Task 1:

Solution:

Controller Code:

namespace Practice.Controllers

{

public class UserController : Controller

{

public ActionResult Login()

{

return View();

}

[HttpPost]

public ActionResult Login(UserDetail model)

{

if (ModelState.IsValid)

{

UserDetail user = UserDetail.GetUserByUsernameAndPassword(model.Username, model.Password);

if (user != null)

{

Session["UserID"] = user.ID;

return RedirectToAction("Profile");

}

else

{

ModelState.AddModelError("", "Invalid username or password");

}

}

return View();

}

public ActionResult Profile()

{

try

{

int userID = (int)Session["UserID"];

UserDetail user = UserDetail.GetUserByID(userID);

if (user != null)

{

return View(user);

}

else

{

return RedirectToAction("Login");

}

}

catch (Exception ex)

{

// Log the exception or handle it as appropriate for your application

return View("Error");

}

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Logout()

{

// Clear the session

Session.Clear();

Session.Abandon();

// Redirect to the login page

return RedirectToAction("Login", "User");

}

}

}

Model Code:

namespace Practice.Models

{

public class UserDetail

{

public int ID { get; set; }

public string FirstName { get; set; }

public string LastName { get; set; }

public string Email { get; set; }

public string Username { get; set; }

public string Password { get; set; }

public static UserDetail GetUserByUsernameAndPassword(string username, string password)

{

UserDetail user = null;

string conn = "Data Source=DESKTOP-27TFO05;Initial Catalog=UserData;Integrated Security=True";

string connectionString = "Data Source = DESKTOP-27TFO05; Initial Catalog = UserData; Integrated Security = True; ";

string query = "Select \* from Users where username=@Username and password=@Password";

using (SqlConnection connection = new SqlConnection(conn))

{

connection.Open();

SqlCommand command = new SqlCommand(query, connection);

command.Parameters.AddWithValue("@Username", username);

command.Parameters.AddWithValue("@Password", password);

SqlDataReader reader = command.ExecuteReader();

if (reader.Read())

{

user = new UserDetail();

user.ID = (int)reader["ID"];

user.FirstName = reader["FirstName"].ToString();

user.LastName = reader["LastName"].ToString();

user.Email = reader["Email"].ToString();

user.Username = reader["Username"].ToString();

user.Password = reader["Password"].ToString();

}

reader.Close();

}

return user;

}

public static UserDetail GetUserByID(int id)

{

UserDetail user = null;

string connectionString = "Data Source = DESKTOP-27TFO05; Initial Catalog = UserData; Integrated Security = True; ";

string query = "SELECT \* FROM Users WHERE ID = @ID";

using (SqlConnection connection = new SqlConnection(connectionString))

{

connection.Open();

SqlCommand command = new SqlCommand(query, connection);

command.Parameters.AddWithValue("@ID", id);

SqlDataReader reader = command.ExecuteReader();

if (reader.Read())

{

user = new UserDetail();

user.ID = (int)reader["ID"];

user.FirstName = reader["FirstName"].ToString();

user.LastName = reader["LastName"].ToString();

user.Email = reader["Email"].ToString();

user.Username = reader["Username"].ToString();

user.Password = reader["Password"].ToString();

}

reader.Close();

}

return user;

}

}

}

View Code:

Login:

@model Test\_MVC\_No\_Entity.Models.UserDetail

@{

/\*\*/

ViewBag.Title = "Login";

}

<h2>Login</h2>

@using (Html.BeginForm("Login", "User", FormMethod.Post))

{

@Html.AntiForgeryToken()

<div class="form-group">

@Html.LabelFor(m => m.Username)

@Html.TextBoxFor(m => m.Username)

</div>

<div class="form-group">

@Html.LabelFor(m => m.Password)

@Html.PasswordFor(m => m.Password)

</div>

@Html.ValidationSummary()

<button class="btn-primary" type="submit">Login</button>

}

Profile:

@model Test\_MVC\_No\_Entity.Models.UserDetail

@{

ViewBag.Title = "ProfilePage";

}

@if (Session["id"] != null)

{

<h1>Hey @Model.Firstname @Model.Lastname</h1>

<h2>Email: @Model.Username</h2>

<h2>Email: @Model.Email</h2>

using (Html.BeginForm("Logout", "User", FormMethod.Post))

{

<button type="submit">Logout</button>

}

}

else

{

<h1>You are not logged in</h1>

using (Html.BeginForm("Logout", "User", FormMethod.Post))

{

<button type="submit">Go Back</button>

}

}