 28  
29 29-JAN-24 29  
30 30-JAN-24 30
```

```
SUPPLIER_ID SUPP_USERNAME SUPP_NAME SUPP_ADDRESS  
===== ====== ===== ======  
1 techsup1 Tech Supplies Inc. 123 Tech Street, Silicon Valley, CA  
2 gadgetguru Gadget Guru Ltd. 456 Gadget Blvd, New York, NY  
3 wearableworld Wearable World Co. 789 Wearable Ave, Los Angeles, CA  
4 soundwave Sound Wave Electronics 101 Sound Rd, Austin, TX  
5 mobilemax Mobile Max Ltd. 202 Mobile St, Miami, FL  
6 appliancehub Appliance Hub LLC 303 Appliance Pkwy, Chicago, IL  
7 gaminggear Gaming Gear Supplies 404 Gaming Ln, Las Vegas, NV  
8 phototech Photo Tech Supplies 505 Photo Dr, Denver, CO  
9 homeplus Home Plus Electronics 606 Home Blvd, San Francisco, CA  
10 healthtech Health Tech Solutions 707 Health St, Boston, MA
```

10 rows selected.

PRODUCT_ID	PROD_NAME	PROD_DESCRIPTION	CATEGORY_ID	SUPPLIER_ID	PRICE	QUANTITY
1	Laptop	High-performance laptop with 16GB RAM	1	1	1200.99	50
2	Smartphone	Latest model with 5G support	2	2	799.99	100
3	Tablet	10-inch tablet with 128GB storage	1	3	499.99	75
4	Headphones	Noise-cancelling over-ear headphones	3	4	199.99	200
5	Smartwatch	Waterproof smartwatch with fitness tracking	2	5	299.99	150
6	Keyboard	Mechanical keyboard with RGB lighting	3	6	79.99	120
7	Mouse	Wireless mouse with ergonomic design	3	7	29.99	250
8	Monitor	27-inch 4K monitor	1	8	499.99	60
9	Charger	Fast-charging wall adapter	3	9	29.99	500
10	Speaker	Bluetooth portable speaker	2	10	99.99	80
11	Camera	Digital camera with 20MP resolution	1	1	450.00	45
12	Printer	Wireless inkjet printer	3	2	129.99	130
13	Projector	Portable mini projector for home theater	1	3	249.99	70
14	Camera Lens	50mm lens for digital cameras	1	4	199.99	90
15	Tripod	Adjustable tripod for photography	1	5	49.99	200
16	Drone	Quadcopter with 4K camera	1	6	599.99	40
17	Smart Lock	Electronic lock with smartphone control	2	7	150.00	180
18	Fitness Tracker	Wearable fitness tracker with heart rate monitor	2	8	99.99	220
19	Virtual Reality Headset	Immersive VR headset for gaming	2	9	399.99	55
20	Smart Glasses	Glasses with augmented reality display	2	10	299.99	110
21	Smartphone Case	Protective case for smartphones	3	1	19.99	400
22	Screen Protector	Tempered glass screen protector for smartphones	3	2	9.99	500
23	Power Bank	Portable power bank with 10000mAh capacity	3	3	29.99	150
24	External Hard Drive	1TB external hard drive for data storage	3	4	69.99	95
25	Wi-Fi Router	Dual-band Wi-Fi router for home network	3	5	59.99	200
26	Router Extender	Wi-Fi extender for boosting signal	3	6	49.99	250
27	Ethernet Cable	High-speed 10ft Ethernet cable	3	7	9.99	350
28	USB Flash Drive	64GB USB 3.0 flash drive	3	8	15.99	500
29	Bluetooth Earbuds	True wireless earbuds with noise-cancellation	3	9	99.99	60

## Apply Some Queries:

### Query#01:

To find all customers who have placed an order?

