

MAHATMA EDUCATION SOCIETY'S
**PILLAI COLLEGE OF ARTS, COMMERCE &
SCIENCE**

(Autonomous)
NEW PANVEL

PROJECT REPORT ON
“Fitness Tracker web Application using asp.net c#”
IN PARTIAL FULFILLMENT OF
BACHELOR OF INFORMATION TECHNOLOGY

SEMESTER V – 2023-24

PROJECT GUIDE
Prof. Rashmi Chavan

SUBMITTED BY: Jatin Dnyaneshwar Gangare

ROLL NO: 9469



Mahatma Education Society's
Pillai College of Arts, Commerce & Science
(Autonomous)
Affiliated to University of Mumbai
NAAC Accredited 'A' grade (3 cycles)
Best College Award by University of Mumbai
ISO 9001:2015 Certified



Certificate

This is to verify that **Mr. Jatin Dnyaneshwar Gangare**
Examination **Seat no. 9469** has successfully completed all the
practicals of the **subject Advance Web Programming**
in partial fulfillment for the degree of **B.Sc. (Information
Technology) SEM V, affiliated to university of Mumbai** for
the academic year **2024-2025**.

Internal Examiner
Examiner

External



Mahatma Education Society's
Pillai College of Arts, Commerce & Science
 (Autonomous)
 Affiliated to University of Mumbai
 NAAC Accredited 'A' grade (3 cycles)
 Best College Award by University of Mumbai
 ISO 9001:2015 Certified



Class: TY Information Technology

Subject: Advance Web Programming

Details about the continuous Assessment 2/Project work:

Student Name: Jatin Dnyaneshwar Gangare

Roll No: 9469

Topic : Fitness Tracker web Application

Name of Evaluator: Prof. Rashmi Chavan

Grading Criteria	Fair	Good	Excellent	Total
Tools & Technique used	2	3	4	/5
Subject Knowledge	2	3	4	/5
Documentation/Report	2	3	4	/5
Total				/15

Co-ordinator,

Mrs. Anju Somani

Introduction :

The application is built using ASP.NET Web Forms and C#, providing users with the ability to track their workouts, exercises, and personal fitness metrics such as BMI and BMR. The system includes essential features like user authentication, personalized workout logging, and dynamic exercise management.

The application leverages a SQL Server database to store user data, workouts, and exercises. Users can sign up, log in, and access their personalized dashboard, which displays their fitness details and workout history. The dynamic interface allows users to add, update, or remove exercises from their workout routines in real time, making it flexible and user-friendly.

This project demonstrates key web development skills, including database connectivity, session management, and user interaction through forms, with a responsive design to ensure usability across various devices. The project is intended to provide a comprehensive fitness tracking solution while showcasing advanced web programming techniques.

Tools and Technologies Used

1. .NET Framework:
 - ASP.NET Web Forms: For creating dynamic web pages.
 - C#: For server-side programming and business logic.
 - Common Language Runtime (CLR): To manage the execution of C# code.
 - .NET Class Library: For accessing various pre-built functionalities.
2. ASP.NET Web Controls:
 - Web Forms Fundamentals: For structuring and managing the web pages.
 - Form Controls: Such as TextBoxes, Buttons, and Labels for user input and interaction.
 - Validation Controls: For validating user input on forms.
 - GridView: To display and manage data from the database.
3. State Management:
 - State Management: Utilizing Session State to maintain user session data across pages.
 - View State: To retain the state of controls on the page.
4. ADO.NET:
 - Direct Data Access: For connecting to and querying the SQL Server database.
 - SqlConnection, SqlCommand, SqlDataAdapter: For database operations like querying and updating data.
 - Data Controls: Such as GridView to display data dynamically.
5. SQL Server:
 - Database: For storing user information, workouts, and exercises.
 - SQL Queries: For retrieving and manipulating data.


Code :**User Database :**

```
CREATE TABLE [dbo].[users] (
    [un]   VARCHAR (50) NOT NULL,
    [ps]   NCHAR (50)  NOT NULL,
    [name] NCHAR (50)  NOT NULL,
    [age]   INT        NOT NULL,
    [gender] NCHAR (10) NOT NULL,
    [weight] FLOAT (53) NOT NULL,
    [height] FLOAT (53) NOT NULL,
    CONSTRAINT [PK_users] PRIMARY KEY CLUSTERED ([un] ASC)
);
```

	Name	Data Type	Allow Nulls	Default	
	un	varchar(50)	<input type="checkbox"/>		
	ps	nchar(50)	<input type="checkbox"/>		
	name	nchar(50)	<input type="checkbox"/>		
	age	int	<input type="checkbox"/>		
	gender	nchar(10)	<input type="checkbox"/>		
	weight	float	<input type="checkbox"/>		
	height	float	<input type="checkbox"/>		
			<input type="checkbox"/>		

Workout Database :

```
CREATE TABLE [dbo].[Workouts] (
    [Id]      INT      IDENTITY (1, 1) NOT NULL,
    [WorkoutType] NVARCHAR (100) NOT NULL,
    [Duration]   INT      NOT NULL,
    [CaloriesBurned] INT      NOT NULL,
    [WorkoutDate] DATE      NOT NULL,
    [Username]   VARCHAR (50) NOT NULL,
    CONSTRAINT [PK_Workouts] PRIMARY KEY CLUSTERED ([Id] ASC),
    CONSTRAINT [FK_Workouts_Users] FOREIGN KEY ([Username]) REFERENCES [dbo].[users] ([un])
ON DELETE CASCADE
);
```


	Name	Data Type	Allow Nulls	Default
	Id	int	<input type="checkbox"/>	
	WorkoutType	nvarchar(100)	<input type="checkbox"/>	
	Duration	int	<input type="checkbox"/>	
	CaloriesBurned	int	<input type="checkbox"/>	
	WorkoutDate	date	<input type="checkbox"/>	
	Username	varchar(50)	<input type="checkbox"/>	
			<input type="checkbox"/>	

Excercise Database :

```

CREATE TABLE [dbo].[Exercises] (
    [Id]          INT          IDENTITY (1, 1) NOT NULL,
    [WorkoutId]   INT          NOT NULL,
    [ExerciseName] NVARCHAR (100) NOT NULL,
    [Sets]        INT          NOT NULL,
    [Reps]        INT          NOT NULL,
    CONSTRAINT [PK_Exercises] PRIMARY KEY CLUSTERED ([Id] ASC),
    CONSTRAINT [FK_Exercises_Workouts] FOREIGN KEY ([WorkoutId]) REFERENCES [dbo].[Workouts]
([Id]) ON DELETE CASCADE
);

```

	Name	Data Type	Allow Nulls	Default
	Id	int	<input type="checkbox"/>	
	WorkoutId	int	<input type="checkbox"/>	
	ExerciseName	nvarchar(100)	<input type="checkbox"/>	
	Sets	int	<input type="checkbox"/>	
	Reps	int	<input type="checkbox"/>	
			<input type="checkbox"/>	

LoginForm.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="LoginForm.aspx.cs"
Inherits="Fitness_Tracker.LoginForm" %>
```

```
<!DOCTYPE html>
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
<head runat="server">
```

```
<title>Login</title>
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<style>
```

```
body {
```

```
background-image: url('FitnessWalpaper.jpg');
```

```
background-size: cover;
```

```
background-position: center;
```

```
background-attachment: fixed;
```

```
font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
```

```
margin: 0;
```

```
padding: 0;
```

```
height: 100vh;
```

```
display: flex;
```

```
justify-content: center;
```

```
align-items: center;
```

```
}
```

```
.login-container {
```

```
max-width: 500px;
```

```
width: 90%;
```

```
background-color: rgba(255, 255, 255, 0.8);
```

```
padding: 30px;
```

```
border-radius: 8px;
```

```
box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
```

```
text-align: center;
```

```
}
```

```
h2 {
```

```
margin-bottom: 20px;
```

```
color: #333;
```

```
}
```

```
.form-group {
```

```
margin-bottom: 20px;
```

```
}
```

```
.form-input {
```

```
width: 100%;
padding: 10px;
margin-bottom: 15px;
border-radius: 5px;
border: 1px solid #ced4da;
transition: border-color 0.3s ease;
}

.form-input:focus {
  border-color: #007bff;
  box-shadow: none;
  outline: none;
}

.login-button {
  width: 100%;
  padding: 10px;
  background-color: #007bff;
  border: none;
  color: #fff;
  border-radius: 5px;
  font-size: 16px;
  cursor: pointer;
  transition: background-color 0.3s ease;
}

.login-button:hover {
  background-color: #0056b3;
}

.sign-up {
  display: block;
  margin-top: 10px;
  font-size: 14px;
  color: #007bff;
  text-decoration: none;
}

.sign-up:hover {
  text-decoration: underline;
}

.text-danger {
  color: #e3342f !important;
  font-size: 0.875rem;
```



```

    }

    .logo {
        display: block;
        margin: 0 auto 20px;
        width: 100px;
    }
</style>
</head>
<body>
    <form id="form1" runat="server">
        <div class="login-container">
            <h2>Log In</h2>
            <div class="login-box">
                <asp:Image ID="LogoImage" runat="server" ImageUrl="loginpage.png" CssClass="logo" />

                <asp:Label ID="Label1" runat="server" AssociatedControlID="uname" CssClass="form-label"
Text="Username"></asp:Label>
                <asp:TextBox ID="uname" runat="server" CssClass="form-input" placeholder="Enter your
username or email"></asp:TextBox>

                <asp:Label ID="Label2" runat="server" AssociatedControlID="passw" CssClass="form-label"
Text="Password"></asp:Label>
                <asp:TextBox ID="passw" runat="server" CssClass="form-input" TextMode="Password"
placeholder="Enter your password"></asp:TextBox>

                <asp:Button ID="Button1" runat="server" CssClass="login-button" OnClick="Button1_Click"
Text="Login" />

                <asp:HyperLink ID="SignUpLink" runat="server" NavigateUrl="SignupForm.aspx"
CssClass="sign-up">Create an Account</asp:HyperLink>
            </div>
        </div>
    </form>
</body>
</html>

```

LoginForm.aspx.cs

```

using System;
using System.Collections.Generic;
using System.Data.SqlClient;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Configuration;

```

```

using Oracle.ManagedDataAccess.Client;
using System.Data;
namespace Fitness_Tracker
{
    public partial class LoginForm : System.Web.UI.Page
    {
        SqlConnection conn = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\jatin\source\repos\Fitness_Tracker\Fi
tness_Tracker\App_Data\Database1.mdf;Integrated Security=True");
        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["username"] != null)
            {
                Response.Redirect("HomeForm.aspx");
            }
            if (conn.State == ConnectionState.Open)
            {
                conn.Close();
            }
            conn.Open();
        }
        protected void Button1_Click(object sender, EventArgs e)
        {
            SqlCommand cmd = conn.CreateCommand();
            cmd.CommandType = CommandType.Text;
            cmd.CommandText = "select count(*) from users where un=@username and ps=@password";
            cmd.Parameters.AddWithValue("@username", uname.Text);
            cmd.Parameters.AddWithValue("@password", passw.Text);
            int count = (int)cmd.ExecuteScalar();
            if (count > 0)
            {
                Session["username"] = uname.Text;
                Response.Redirect("HomeForm.aspx");
            }
            else
            {
                ClientScript.RegisterStartupScript(this.GetType(), "alert", "alert('Wrong username or
password!');", true);
            }
            cmd.ExecuteNonQuery();
            uname.Text = "";
            passw.Text = "";
        }
    }
}

```

SignupForm.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="SignupForm.aspx.cs"
Inherits="Fitness_Tracker.SignupForm" UnobtrusiveValidationMode="None" %>
```

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head runat="server">
  <title>Signup Form</title>
  <style>
    body {
      background-image: url('deadpool.jpg');
      background-size: cover;
      background-position: center;
      background-attachment: fixed;
      font-family: 'Segoe UI', Tahoma, Geneva, Verdana, sans-serif;
      margin: 0;
      padding: 0;
      height: 100vh;
    }
    .signup-container {
      max-width: 500px;
      margin: 50px auto;
      background-color: rgba(255, 255, 255, 0.8);
      padding: 30px;
      border-radius: 8px;
      box-shadow: 0 4px 8px rgba(0, 0, 0, 0.2);
    }

    h2 {
      text-align: center;
      margin-bottom: 20px;
      color: #333;
    }
    .form-group {
      margin-bottom: 20px;
    }
    .form-control {
      border-radius: 5px;
      border: 1px solid #ced4da;
      transition: border-color 0.3s ease;
    }
  </style>
```

```

.form-control:focus {
  border-color: #007bff;
  box-shadow: none;
}

.btn-primary {
  background-color: #007bff;
  border-color: #007bff;
  transition: background-color 0.3s ease, border-color 0.3s ease;
}
.btn-primary:hover {
  background-color: #0056b3;
  border-color: #004085;
}
.btn-secondary {
  background-color: #6c757d;
  border-color: #6c757d;
  transition: background-color 0.3s ease, border-color 0.3s ease;
}
.btn-secondary:hover {
  background-color: #5a6268;
  border-color: #4e555b;
}

.text-danger {
  color: #e3342f !important;
  font-size: 0.875rem;
}

.form-check-label {
  font-size: 0.9rem;
}
</style>
</head>
<body>
<form id="form1" runat="server">
  <div class="signup-container">
    <h2>Create an Account</h2>

    <div class="row">

```

```

<div class="col-md-6 form-group">
    <asp:Label ID="lblName" runat="server" Text="Full Name"
AssociatedControlID="txtName" CssClass="font-weight-bold"></asp:Label>
    <asp:TextBox ID="txtName" runat="server" CssClass="form-control"
placeholder="Enter your full name"></asp:TextBox>
    <asp:RequiredFieldValidator ID="rfvName" runat="server"
ControlToValidate="txtName" ErrorMessage="Full Name is required" CssClass="text-danger"
Display="Dynamic" />
    <asp:RegularExpressionValidator ID="revName" runat="server"
ControlToValidate="txtName" ErrorMessage="Name should not contain numbers or
symbols" ValidationExpression="^[a-zA-Z\s]+$" CssClass="text-danger" Display="Dynamic"
/>
</div>
<div class="col-md-6 form-group">
    <asp:Label ID="lblUsername" runat="server" Text="Username"
AssociatedControlID="txtUsername" CssClass="font-weight-bold"></asp:Label>
    <asp:TextBox ID="txtUsername" runat="server" CssClass="form-control"
placeholder="Enter your username"></asp:TextBox>
    <asp:RequiredFieldValidator ID="rfvUsername" runat="server"
ControlToValidate="txtUsername" ErrorMessage="Username is required" CssClass="text-
danger" Display="Dynamic" />
    <asp:RegularExpressionValidator ID="revUsername" runat="server"
ControlToValidate="txtUsername" ErrorMessage="Username should only contain letters and
numbers" ValidationExpression="^[a-zA-Z0-9]+$" CssClass="text-danger" Display="Dynamic"
/>
</div>
</div>
<div class="row">
    <div class="col-md-6 form-group">
        <asp:Label ID="lblAge" runat="server" Text="Age" AssociatedControlID="txtAge"
CssClass="font-weight-bold"></asp:Label>
        <asp:TextBox ID="txtAge" runat="server" CssClass="form-control"
placeholder="Enter your age"></asp:TextBox>
        <asp:RequiredFieldValidator ID="rfvAge" runat="server"
ControlToValidate="txtAge" ErrorMessage="Age is required" CssClass="text-danger"
Display="Dynamic" />
        <asp:RegularExpressionValidator ID="revAge" runat="server"
ControlToValidate="txtAge" ErrorMessage="Age should be a whole number"
ValidationExpression="^\d+$" CssClass="text-danger" Display="Dynamic" />
    </div>

```