```
SELECT
    Customer_ID,
    First_Name,
    Last_Name,
    Cust_UserName
FROM
    Customer
WHERE
    Customer_ID IN (
        SELECT DISTINCT Customer_ID
        FROM Customer_Order
    );
|
```

Results Explain Describe Saved SQL History

CUSTOMER_ID	FIRST_NAME	LAST_NAME	CUST_USERNAME
1	John	Doe	johndoe
2	Jane	Smith	janesmith
3	Alice	Johnson	alicej
4	Bob	Brown	bobbrown
5	Charlie	Taylor	charliet
6	Diana	Moore	dianam
7	Ethan	Clark	ethanc
8	Fiona	Lewis	fional
9	George	Martin	georgem
10	Hannah	Walker	hannahw
11	Ian	Harris	ianh
12	Julia	Young	juliay
13	Kevin	King	kevink
14	Laura	Wright	lauraw
15	Mike	Scott	mikesc
16	Nina	Green	ninag

### Query#02:

To find all products whose price is higher than the average price?

```
 Autocommit Display 30 ▾  
SELECT  
    Product_ID,  
    Prod_Name,  
    Price  
FROM  
    Product  
WHERE  
    Price > (SELECT AVG(Price) FROM Product);  
  
Results Explain Describe Saved SQL History
```

PRODUCT_ID	PROD_NAME	PRICE
1	Laptop	1200.99
2	Smartphone	799.99
3	Tablet	499.99
5	Smartwatch	299.99
8	Monitor	499.99
11	Camera	450
13	Projector	249.99
16	Drone	599.99
19	Virtual Reality Headset	399.99
20	Smart Glasses	299.99

10 rows returned in 0.02 seconds [CSV Export](#)

### Query#03:

To find the product description with the maximum sales (highest quantity ordered)?

```
 Autocommit Display 30 ▾  
SELECT  
    Prod_Description  
FROM  
(SELECT  
    Prod_Description  
    FROM  
        Product p  
    JOIN  
        Order_Details od ON p.Product_ID = od.Product_ID  
    JOIN  
        Customer_Order co ON od.Order_ID = co.Order_ID  
    GROUP BY  
        p.Product_ID, p.Prod_Description  
    ORDER BY
```

PROD_DESCRIPTION
Digital camera with 20MP resolution

1 rows returned in 0.04 seconds [CSV Export](#)

### Query#04:

To find the customer name who bought the most expensive product?

Autocommit Display 30

```
SELECT
    c.First_Name,
    c.Last_Name
FROM
    Customer c
JOIN
    Customer_Order co ON c.Customer_ID = co.Customer_ID
JOIN
    Order_Details od ON co.Order_ID = od.Order_ID
JOIN
    Product p ON od.Product_ID = p.Product_ID
WHERE
    p.Price = (SELECT MAX(Price) FROM Product);
```

Results Explain Describe Saved SQL History

FIRST_NAME	LAST_NAME
John	Doe

1 rows returned in 0.02 seconds [CSV Export](#)

## Create Views:

### View 1: Product Details with Supplier Information

User: GROCERY\_STORE

Home > SQL > **SQL Commands**

Autocommit Display 30

```
CREATE VIEW product_supplier_view AS
SELECT p.PRODUCT_ID,
       p.PROD_NAME,
       p.PROD_DESCRIPTION,
       p.PRICE,
       p.QUANTITY,
       s.SUPP_NAME,
       s.SUPP_ADDRESS
FROM product p
JOIN supplier s
  ON p.SUPPLIER_ID = s.SUPPLITER_ID;
```

Results Explain Describe Saved SQL History

View created.

0.01 seconds

Autocommit Display 30