```

<div class="col-md-6 form-group">
    <asp:Label ID="lblGender" runat="server" Text="Gender"
AssociatedControllID="ddlGender" CssClass="font-weight-bold"></asp:Label>
    <asp:DropDownList ID="ddlGender" runat="server" CssClass="form-control">
        <asp:ListItem Text="Select Gender" Value="" />
        <asp:ListItem Text="Male" Value="Male" />
        <asp:ListItem Text="Female" Value="Female" />
        <asp:ListItem Text="Other" Value="Other" />
    </asp:DropDownList>
    <asp:RequiredFieldValidator ID="rfvGender" runat="server"
ControlToValidate="ddlGender" InitialValue="" ErrorMessage="Please select a gender"
CssClass="text-danger" Display="Dynamic" />
</div>
</div>
<div class="row">
    <div class="col-md-6 form-group">
        <asp:Label ID="lblWeight" runat="server" Text="Weight (kg)"
AssociatedControllID="txtWeight" CssClass="font-weight-bold"></asp:Label>
        <asp:TextBox ID="txtWeight" runat="server" CssClass="form-control"
placeholder="Enter your weight in kg"></asp:TextBox>
        <asp:RequiredFieldValidator ID="rfvWeight" runat="server"
ControlToValidate="txtWeight" ErrorMessage="Weight is required" CssClass="text-danger"
Display="Dynamic" />
        <asp:RegularExpressionValidator ID="revWeight" runat="server"
ControlToValidate="txtWeight" ErrorMessage="Weight should be a numeric value"
ValidationExpression="^\d+(\.\d+)?$" CssClass="text-danger" Display="Dynamic" />
    </div>

    <div class="col-md-6 form-group">
        <asp:Label ID="lblHeight" runat="server" Text="Height (cm)"
AssociatedControllID="txtHeight" CssClass="font-weight-bold"></asp:Label>
        <asp:TextBox ID="txtHeight" runat="server" CssClass="form-control"
placeholder="Enter your height in cm"></asp:TextBox>
        <asp:RequiredFieldValidator ID="rfvHeight" runat="server"
ControlToValidate="txtHeight" ErrorMessage="Height is required" CssClass="text-danger"
Display="Dynamic" />
        <asp:RegularExpressionValidator ID="revHeight" runat="server"
ControlToValidate="txtHeight" ErrorMessage="Height should be a numeric value"
ValidationExpression="^\d+(\.\d+)?$" CssClass="text-danger" Display="Dynamic" />
    </div>
</div>

```

```

<div class="row">
  <div class="col-md-6 form-group">
    <asp:Label ID="lblPassword" runat="server" Text="Password"
AssociatedControlID="txtPassword" CssClass="font-weight-bold"></asp:Label>
    <asp:TextBox ID="txtPassword" runat="server" TextMode="Password"
CssClass="form-control" placeholder="Enter your password"></asp:TextBox>
    <asp:RequiredFieldValidator ID="rfvPassword" runat="server"
ControlToValidate="txtPassword" ErrorMessage="Password is required" CssClass="text-
danger" Display="Dynamic" />
    <asp:RegularExpressionValidator ID="revPassword" runat="server"
ControlToValidate="txtPassword" ErrorMessage="Password must be at least 8 characters
long" ValidationExpression="^.{8,}$" CssClass="text-danger" Display="Dynamic" />
  </div>

  <div class="col-md-6 form-group">
    <asp:Label ID="lblConfirmPassword" runat="server" Text="Confirm Password"
AssociatedControlID="txtConfirmPassword" CssClass="font-weight-bold"></asp:Label>
    <asp:TextBox ID="txtConfirmPassword" runat="server" TextMode="Password"
CssClass="form-control" placeholder="Re-enter your password"></asp:TextBox>
    <asp:RequiredFieldValidator ID="rfvConfirmPassword" runat="server"
ControlToValidate="txtConfirmPassword" ErrorMessage="Please confirm your password"
CssClass="text-danger" Display="Dynamic" />
    <asp:CompareValidator ID="cvPassword" runat="server"
ControlToCompare="txtPassword" ControlToValidate="txtConfirmPassword"
ErrorMessage="Passwords do not match" CssClass="text-danger" Display="Dynamic" />
  </div>
</div>

<div class="form-group">
  <asp:Button ID="btnSubmit" runat="server" Text="Sign Up" CssClass="btn btn-
primary btn-block" OnClick="btnSubmit_Click" />
  <asp:Button ID="btnReset" runat="server" Text="Reset" CssClass="btn btn-
secondary btn-block" OnClick="btnReset_Click" CausesValidation="false" />
</div>
<asp:HyperLink ID="LoginLink" runat="server" NavigateUrl="LoginForm.aspx"
CssClass="sign-up">Have Account Login</asp:HyperLink>
</div>
</form>
</body>
</html>

```

SignupForm.aspx.cs

```

using System;
using System.Data;
using System.Data.SqlClient;
using System.Web.UI;

namespace Fitness_Tracker
{
    public partial class SignupForm : System.Web.UI.Page
    {
        SqlConnection conn = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\jatin\source\repos\Fitness_T
racker\Fitness_Tracker\App_Data\Database1.mdf;Integrated Security=True");

        protected void Page_Load(object sender, EventArgs e)
        {
            if (conn.State == ConnectionState.Open)
            {
                conn.Close();
            }
            conn.Open();
        }

        protected void btnSubmit_Click(object sender, EventArgs e)
        {
            SqlCommand checkUsernameCmd = new SqlCommand("SELECT COUNT(*) FROM
users WHERE un = @username", conn);
            checkUsernameCmd.Parameters.AddWithValue("@username", txtUsername.Text);
            int userExists = (int)checkUsernameCmd.ExecuteScalar();

            if (userExists > 0)
            {
                ScriptManager.RegisterStartupScript(this, this.GetType(), "alert", "alert('Username
already exists. Please choose a different username.');" , true);
            }
            else
            {
                SqlCommand cmd = conn.CreateCommand();
                cmd.CommandType = CommandType.Text;
                cmd.CommandText = "INSERT INTO users (un, ps, name, age, gender, weight,
height) VALUES (@username, @password, @name, @age, @gender, @weight, @height)";
                cmd.Parameters.AddWithValue("@username", txtUsername.Text);
            }
        }
    }
}

```



```

cmd.Parameters.AddWithValue("@password", txtPassword.Text);
cmd.Parameters.AddWithValue("@name", txtName.Text);
cmd.Parameters.AddWithValue("@age", txtAge.Text);
cmd.Parameters.AddWithValue("@gender", ddlGender.Text);
cmd.Parameters.AddWithValue("@weight", txtWeight.Text);
cmd.Parameters.AddWithValue("@height", txtHeight.Text);

int count = cmd.ExecuteNonQuery();
if (count > 0)
{

    ScriptManager.RegisterStartupScript(this, this.GetType(), "alert",
"alert('Successfully Signed Up! Ready to Login.');" + window.location='LoginForm.aspx';", true);