```
select * from product_supplier_view;
```

**Results Explain Describe Saved SQL History**

PRODUCT_ID	PROD_NAME	PROD_DESCRIPTION	PRICE	QUANTITY	SUPP_NAME	SUPP_ADDRESS
1	Laptop	High-performance laptop with 16GB RAM	1200.99	50	Tech Supplies Inc.	123 Tech Street, Silicon Valley, CA
2	Smartphone	Latest model with 5G support	799.99	100	Gadget Guru Ltd.	456 Gadget Blvd, New York, NY
3	Tablet	10-inch tablet with 128GB storage	499.99	75	Wearable World Co.	789 Wearable Ave, Los Angeles, CA
4	Headphones	Noise-cancelling over-ear headphones	199.99	200	Sound Wave Electronics	101 Sound Rd, Austin, TX
5	Smartwatch	Waterproof smartwatch with fitness tracking	299.99	150	Mobile Max Ltd.	202 Mobile St, Miami, FL
6	Keyboard	Mechanical keyboard with RGB lighting	79.99	120	Appliance Hub LLC	303 Appliance Pkwy, Chicago, IL
7	Mouse	Wireless mouse with ergonomic design	29.99	250	Gaming Gear Supplies	404 Gaming Ln, Las Vegas, NV
8	Monitor	27-inch 4K monitor	499.99	60	Photo Tech Supplies	505 Photo Dr, Denver, CO
9	Charger	Fast-charging wall adapter	29.99	500	Home Plus Electronics	606 Home Blvd, San Francisco, CA
10	Speaker	Bluetooth portable speaker	99.99	80	Health Tech Solutions	707 Health St, Boston, MA
11	Camera	Digital camera with 20MP resolution	450	45	Tech Supplies Inc.	123 Tech Street, Silicon Valley, CA
12	Printer	Wireless inkjet printer	129.99	130	Gadget Guru Ltd.	456 Gadget Blvd, New York, NY
13	Projector	Portable mini projector for home theater	249.99	70	Wearable World Co.	789 Wearable Ave, Los Angeles, CA
14	Camera Lens	50mm lens for digital cameras	199.99	90	Sound Wave Electronics	101 Sound Rd, Austin, TX
15	Tripod	Adjustable tripod for photography	49.99	200	Mobile Max Ltd.	202 Mobile St, Miami, FL
16	Drone	Quadcopter with 4K camera	599.99	40	Appliance Hub LLC	Run SQL Command Line

## View 2: Product Details with Category and Supplier Information

User: GROCERY\_STORE

Home > SQL > SQL Commands

Autocommit Display 30

```
CREATE VIEW product_category_supplier_view AS
SELECT p.PRODUCT_ID,
       p.PROD_NAME,
       p.PROD_DESCRIPTION,
       p.PRICE,
       p.QUANTITY,
       s.SUPP_NAME,
       s.SUPP_ADDRESS,
       c.CAT_NAME
FROM product p
JOIN supplier s
ON p.SUPPLIER_ID = s.SUPPLIER_ID
JOIN category c
ON p.CATEGORY_ID = c.CATEGORY_ID;
```

**Results Explain Describe Saved SQL History**

View created.

0.02 seconds