}
else
{
    ScriptManager.RegisterStartupScript(this, this.GetType(), "alert", "alert('Failed to
sign up. Please try again!');" + true);
}
}
}

protected void btnReset_Click(object sender, EventArgs e)
{
    txtName.Text = "";
    txtUsername.Text = "";
    txtAge.Text = "";
    ddlGender.SelectedIndex = 0;
    txtWeight.Text = "";
    txtHeight.Text = "";
    txtPassword.Text = "";
    txtConfirmPassword.Text = "";
}
}
}

```

WorkoutForm.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WorkoutForm.aspx.cs"
Inherits="Fitness_Tracker.WorkoutForm" UnobtrusiveValidationMode="None" %>
```

```
<!DOCTYPE html>
<head runat="server">
  <title>Log Workout</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      margin: 0;
      padding: 0;
    }

    form {
      max-width: 800px;
      margin: 50px auto;
      background-color: #fff;
      padding: 20px;
      border-radius: 8px;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    }

    h2 {
      text-align: center;
      margin-bottom: 20px;
      color: #333;
    }

    .form-group {
      margin-bottom: 15px;
    }

    .form-group label {
      display: block;
      margin-bottom: 5px;
      color: #555;
    }

    .form-control {
      width: 100%;
      padding: 10px;
      border-radius: 4px;
      border: 1px solid #ddd;
      box-sizing: border-box;
    }
```

```
.btn {  
  padding: 10px 20px;  
  border-radius: 4px;  
  border: none;  
  cursor: pointer;  
}  
  
.btn-primary {  
  background-color: #007bff;  
  color: white;  
}  
  
.btn-secondary {  
  background-color: #6c757d;  
  color: white;  
  margin-bottom: 20px;  
}  
  
.btn-danger {  
  background-color: #dc3545;  
  color: white;  
}  
  
.btn:hover {  
  opacity: 0.9;  
}  
  
.exercise-container {  
  margin-top: 20px;  
  margin-bottom: 20px;  
  background-color: #f8f9fa;  
  padding: 15px;  
  border-radius: 5px;  
  border: 1px solid #ddd;  
}  
  
.exercise-item {  
  margin-bottom: 20px;  
  padding: 10px;  
  border: 1px solid #ddd;  
  border-radius: 5px;  
  background-color: #ffffff;  
}  
  
.exercise-item .form-group {  
  margin-bottom: 10px;  
}
```

```

.remove-exercise {
    margin-top: 10px;
    float: right;
}

hr {
    border-top: 1px solid #ddd;
}

.text-danger {
    color: #dc3545;
}
</style>
</head>
<body>
<form id="form1" runat="server">
    <div>
        <h2>Log Workout</h2>
        <asp:ValidationSummary ID="ValidationSummary1" runat="server" CssClass="text-danger" />

        <div class="form-group">
            <asp:Label ID="lblWorkoutType" runat="server" Text="Workout Type" CssClass="control-label" Font-Bold="True" Font-Italic="False" Font-Overline="False" ForeColor="#3399FF"></asp:Label>
            <asp:TextBox ID="txtWorkoutType" runat="server" CssClass="form-control"></asp:TextBox>
            <asp:RequiredFieldValidator ID="rfvWorkoutType" runat="server"
ControlToValidate="txtWorkoutType" ErrorMessage="Workout type is required." CssClass="text-danger" />
        </div>
        <div class="form-group">
            <asp:Label ID="lblDuration" runat="server" Text="Duration (minutes)" CssClass="control-label" Font-Bold="True" ForeColor="#3399FF"></asp:Label>
            <asp:TextBox ID="txtDuration" runat="server" CssClass="form-control"></asp:TextBox>
            <asp:RequiredFieldValidator ID="rfvDuration" runat="server"
ControlToValidate="txtDuration" ErrorMessage="Duration is required." CssClass="text-danger" />
            <asp:RangeValidator ID="rvDuration" runat="server" ControlToValidate="txtDuration"
MinimumValue="1" MaximumValue="1440" Type="Integer" ErrorMessage="Please enter a valid duration (1-1440 minutes)." CssClass="text-danger" />
        </div>
        <div class="form-group">
            <asp:Label ID="lblCaloriesBurned" runat="server" Text="Calories Burned"
CssClass="control-label" Font-Bold="True" ForeColor="#3399FF"></asp:Label>
            <asp:TextBox ID="txtCaloriesBurned" runat="server" CssClass="form-control"></asp:TextBox>
            <asp:RequiredFieldValidator ID="rfvCaloriesBurned" runat="server"
ControlToValidate="txtCaloriesBurned" ErrorMessage="Calories burned is required." CssClass="text-danger" />
        </div>
    </div>
</form>

```

```

        <asp:RangeValidator ID="rvCaloriesBurned" runat="server"
ControlToValidate="txtCaloriesBurned" MinimumValue="1" MaximumValue="10000" Type="Integer"
ErrorMessage="Please enter a valid number of calories burned." CssClass="text-danger" />
    </div>
    <div class="form-group">
        <asp:Label ID="lblDate" runat="server" Text="Workout Date" CssClass="control-label" Font-
Bold="True" ForeColor="#3399FF"></asp:Label>
        <asp:TextBox ID="txtDate" runat="server" CssClass="form-control"
TextMode="Date"></asp:TextBox>
        <asp:RequiredFieldValidator ID="rfvDate" runat="server" ControlToValidate="txtDate"
ErrorMessage="Date is required." CssClass="text-danger" />
    </div>

    <div class="form-group">
        <asp:Label ID="lblNumberOfExercises" runat="server" Text="Number of Exercises"
CssClass="control-label" Font-Bold="True" ForeColor="#3399FF"></asp:Label>
        <asp:TextBox ID="txtNumberOfExercises" runat="server" CssClass="form-
control"></asp:TextBox>
        <asp:RequiredFieldValidator ID="rfvNumberOfExercises" runat="server"
ControlToValidate="txtNumberOfExercises" ErrorMessage="Number of exercises is required."
CssClass="text-danger" />
        <asp:RangeValidator ID="rvNumberOfExercises" runat="server"
ControlToValidate="txtNumberOfExercises" MinimumValue="1" MaximumValue="50"
Type="Integer" ErrorMessage="Please enter a valid number of exercises (1-50)." CssClass="text-
danger" />
    </div>
    <div class="form-group">
        <asp:Button ID="btnAddExercises" runat="server" Text="Add Exercises" CssClass="btn btn-
secondary" OnClick="btnAddExercises_Click" />
    </div>

    <div class="exercise-container" id="exerciseContainer" runat="server">

</div>

    <div class="form-group">
        <asp:Button ID="btnSave" runat="server" Text="Save Workout" CssClass="btn btn-primary"
OnClick="btnSave_Click" />
        <asp:Button ID="btncancel" runat="server" Text="Cancel" CssClass="btn btn-primary"
OnClick="btnCancel_Click" CausesValidation="False"/>
    </div>
</div>
</form>
</body>
</html>

```

WorkoutForm.aspx.cs

```

using System;
using System.Data;
using System.Data.SqlClient;
using System.Web.UI;
using System.Web.UI.WebControls;

namespace Fitness_Tracker
{
    public partial class WorkoutForm : System.Web.UI.Page
    {
        SqlConnection conn = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\jatin\source\repos\Fitness_Tracker\Fi
tness_Tracker\App_Data\Database1.mdf;Integrated Security=True");

        private int NumberOfExercises
        {
            get
            {
                return ViewState["NumberOfExercises"] != null ? (int)ViewState["NumberOfExercises"] : 0;
            }
            set
            {
                ViewState["NumberOfExercises"] = value;
            }
        }

        protected void Page_Load(object sender, EventArgs e)
        {
            if (!IsPostBack)
            {
                txtDate.Text = DateTime.Now.ToString("yyyy-MM-dd");
            }

            if (conn.State == ConnectionState.Open)
            {
                conn.Close();
            }
            conn.Open();

            RecreateExerciseControls();
        }

        protected void btnSave_Click(object sender, EventArgs e)
        {
            try