```
select * from product_category_supplier_view;
```

Results Explain Describe Saved SQL History

PRODUCT_ID	PROD_NAME	PROD_DESCRIPTION	PRICE	QUANTITY	SUPP_NAME	SUPP_ADDRESS	CAT_NAME
1	Laptop	High-performance laptop with 16GB RAM	1200.99	50	Tech Supplies Inc.	123 Tech Street, Silicon Valley, CA	Electronics
2	Smartphone	Latest model with 5G support	799.99	100	Gadget Guru Ltd.	456 Gadget Blvd, New York, NY	Wearables
3	Tablet	10-inch tablet with 128GB storage	499.99	75	Wearable World Co.	789 Wearable Ave, Los Angeles, CA	Electronics
4	Headphones	Noise-cancelling over-ear headphones	199.99	200	Sound Wave Electronics	101 Sound Rd, Austin, TX	Accessories
5	Smartwatch	Waterproof smartwatch with fitness tracking	299.99	150	Mobile Max Ltd.	202 Mobile St, Miami, FL	Wearables
6	Keyboard	Mechanical keyboard with RGB lighting	79.99	120	Appliance Hub LLC	303 Appliance Pkwy, Chicago, IL	Accessories
7	Mouse	Wireless mouse with ergonomic design	29.99	250	Gaming Gear Supplies	404 Gaming Ln, Las Vegas, NV	Accessories
8	Monitor	27-inch 4K monitor	499.99	60	Photo Tech Supplies	505 Photo Dr, Denver, CO	Electronics
9	Charger	Fast-charging wall adapter	29.99	500	Home Plus Electronics	606 Home Blvd, San Francisco, CA	Accessories
10	Speaker	Bluetooth portable speaker	99.99	80	Health Tech Solutions	707 Health St, Boston, MA	Wearables
11	Camera	Digital camera with 20MP resolution	450	45	Tech Supplies Inc.	123 Tech Street, Silicon Valley, CA	Electronics
12	Printer	Wireless inkjet printer	129.99	130	Gadget Guru Ltd.	456 Gadget Blvd, New York, NY	Accessories
13	Projector	Portable mini projector for home theater	249.99	70	Wearable World Co.	789 Wearable Ave, Los Angeles, CA	Electronics
14	Camera Lens	50mm lens for digital cameras	199.99	90	Sound Wave Electronics	101 Sound Rd, Austin, TX	Electronics
15	Tripod	Adjustable tripod for photography	49.99	200	Mobile Max Ltd.	202 Mobile St, Miami, FL	Electronics
16	Drone	Quadcopter with 4K camera	599.99	40	Appliance Hub LLC	303 Appliance Pkwy, Chicago, IL	Electronics

## Create Triggers:

### Trigger to Prevent Insertion of Empty First\_Name for Customer.

User: GROCERY\_STORE

Home > SQL > SQL Commands

Autocommit Display 30

```
CREATE OR REPLACE TRIGGER Before_Insert_Customer
BEFORE INSERT ON Customer
FOR EACH ROW
BEGIN
    IF :NEW.First_Name IS NULL OR :NEW.First_Name = '' THEN
        RAISE_APPLICATION_ERROR(-20001, 'First name cannot be empty');
    END IF;
END;
```

Results Explain Describe Saved SQL History

Trigger created.

0.27 seconds

Autocommit Display 30

```
INSERT INTO Customer (Customer_ID, First_Name, Last_Name, Cust_UserName, Cust_password, Cust_Address)
VALUES (31, '', 'Smith', 'johnsmith', 'password123', '12349 Elm St');
```

Results Explain Describe Saved SQL History

```
ORA-20001: First name cannot be empty
ORA-06512: at "GROCERY_STORE.BEFORE_INSERT_CUSTOMER", line 3
ORA-04088: error during execution of trigger 'GROCERY_STORE.BEFORE_INSERT_CUSTOMER'
1. INSERT INTO Customer (Customer_ID, First_Name, Last_Name, Cust_UserName, Cust_password, Cust_Address)
2. VALUES (31, '', 'Smith', 'johnsmith', 'password123', '12349 Elm St');
```

## Trigger to Log Price Changes in Product Table

User: GROCERY\_STORE

Home > SQL > SQL Commands

Autocommit Display 30

```
CREATE TABLE Audit_Product (
    Product_ID NUMBER,
    Old_Price NUMBER,
    New_Price NUMBER,
    Change_Timestamp TIMESTAMP
);
CREATE OR REPLACE TRIGGER After_Update_Product
AFTER UPDATE ON Product
FOR EACH ROW
BEGIN
    INSERT INTO Audit_Product (Product_ID, Old_Price, New_Price, Change_Timestamp)
    VALUES (:OLD.Product_ID, :OLD.Price, :NEW.Price, SYSDATE);
END;
/
```

Results Explain Describe Saved SQL History

Trigger created.

0.06 seconds

User: GROCERY\_STORE

Home > SQL > **SQL Commands**

Autocommit Display 30 ▾

```
UPDATE Product
SET Price = 150
WHERE Product_ID = 15;

select * from Audit_Product;
```

Results Explain Describe Saved SQL History

PRODUCT_ID	OLD_PRICE	NEW_PRICE	CHANGE_TIMESTAMP
15	49.99	150	15-DEC-24 01.11.10.000000 PM

1 rows returned in 0.04 seconds [CSV Export](#)

## Define Roles:

**ADMIN\_ROLE** might have full access (SELECT, INSERT, UPDATE, DELETE) to all tables.

**CUSTOMER\_ROLE** might only have SELECT and UPDATE access to their own orders and data.

**SUPPLIER\_ROLE** might only have SELECT and UPDATE access to the Product table and view orders related to their products.

**SALES\_ROLE** might have SELECT access to customer orders and products but cannot modify.

```
SQL> -- Create roles
SQL> CREATE ROLE ADMIN_ROLE;

Role created.

SQL> CREATE ROLE CUSTOMER_ROLE;

Role created.

SQL> CREATE ROLE SUPPLIER_ROLE;

Role created.

SQL> CREATE ROLE SALES_ROLE;

Role created.
```

## Create Users:

```
SQL> -- Create users with valid passwords
SQL> CREATE USER admin_user IDENTIFIED BY Admin123;

User created.

SQL> CREATE USER customer_user IDENTIFIED BY Customer123;

User created.

SQL> CREATE USER supplier_user IDENTIFIED BY Supplier123;

User created.

SQL> CREATE USER sales_user IDENTIFIED BY Sales123;

User created.
```

### Assign Roles to Users:

```
SQL> -- Assign roles to users
SQL> GRANT ADMIN_ROLE TO admin_user;

Grant succeeded.

SQL> GRANT CUSTOMER_ROLE TO customer_user;

Grant succeeded.

SQL> GRANT SUPPLIER_ROLE TO supplier_user;

Grant succeeded.

SQL> GRANT SALES_ROLE TO sales_user;

Grant succeeded.
```

### Grant Privileges:

```
SQL>
SQL> -- Grant privileges for CUSTOMER_ROLE
SQL> GRANT SELECT, UPDATE ON Customer TO CUSTOMER_ROLE;

Grant succeeded.

SQL> GRANT SELECT, INSERT, UPDATE ON Customer_Order TO CUSTOMER_ROLE;

Grant succeeded.

SQL> GRANT SELECT ON Product TO CUSTOMER_ROLE;

Grant succeeded.

SQL>
SQL> -- Grant privileges for SUPPLIER_ROLE
SQL> GRANT SELECT, UPDATE ON Product TO SUPPLIER_ROLE;

Grant succeeded.

SQL> GRANT SELECT, UPDATE ON Order_Details TO SUPPLIER_ROLE;

Grant succeeded.

SQL>
SQL> -- Grant privileges for SALES_ROLE
SQL> GRANT SELECT ON Customer_Order TO SALES_ROLE;

Grant succeeded.

SQL> GRANT SELECT ON Product TO SALES_ROLE;

Grant succeeded.

SQL> GRANT SELECT ON Customer TO SALES_ROLE;