```

```

{

    string username = Session["Username"] as string;

    SqlCommand cmd = conn.CreateCommand();
    cmd.CommandType = CommandType.Text;
    cmd.CommandText = "INSERT INTO Workouts (WorkoutType, Duration, CaloriesBurned,
WorkoutDate, Username) OUTPUT INSERTED.Id VALUES (@WorkoutType, @Duration,
@CaloriesBurned, @WorkoutDate, @Username)";
    cmd.Parameters.AddWithValue("@WorkoutType", txtWorkoutType.Text);
    cmd.Parameters.AddWithValue("@Duration", int.Parse(txtDuration.Text));
    cmd.Parameters.AddWithValue("@CaloriesBurned", int.Parse(txtCaloriesBurned.Text));
    cmd.Parameters.AddWithValue("@WorkoutDate", DateTime.Parse(txtDate.Text));
    cmd.Parameters.AddWithValue("@Username", username);

    int workoutId = (int)cmd.ExecuteScalar();

    for (int i = 0; i < NumberOfExercises; i++)
    {

        string exerciseName =
((TextBox)exerciseContainer.FindControl($"exerciseName_{i}")).Text;
        int sets = int.Parse(((TextBox)exerciseContainer.FindControl($"exerciseSets_{i}")).Text);
        int reps = int.Parse(((TextBox)exerciseContainer.FindControl($"exerciseReps_{i}")).Text);

        SqlCommand cmdExercise = conn.CreateCommand();
        cmdExercise.CommandType = CommandType.Text;
        cmdExercise.CommandText = "INSERT INTO Exercises (WorkoutId, ExerciseName, Sets,
Reps) VALUES (@WorkoutId, @ExerciseName, @Sets, @Reps)";
        cmdExercise.Parameters.AddWithValue("@WorkoutId", workoutId);
        cmdExercise.Parameters.AddWithValue("@ExerciseName", exerciseName);
        cmdExercise.Parameters.AddWithValue("@Sets", sets);
        cmdExercise.Parameters.AddWithValue("@Reps", reps);
        cmdExercise.ExecuteNonQuery();
    }

    Response.Write("<script>alert('Workout and exercises saved successfully!');</script>");
    ClearForm();
}
catch (Exception ex)
{
    Response.Write("<script>alert('An error occurred: " + ex.Message + "');</script>");
}
}

```

```

protected void btnAddExercises_Click(object sender, EventArgs e)
{

    NumberOfExercises = int.Parse(txtNumberOfExercises.Text);

    AddExerciseControls(NumberOfExercises);
}

private void AddExerciseControls(int count)
{
    for (int i = 0; i < count; i++)
    {

        if (exerciseContainer.FindControl($"exercisePanel_{i}") == null)
        {
            Panel exercisePanel = new Panel();
            exercisePanel.ID = $"exercisePanel_{i}";
            exercisePanel.CssClass = "exercise-item";

            Label lblExerciseName = new Label();
            lblExerciseName.Text = "Exercise Name:";
            lblExerciseName.CssClass = "control-label";
            exercisePanel.Controls.Add(lblExerciseName);

            TextBox txtExerciseName = new TextBox();
            txtExerciseName.ID = $"exerciseName_{i}";
            txtExerciseName.CssClass = "form-control";
            exercisePanel.Controls.Add(txtExerciseName);

            Label lblSets = new Label();
            lblSets.Text = "Sets:";
            lblSets.CssClass = "control-label";
            exercisePanel.Controls.Add(lblSets);

            TextBox txtSets = new TextBox();
            txtSets.ID = $"exerciseSets_{i}";
            txtSets.CssClass = "form-control";
            exercisePanel.Controls.Add(txtSets);

            Label lblReps = new Label();
            lblReps.Text = "Reps:";
            lblReps.CssClass = "control-label";
            exercisePanel.Controls.Add(lblReps);
        }
    }
}

```



```

        TextBox txtReps = new TextBox();
        txtReps.ID = $"exerciseReps_{i}";
        txtReps.CssClass = "form-control";
        exercisePanel.Controls.Add(txtReps);

        Button btnRemove = new Button();
        btnRemove.Text = "Remove";
        btnRemove.CssClass = "btn btn-danger remove-exercise";
        btnRemove.Click += (s, args) => { exerciseContainer.Controls.Remove(exercisePanel); };
        exercisePanel.Controls.Add(btnRemove);

        exerciseContainer.Controls.Add(exercisePanel);
    }
}

private void RecreateExerciseControls()
{
    AddExerciseControls(NumberOfExercises);
}

private void ClearForm()
{
    txtWorkoutType.Text = "";
    txtDuration.Text = "";
    txtCaloriesBurned.Text = "";
    txtDate.Text = DateTime.Now.ToString("yyyy-MM-dd");
    txtNumberOfExercises.Text = "";
    exerciseContainer.Controls.Clear();
    NumberOfExercises = 0;
}

protected void btnCancel_Click(object sender, EventArgs e)
{
    Response.Redirect("HomeForm.aspx");
}
}

```

HomeForm.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="HomeForm.aspx.cs"
Inherits="Fitness_Tracker.HomeForm" %>
```

```
<!DOCTYPE html>
```

```
<head runat="server">
```

```
  <title>Home</title>
```

```
  <style>
```

```
    body {
```

```
      font-family: Arial, sans-serif;
```

```
      margin: 0;
```

```
      padding: 0;
```

```
    }
```

```
    .navbar {
```

```
      background-color: #007bff;
```

```
      color: white;
```

```
      padding: 10px;
```

```
      display: flex;
```

```
      align-items: center;
```

```
      justify-content: space-between;
```

```
    }
```

```
    .navbar-content {
```

```
      display: flex;
```

```
      align-items: center;
```

```
      flex-grow: 1;
```

```
    }
```

```
    .navbar-left {
```

```
      flex: 1;
```

```
    }
```

```
    .navbar-right {
```

```
      margin-left: auto;
```

```
    }
```

```
    .navbar span {
```

```
      color: white;
```

```
      text-decoration: none;
```

```
      padding: 10px;
```

```
      font-size: 16px;
```

```
      font-weight: bold;
```

```
      background-color: #0056b3;
```

```
      border-radius: 5px;
```

```
    }
```

```
    .navbar a {
```

```
    color: black;
    text-decoration: none;
    padding: 10px;
    font-size: 16px;
    font-weight: bold;
}

.navbar a:hover {
    background-color: #0056b3;
    border-radius: 5px;
}

.navbar-link {
    color: white;
    text-decoration: none;
    padding: 10px;
    font-size: 16px;
    font-weight: bold;
}

.navbar-link:hover {
    background-color: transparent;
    text-decoration-color: red;
}

.logout-button {
    background-color: #dc3545;
    color: white;
    border: none;
    padding: 10px 20px;
    border-radius: 5px;
    cursor: pointer;
}

.logout-button:hover {
    background-color: #c82333;
}

.workout-table {
    width: 100%;
    border-collapse: collapse;
    margin: 20px 0;
}

.workout-table th, .workout-table td {
    border: 1px solid #ddd;
    padding: 8px;
}
```

```

.workout-table th {
    background-color: #f4f4f4;
}

.user-info-table {
    width: 100%;
    margin: 20px 0;
    border-collapse: collapse;
}

.user-info-table th, .user-info-table td {
    border: 1px solid #ddd;
    padding: 8px;
}

.user-info-table th {
    background-color: #f4f4f4;
}
</style>
</head>
<body>
    <form id="form1" runat="server">
        <div class="navbar">
            <div class="navbar-left">
                <span>Fitness Tracker</span>
            </div>
            <div class="navbar-content">
                <asp:HyperLink ID="addWorkout" runat="server" NavigateUrl="WorkoutForm.aspx"
                CssClass="navbar-link" Font-Bold="True">Add Workout</asp:HyperLink>
            </div>
            <div class="navbar-right">
                <asp:Button ID="btnLogout" runat="server" Text="Logout" CssClass="logout-button"
                OnClick="btnLogout_Click" />
            </div>
        </div>