Grant succeeded.
```

## Storage Estimation:

To estimate the storage requirements for each table in the database, we can consider the following factors:

1. **Number of rows** in each table.
2. **Size of each column**, which depends on the data type used (e.g., INT, VARCHAR, DATE, etc.).

## Customer Table

Columns:

- Customer\_ID: INT (4 bytes)
- First\_Name: VARCHAR(50) (assuming average 25 characters, 25 bytes)
- Last\_Name: VARCHAR(50) (assuming average 25 characters, 25 bytes)
- Cust\_UserName: VARCHAR(50) (assuming average 25 characters, 25 bytes)

- 
- 

Cust\_password: VARCHAR(50) (assuming average 25 characters, 25 bytes)

Cust\_Address: VARCHAR(50) (assuming average 25 characters, 25 bytes)

There are 30 rows.

#### Row Size Calculation:

- Customer\_ID (INT): 4 bytes
- First\_Name (VARCHAR): 25 bytes
- Last\_Name (VARCHAR): 25 bytes
- Cust\_UserName (VARCHAR): 25 bytes
- Cust\_password (VARCHAR): 25 bytes
- Cust\_Address (VARCHAR): 25 bytes

Total row size:

$$4 + 25 + 25 + 25 + 25 + 25 = 129 \text{ bytes}$$

#### Estimated Storage:

- For 30 rows:  
$$30 * 129 \text{ bytes} = 3,870 \text{ bytes} \approx 3.87 \text{ KB}$$
 [Billing\\_Info](#)

#### Table:

Columns:

- Billing\_ID: INT (4 bytes)
- Bill\_Date: DATE (3 bytes)
- Customer\_ID: INT (4 bytes)

Assume there are 30 rows.

#### Row Size Calculation:

- Billing\_ID (INT): 4 bytes
- Bill\_Date (DATE): 3 bytes
- Customer\_ID (INT): 4 bytes

Total row size:

$$4 + 3 + 4 = 11 \text{ bytes}$$

**Estimated Storage:**

For 30 rows:

$$30 * 11 \text{ bytes} = 330 \text{ bytes } \textcolor{blue}{\text{Address}}$$

**Table:**

Columns:

- Address\_ID: INT (4 bytes)
  - Customer\_ID: INT (4 bytes)
  - Bill\_Address: VARCHAR(50) (assuming average 25 characters, 25 bytes)
- Assume there are 30 rows.

**Row Size Calculation:**

- Address\_ID (INT): 4 bytes
- Customer\_ID (INT): 4 bytes
- Bill\_Address (VARCHAR): 25 bytes

Total row size: 4 + 4

$$+ 25 = 33 \text{ bytes}$$

**Estimated Storage:**

- For 30 rows:

$$30 * 33 \text{ bytes} = 990 \text{ bytes } \textcolor{blue}{\text{Order_Table:}}$$

Columns:

- Order\_ID: INT (4 bytes)
- Order\_Date: DATE (3 bytes)
- Customer\_ID: INT (4 bytes)
- Shipping\_Method: VARCHAR(50) (assuming average 25 characters, 25 bytes)
- Billing\_ID: INT (4 bytes)

Assume there are 30 rows.

**Row Size Calculation:**

- Order\_ID (INT): 4 bytes

□

- Order\_Date (DATE): 3 bytes
- Customer\_ID (INT): 4 bytes
- Shipping\_Method (VARCHAR): 25 bytes
- Billing\_ID (INT): 4 bytes

Total row size:  $4 + 3 + 4 + 25$

+ 4 = 40 bytes **Estimated**

#### Storage:

- For 30 rows:  
 $30 * 40 \text{ bytes} = 1200 \text{ bytes.}$

### **Orde\_Details Table:**

Columns:

- Order\_Details\_ID: INT (4 bytes)
- Product\_ID: INT (4 bytes)
- Order\_ID: INT (4 bytes)
- Quantity: INT (4 bytes) Assume there are 30 rows.

#### Row Size Calculation:

- Order\_Details\_ID (INT): 4 bytes
- Product\_ID (INT): 4 bytes
- Order\_ID (INT): 4 bytes
- Quantity (INT): 4 bytes

Total row size:  $4 + 4 + 4$

+ 4 = 16 bytes

#### Estimated Storage:

- For 30 rows:  
 $30 * 16 \text{ bytes} = 480 \text{ bytes}$

### **Category Table:**

Assuming the following columns:

category\_id (INT) = 4 bytes category\_name  
(VARCHAR(100)) = 100 bytes  
cat\_description (VARCHAR(255)) = 255 bytes Total  
per row = 4 + 100 + 255 = **359 bytes**

#### **Estimated Storage:**

For 5 rows:

$5 \times 359 \text{ bytes} = 1795 \text{ bytes.}$

#### **Supplier Table:**

- SUPPLIER\_ID (INT) = 4 bytes
- SUPP\_USERNAME (VARCHAR(50)) = 50 bytes