        <asp:GridView ID="gvUserInfo" runat="server" AutoGenerateColumns="False" CssClass="user-
        info-table">
            <Columns>
                <asp:BoundField DataField="name" HeaderText="Name" />
                <asp:BoundField DataField="age" HeaderText="Age" />
                <asp:BoundField DataField="gender" HeaderText="Gender" />
                <asp:BoundField DataField="weight" HeaderText="Weight (kg)" />
                <asp:BoundField DataField="height" HeaderText="Height (cm)" />
                <asp:BoundField DataField="BMI" HeaderText="BMI" />
                <asp:BoundField DataField="BMR" HeaderText="BMR" />
            </Columns>

```

```

</asp:GridView>

<asp:GridView ID="gvWorkouts" runat="server" AutoGenerateColumns="False"
CssClass="workout-table">
    <Columns>
        <asp:BoundField DataField="WorkoutType" HeaderText="Workout Type" />
        <asp:BoundField DataField="Duration" HeaderText="Duration (minutes)" />
        <asp:BoundField DataField="CaloriesBurned" HeaderText="Calories Burned" />
        <asp:BoundField DataField="WorkoutDate" HeaderText="Workout Date" />
        <asp:TemplateField HeaderText="Exercises">
            <ItemTemplate>
                <asp:GridView ID="gvExercises" runat="server" AutoGenerateColumns="False"
CssClass="workout-table">
                    <Columns>
                        <asp:BoundField DataField="ExerciseName" HeaderText="Exercise Name" />
                        <asp:BoundField DataField="Sets" HeaderText="Sets" />
                        <asp:BoundField DataField="Reps" HeaderText="Reps" />
                    </Columns>
                </asp:GridView>
            </ItemTemplate>
        </asp:TemplateField>
    </Columns>
</asp:GridView>
</form>
</body>
</html>

```

HomeForm.aspx.cs

```

using System;
using System.Data;
using System.Data.SqlClient;
using System.Web.UI.WebControls;

namespace Fitness_Tracker
{
    public partial class HomeForm : System.Web.UI.Page
    {
        SqlConnection conn = new SqlConnection(@"Data
Source=(LocalDB)\MSSQLLocalDB;AttachDbFilename=C:\Users\jatin\source\repos\Fitness_Tracker\Fi
tness_Tracker\App_Data\Database1.mdf;Integrated Security=True");

        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["Username"] == null)
            {
                Response.Redirect("LoginForm.aspx");
            }
        }
    }
}

```

```

else
{
    if (!IsPostBack)
    {
        LoadUserInfo();
        LoadWorkouts();
    }
}
}

protected void btnLogout_Click(object sender, EventArgs e)
{
    Session.Clear();
    Session.Abandon();
    Response.Redirect("LoginForm.aspx");
}

private void LoadUserInfo()
{
    string username = Session["Username"] as string;

    using (SqlCommand cmd = new SqlCommand("SELECT name, age, gender, weight, height
FROM users WHERE un = @Username", conn))
    {
        cmd.Parameters.AddWithValue("@Username", username);

        SqlDataAdapter da = new SqlDataAdapter(cmd);
        DataTable dtUserInfo = new DataTable();
        da.Fill(dtUserInfo);

        if (dtUserInfo.Rows.Count > 0)
        {
            DataRow row = dtUserInfo.Rows[0];

            double weight = Convert.ToDouble(row["weight"]);
            double height = Convert.ToDouble(row["height"]);
            int age = Convert.ToInt32(row["age"]);
            string gender = Convert.ToString(row["gender"]);

            double bmi = weight / Math.Pow(height / 100, 2);

            double bmr = 0;
            bmr = 88.362 + (13.397 * weight) + (4.799 * height) - (5.677 * age);

            dtUserInfo.Columns.Add("BMI", typeof(double));
            dtUserInfo.Columns.Add("BMR", typeof(double));
            row["BMI"] = Math.Round(bmi, 2);
            row["BMR"] = Math.Round(bmr, 2);
        }
    }
}

```

```

        gvUserInfo.DataSource = dtUserInfo;
        gvUserInfo.DataBind();
    }
}

private void LoadWorkouts()
{
    string username = Session["Username"] as string;

    using (SqlCommand cmd = new SqlCommand("SELECT Id, WorkoutType, Duration,
CaloriesBurned, WorkoutDate FROM Workouts WHERE Username = @Username", conn))
    {
        cmd.Parameters.AddWithValue("@Username", username);

        SqlDataAdapter da = new SqlDataAdapter(cmd);
        DataTable dtWorkouts = new DataTable();
        da.Fill(dtWorkouts);

        gvWorkouts.DataSource = dtWorkouts;
        gvWorkouts.DataBind();

        foreach (GridViewRow row in gvWorkouts.Rows)
        {
            int workoutId = Convert.ToInt32(dtWorkouts.Rows[row.RowIndex]["Id"]);

            GridView gvExercises = (GridView)row.FindControl("gvExercises");

            using (SqlCommand cmdExercises = new SqlCommand("SELECT ExerciseName, Sets, Reps
FROM Exercises WHERE WorkoutId = @WorkoutId", conn))
            {
                cmdExercises.Parameters.AddWithValue("@WorkoutId", workoutId);

                SqlDataAdapter daExercises = new SqlDataAdapter(cmdExercises);
                DataTable dtExercises = new DataTable();
                daExercises.Fill(dtExercises);

                gvExercises.DataSource = dtExercises;
                gvExercises.DataBind();
            }
        }
    }
}

```

Output :

The screenshot shows a web browser window with the address bar displaying `https://localhost:44325/LoginForm.aspx`. The page features a login form overlay with the title "Log In". The form includes a "Username" field with the placeholder text "Enter your username or email", a "Password" field with the placeholder text "Enter your password", and a blue "Login" button. Below the button is a link that says "Create an Account". The background image is a black and white photograph of a muscular man in a gym setting.

The screenshot shows a web browser window with the address bar displaying `https://localhost:44325/SignupForm.aspx`. The page features a "Create an Account" form overlay. The form has two columns of input fields: "Full Name" (placeholder: "Enter your full name"), "Age" (placeholder: "Enter your age"), "Weight (kg)" (placeholder: "Enter your weight in kg"), "Password" (placeholder: "Enter your password"), "Username" (placeholder: "Enter your username"), "Gender" (dropdown menu with "Select Gender"), "Height (cm)" (placeholder: "Enter your height in cm"), and "Confirm Password" (placeholder: "Re-enter your password"). There is a blue "Sign Up" button, a grey "Reset" button, and a link that says "Have Account Login". The background image is a Deadpool character.

This screenshot shows the same "Create an Account" form as the previous one, but with sample data entered into the fields. The "Full Name" field contains "Jatin Dryaneshware Gang", the "Age" field contains "20", the "Weight (kg)" field contains "58", the "Password" field contains "*****", the "Username" field contains "jatingangare", the "Gender" dropdown menu is set to "Male", and the "Height (cm)" field contains "164". The "Confirm Password" field also contains "*****". The "Sign Up" button is highlighted in blue, and the "Reset" button is grey. The "Have Account Login" link is visible at the bottom.

Create an Account

Full Name	Username
<input type="text" value="jatin @"/>	<input type="text" value="Enter your username"/>
Name should not contain numbers or symbols	
Age	Gender
<input type="text" value="Enter your age"/>	<input type="text" value="Select Gender"/>
Weight (kg)	Height (cm)
<input type="text" value="Enter your weight in kg"/>	<input type="text" value="Enter your height in cm"/>
Password	Confirm Password
<input type="text" value="Enter your password"/>	<input type="text" value="Re-enter your password"/>
<input type="button" value="Sign Up"/>	
<input type="button" value="Reset"/>	
Have Account Login	

Create an Account

Full Name	Username
<input type="text" value="Jatin Dnyaneshwar Ganga"/>	<input type="text" value="jatin_"/>
Username should only contain letters and numbers	
Age	Gender
<input type="text" value="Enter your age"/>	<input type="text" value="Select Gender"/>
Weight (kg)	Height (cm)
<input type="text" value="Enter your weight in kg"/>	<input type="text" value="Enter your height in cm"/>
Password	Confirm Password
<input type="text" value="Enter your password"/>	<input type="text" value="Re-enter your password"/>
<input type="button" value="Sign Up"/>	
<input type="button" value="Reset"/>	
Have Account Login	

Create an Account

Full Name	Username
<input type="text" value="Jatin Dnyaneshwar Ganga"/>	<input type="text" value="jatingangare22"/>
Age	Gender
<input type="text" value="jatin"/>	<input type="text" value="Select Gender"/>
Age should be a whole number	
Weight (kg)	Height (cm)
<input type="text" value="Enter your weight in kg"/>	<input type="text" value="Enter your height in cm"/>
Password	Confirm Password
<input type="text" value="Enter your password"/>	<input type="text" value="Re-enter your password"/>
<input type="button" value="Sign Up"/>	
<input type="button" value="Reset"/>	
Have Account Login	

Create an Account

Full Name	Username
<input type="text" value="Jatin Dnyaneshwar Ganga"/>	<input type="text" value="jatingangare22"/>
Age	Gender
<input type="text" value="20"/>	<input type="text" value="Select Gender"/>
Please select a gender	
Weight (kg)	Height (cm)
<input type="text" value="58"/>	<input type="text" value="164"/>
Password	Confirm Password
<input type="text" value="*****"/>	<input type="text" value="*****"/>
<input type="button" value="Sign Up"/>	
<input type="button" value="Reset"/>	
Have Account Login	

Create an Account

Full Name	Username
<input type="text" value="Jatin Dnyaneshwar Ganga"/>	<input type="text" value="jatingangare22"/>
Age	Gender
<input type="text" value="20"/>	<input type="text" value="Male"/>
Weight (kg)	Height (cm)
<input type="text" value="ss"/>	<input type="text" value="Enter your height in cm"/>
Weight should be a numeric value	
Password	Confirm Password
<input type="text" value="Enter your password"/>	<input type="text" value="Re-enter your password"/>
<input type="button" value="Sign Up"/>	
<input type="button" value="Reset"/>	
Have Account Login	

Create an Account

Full Name	Username
<input type="text" value="Jatin Dnyaneshwar Ganga"/>	<input type="text" value="jatingangare22"/>
Age	Gender
<input type="text" value="20"/>	<input type="text" value="Male"/>
Weight (kg)	Height (cm)
<input type="text" value="58"/>	<input type="text" value="jatin"/>
Height should be a numeric value	
Password	Confirm Password
<input type="text" value="Enter your password"/>	<input type="text" value="Re-enter your password"/>
<input type="button" value="Sign Up"/>	
<input type="button" value="Reset"/>	
Have Account Login	

Create an Account

Full Name	Username
<input type="text" value="Jatin Dnyaneshwar Ganga"/>	<input type="text" value="jatingangare22"/>
Age	Gender
<input type="text" value="20"/>	<input type="text" value="Male"/>
Weight (kg)	Height (cm)
<input type="text" value="58"/>	<input type="text" value="164"/>
Password	Confirm Password
<input type="text" value="Enter your password"/>	<input type="text" value="Re-enter your password"/>

Password is required

[Sign Up](#)

[Reset](#)

[Have Account Login](#)

Create an Account

Full Name	Username
<input type="text" value="Jatin Dnyaneshwar Ganga"/>	<input type="text" value="jatingangare22"/>
Age	Gender
<input type="text" value="20"/>	<input type="text" value="Male"/>
Weight (kg)	Height (cm)
<input type="text" value="58"/>	<input type="text" value="164"/>
Password	Confirm Password
<input type="text" value="....."/>	<input type="text" value="Re-enter your password"/>