- SUPP\_NAME (VARCHAR(100)) = 100 bytes
- SUPP\_ADDRESS (VARCHAR(255)) = 255 bytes

Storage Per Row:

- SUPPLIER\_ID: 4 bytes
- SUPP\_USERNAME: 50 bytes
- SUPP\_NAME: 100 bytes
- SUPP\_ADDRESS: 255 bytes Total per row =  $4 + 50 + 100 + 255 = 409 \text{ bytes.}$

#### **Estimated Storage:**

For 10 Rows:

$10 \times 409 \text{ bytes} = 4,090 \text{ bytes.}$

#### **Cart Table:**

- CART\_ID (INT) = 4 bytes
- CUSTOMER\_ID (INT) = 4 bytes
- PRODUCT\_ID (INT) = 4 bytes
- QUANTITY (INT) = 4 bytes Storage Per Row:

CART\_ID: 4 bytes

CUSTOMER\_ID: 4 bytes

PRODUCT\_ID: 4 bytes

- QUANTITY: 4 bytes

Total per row = 4 + 4 + 4 + 4 = 16 bytes.

#### **Estimated Storage:**

For 30 Rows:

$30 \times 16 \text{ bytes} = 480 \text{ bytes}$ .

#### **Admin Table:**

- ADMIN\_ID (INT) = 4 bytes
- ADMIN\_NAME (VARCHAR) = Let's assume 50 characters (50 bytes)
- ADMIN\_PASSWORD (VARCHAR) = Let's assume 50 characters (50 bytes)

Storage Per Row:

- ADMIN\_ID: 4 bytes
- ADMIN\_NAME: 50 bytes
- ADMIN\_PASSWORD: 50 bytes

Total per row = 4 + 50 + 50 = 104 bytes.

#### **Estimated Storage:**

For 1 Row:

$1 \times 104 \text{ bytes} = 104 \text{ bytes}$ .

### **Physical Database Design and Performance:**

#### **Composite Keys:**

- **Order\_Details:** This table will likely use a composite key of **Order\_ID** and **Product\_ID** to uniquely identify each product within an order. This helps track multiple products in a single order.
- **Cart:** A composite key of **Customer\_ID** and **Product\_ID** can be used to represent which products are in a customer's cart, especially since each customer can have many products in their cart.

## 2. Data Volumes and Access Frequencies

- **High Volume Tables:**
  - **Order and Order\_Details:** These tables will accumulate quickly as customers make purchases, so they will have high data volumes.
  - **Product:** The **Product** table will also grow in size as the store adds more items for sale.

### Access Frequency:

- **Order and Order\_Details:** These will be queried frequently during customer checkouts, updates to the cart, and customer order history retrieval.
- **Product and Category:** Queries to these tables will be common when customers browse or search products by category.
- **Low Frequency Tables:**
  - **Admin and Supplier:** These tables will be queried less frequently, mainly for system maintenance or updates related to suppliers or admin users.

## 3. Usage Analysis for Fields

- **Fields:**
  - Each field corresponds to a single piece of data. For example:
    - In **Customer**, the **First\_Name**, **Last\_Name**, **Cust\_UserName** are fields representing individual attributes of a customer.
    - In **Order**, **Order\_Date** and **Shipping\_Method** represent details related to the order.
  - **Attributes** like **Product\_ID** in **Order\_Details** represent foreign keys from the **Product** table, linking each product to its order.

## 4. Basic Design Decisions

- **Storage Type:**
  - **Number** data type is used for IDs, such as **Customer\_ID**, **Order\_ID**, and **Product\_ID**. This allows for efficient storage and easy indexing.

**VARCHAR** or **CHAR** is used for text fields like **First\_Name**, **Prod\_Name**, **Cust\_Address**, etc., depending on the expected length.

#### Data Integrity:

- **Primary Keys:**
  - Each table has a **Primary Key** (e.g., **Customer\_ID** for **Customer**, **Order\_ID** for **Order**).
- **Foreign Keys:**