Password must be at least 8 characters long

[Sign Up](#)

[Reset](#)

[Have Account Login](#)

Create an Account

Full Name	Username
<input type="text" value="Jatin Dnyaneshwar Ganga"/>	<input type="text" value="jatingangare22"/>
Age	Gender
<input type="text" value="20"/>	<input type="text" value="Male"/>
Weight (kg)	Height (cm)
<input type="text" value="58"/>	<input type="text" value="164"/>
Password	Confirm Password
<input type="text" value="....."/>	<input type="text" value="Re-enter your password"/>

Please confirm your password

[Sign Up](#)

[Reset](#)

[Have Account Login](#)

Create an Account

Full Name	Username
<input type="text" value="Jatin Dnyaneshwar Ganga"/>	<input type="text" value="jatingangare22"/>
Age	Gender
<input type="text" value="20"/>	<input type="text" value="Male"/>
Weight (kg)	Height (cm)
<input type="text" value="58"/>	<input type="text" value="164"/>
Password	Confirm Password
<input type="text" value="....."/>	<input type="text" value="....."/>

Passwords do not match

[Sign Up](#)

[Reset](#)

[Have Account Login](#)

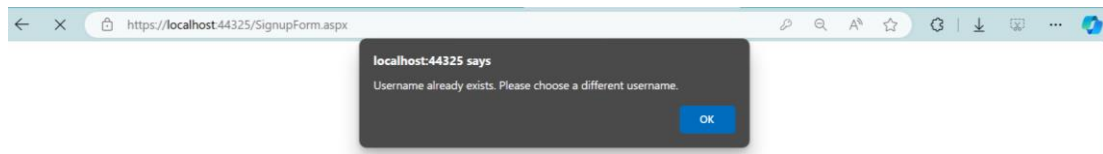
Create an Account

Full Name	Username
<input type="text" value="Jatin Dnyaneshwar Ganga"/>	<input type="text" value="jatingangare22"/>
Age	Gender
<input type="text" value="20"/>	<input type="text" value="Male"/>
Weight (kg)	Height (cm)
<input type="text" value="58"/>	<input type="text" value="164"/>
Password	Confirm Password
<input type="text" value="....."/>	<input type="text" value="....."/>

[Sign Up](#)

[Reset](#)

[Have Account Login](#)



A screenshot of a web browser window showing the "Create an Account" form. The form is centered on a background image of Deadpool. The form fields are as follows:

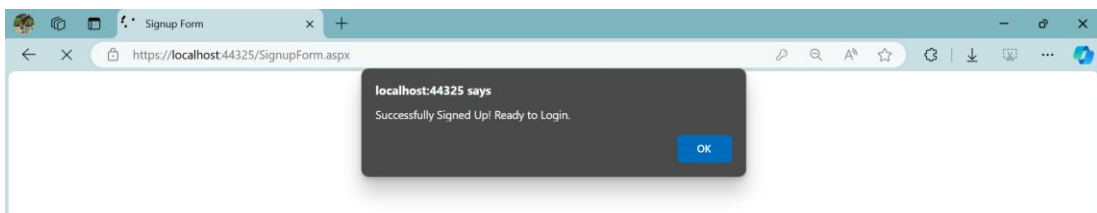
Full Name	Username
Jatin Dryaneshwar Ganga	jatingangare

Age	Gender
20	Male


Weight (kg)	Height (cm)
58	164

Password	Confirm Password
*****	*****

Buttons: "Sign Up" (blue), "Reset" (gray), and a link "Have Account Login" (blue).



A screenshot of a web browser window showing the "Log In" form. The form is centered on a background image of a muscular man. The form fields are as follows:

Log In

Username
Enter your username or email
Password
Enter your password
Login
Create an Account

Home x +

https://localhost:44325/HomeForm.aspx

Fitness Tracker **Add Workout** **Login**

Name	Age	Gender	Weight (kg)	Height (cm)	BMI	BMR
Jatin Dnyaneshwar Gangare	20	Male	50	164	18.59	1431.71

Home x +

https://localhost:44325/HomeForm.aspx

Fitness Tracker **Add Workout** **Login**

Name	Age	Gender	Weight (kg)	Height (cm)	BMI	BMR
Jatin Dnyaneshwar Gangare	20	Male	50	164	18.59	1431.71

Log Workout x +

https://localhost:44325/WorkoutForm.aspx

Log Workout

Workout Type

Duration (minutes)

Calories Burned

Workout Date

10-09-2024

Number of Exercises

Add Exercises

Save Workout Cancel

Log Workout x +

https://localhost:44325/WorkoutForm.aspx

Log Workout

Workout Type

Back And Biceps

Duration (minutes)

45

Calories Burned

225

Workout Date

10-09-2024

September, 2024

Su Mo Tu We Th Fr Sa

1 2 3 4 5 6 7

8 9 10 11 12 13 14

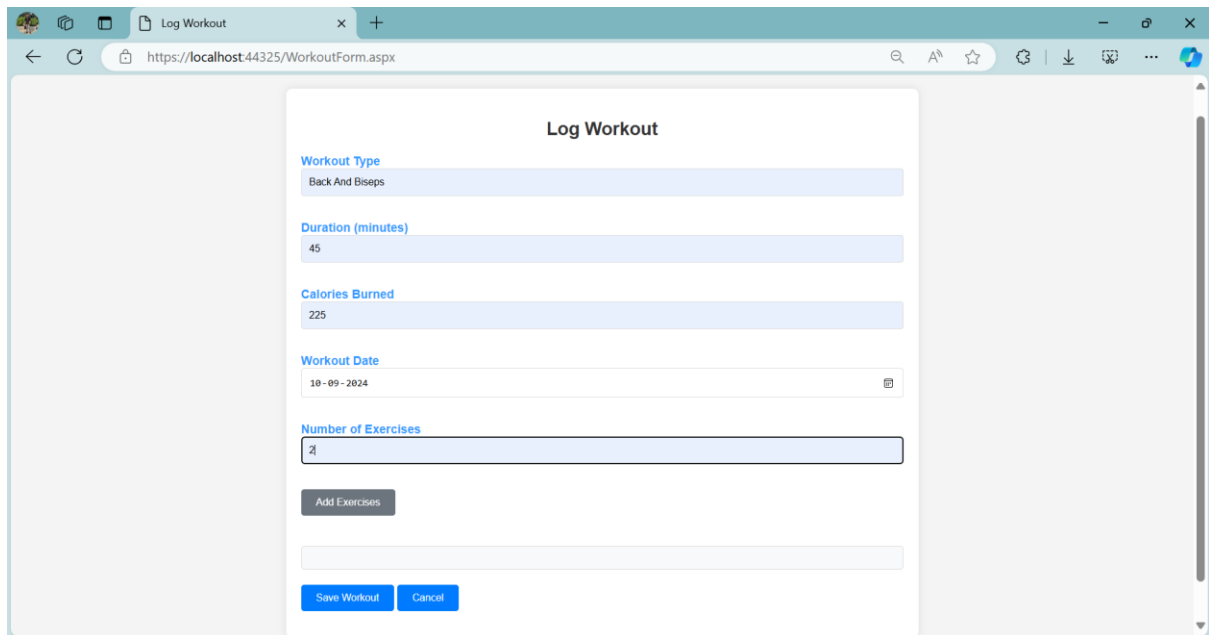
15 16 17 18 19 20 21

22 23 24 25 26 27 28

29 30 1 2 3 4 5

6 7 8 9 10 11 12

Clear Today



Log Workout

Workout Type
Back And Biceps

Duration (minutes)
45

Calories Burned
225

Workout Date
10-09-2024

Number of Exercises
2

Add Exercises

Save Workout Cancel

Number of Exercises

2

Add Exercises

Exercise Name:

Sets:

Reps:

Remove

Exercise Name:

Sets:

Reps:

Remove

Save Workout

Cancel

Number of Exercises

2

Add Exercises

Exercise Name:

row

Sets:

3

Reps:

12

Remove

Exercise Name:

latpull down

Sets:

4

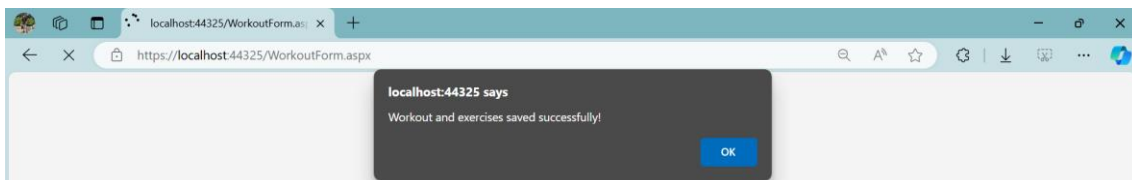
Reps:

20

Remove

Save Workout

Cancel



Home

https://localhost:44325/HomeForm.aspx

Logout

Fitness Tracker

Add Workout

Name	Age	Gender	Weight (kg)	Height (cm)	BMI	BMR
Jatin Dnyaneshwar Gangare	20	Male	50	164	18.59	1431.71

Workout Type	Duration (minutes)	Calories Burned	Workout Date	Exercises		
Back And Biceps	45	225	10-09-2024 00:00:00	Exercise Name	Sets	Reps
				row	3	12
				latpull down	4	20

Log Workout

https://localhost:44325/WorkoutForm.aspx

Exercise Name:
Dumbbell Press

Sets:
2

Reps:
12

Remove

Exercise Name:
Lateral Raises

Sets:
4

Reps:
15

Remove

Exercise Name:
Crunches

Sets:
3

Reps:
20

Remove

Save Workout

Cancel

Home

https://localhost:44325/HomeForm.aspx

Search

Address

Star

Refresh

Download

Extensions

More

Fitness Tracker

Add Workout

Logout

Name	Age	Gender	Weight (kg)	Height (cm)	BMI	BMR
Jatin Dnyaneshwar Gangare	20	Male	50	164	18.59	1431.71

Workout Type	Duration (minutes)	Calories Burned	Workout Date	Exercises		
Back And Biceps	45	225	10-09-2024 00:00:00	Exercise Name	Sets	Reps
				row	3	12
				latpull down	4	20
Shoulder and Abs	120	340	11-09-2024 00:00:00	Exercise Name	Sets	Reps
				Dumbell Press	2	12
				Lateral Raises	4	15
				Crunches	3	20

Conclusion :

The Fitness Tracker Web Application offers users an efficient and user-friendly platform to monitor their workouts, exercises, and personal fitness data such as BMI and BMR. The application's intuitive design and real-time updates make it easy for users to track their progress and stay motivated. By using ASP.NET Web Forms and C#, the application ensures a smooth and dynamic user experience, allowing for easy management of workout routines and personalized feedback.

With the integration of a SQL Server database, the application securely stores user information and workout logs. This robust database structure enables reliable data retrieval and storage, supporting comprehensive reporting and analysis. Features like dynamic exercise management make the system adaptable and personalized for each user, allowing for custom workout plans and efficient tracking of fitness goals.

While the application fulfills its core purpose, there is potential for further improvements, such as enhancing the user interface with more engaging graphics and interactive elements. Adding more fitness tracking features, such as integration with wearable devices or advanced analytics, and expanding support for mobile devices could further enhance the user experience and accessibility.

Overall, this project demonstrates the effective use of web development techniques to create a functional and practical fitness tracking tool. It not only serves its purpose but also provides a solid foundation for future growth and enhancements, setting the stage for advanced features and broader user engagement.