  - **Order** has a foreign key to **Customer\_ID** (referencing **Customer**), and **Order\_Details** references both **Order\_ID** and **Product\_ID**.
  - **Billing\_info** and **Address** tables have foreign keys to **Customer\_ID**.
- **Unique Constraints:**
  - **Cust\_UserName** in **Customer** can have a unique constraint to prevent duplicate usernames.
  - **Supp\_UserName** in **Supplier** might also need to be unique.

### 5. Data Integrity Controls

- **Data Type Enforcement:**
  - Data types ensure that only valid data is stored:
    - **Price** in **Product** should be a **NUMBER** type to store decimal values accurately.
    - **Quantity** fields like in **Product** or **Order\_Details** should be integers, and non-integer values should raise an error.
- **Range Control:**
  - **Price** in **Product** may have a check constraint like **CHECK (Price > 0)** to ensure products are sold at valid prices.
  - **Quantity** in **Order\_Details** might have a range control ensuring values are within a realistic range (e.g., **CHECK (Quantity > 0)**).
- **Null Value Control:**
  - **First\_Name**, **Last\_Name** in **Customer** should not be null (they are required fields).

- **Address\_ID** in **Order** might allow **NULL** if shipping is not provided, but a **Billing\_ID** should always be present for a valid order.
- **Referential Integrity:**
  - **Billing\_info** refers to **Customer\_ID**, ensuring a billing record is always associated with a valid customer.
  - **Order** has a foreign key to **Customer\_ID**, ensuring orders are placed by valid customers.

## 6. Physical Database Design and Performance

- **Storage Formats for Attributes:**
  - **Product** fields like **Prod\_Name** and **Prod\_Description** will use **VARCHAR** to accommodate text-based product descriptions.
  - **Order\_ID** and **Product\_ID** will likely be **NUMBER** types for efficient numeric indexing and joins.
- **File Organization:**
  - For high-frequency tables like **Order**, **Order\_Details**, and **Product**, you may consider using **clustered indexing** to group data related to these tables together physically on the disk. This will help reduce disk I/O when querying.
- **Types of File Organization:**
  - **Heap Organization:** Simple, unsorted storage, appropriate for smaller, less frequently queried tables like **Supplier** or **Admin**.
  - **Indexed Organization:** **Product** and **Order** might benefit from indexed storage to optimize search operations.
- **Indexes and Their Appropriate Use:**
  - **Indexes** can be created on frequently searched fields, such as:
    - **Customer\_ID** in **Order** and **Billing\_info** for quick lookup of all orders related to a customer.
    - **Product\_ID** in **Order\_Details** for quickly identifying products within orders.
  - **Unique indexes** can be used on **Cust\_UserName** and **Supp\_UserName** to maintain uniqueness.

## 7. Translation of Database Model into Efficient Structures

- **Logical to Physical Translation:**

- The **ER Diagram** needs to be translated into tables, and then indexes and storage structures should be created for optimization.
- The **Order** table should be partitioned by **Order\_Date** to improve query performance when retrieving orders by date range.
- **Order\_Details** should use composite indexes for **Order\_ID** and **Product\_ID** for efficient querying.

This system serves as a valuable tool for grocery store owners and managers, helping them manage their resources effectively and stay competitive in a fast-paced retail environment.

**The End.....**