Task 1:

Solution:

Controller:

namespace Test\_MVC\_Entity.Controllers

{

public class UserController : Controller

{

// GET: User

public ActionResult Login()

{

return View();

}

[HttpPost]

[ValidateAntiForgeryToken]

public ActionResult Login(User model)

{

if (ModelState.IsValid)

{

using (var db = new UserDataEntities())

{

var user = db.Users.SingleOrDefault(u => u.Username == model.Username && u.Password == model.Password);

if (user != null)

{

Session["UserId"] = user.ID;

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

}

else

{

ModelState.AddModelError("", "Username or Password mismatch");

}

}

}

return View(model);

}

public ActionResult UserProfile()

{

try

{

int id = (int)Session["UserId"];

using(var db=new UserDataEntities())

{

var user = db.Users.SingleOrDefault(m => m.ID == id);

if (user != null)

{

return View(user);

}

}

}

catch(Exception ex)

{

return View("Error");

}

return View();

}

public ActionResult Logout()

{

Session.Clear();

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

}

}

}

View:

Login:

@model Test\_MVC\_Entity.Models.User

@{

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 type="submit">Login</button>

}

Profile:

@model Test\_MVC\_Entity.Models.User

@{

ViewBag.Title = "UserProfile";

}

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

{

<h2>UserProfile</h2>

<h1>Hey @Model.FirstName @Model.LastName</h1>

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

{

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

}

}

else

{

<h3>You are not logged in yet!</h3>

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

{

<button type="submit">Go to Login Page</button>

}

}

Task 2:

Controller Code:

namespace Test\_MVC\_Entity.Controllers

{

public class CRUDController : Controller

{

private UserDataEntities db = new UserDataEntities();

// GET: CRUD

public ActionResult Index()

{

List<User> user = db.Users.ToList();

return View(user);

}

[HttpPost]

public ActionResult Index(User model,string action)

{

if (ModelState.IsValid)

{

if (action == "Create")

{

db.Users.Add(model);

db.SaveChanges();

}

else if (action == "Update")

{

var user = db.Users.SingleOrDefault(u => u.Username == model.Username);

if (user != null)

{

user.FirstName = model.FirstName;

user.LastName = model.LastName;

user.Email = model.Email;

user.Password = model.Password;

db.SaveChanges();

}

else

{

ModelState.AddModelError("", "No user found");

}

}

else if (action == "Delete")

{

var user = db.Users.SingleOrDefault(u => u.Username == model.Username);

if (user != null)

{

db.Users.Remove(user);

db.SaveChanges();

}

else

{

ModelState.AddModelError("", "No user found");

}

}

}

List<User> users = db.Users.ToList();

return View(users);

}

}

}

View Code:

@model List<Test\_MVC\_Entity.Models.User>

@{

ViewBag.Title = "Index";

}

<h2>Index</h2>

@using (Html.BeginForm("Index", "CRUD", FormMethod.Post))

{

<div>

<label>FirstName: </label>

<input type="text" name="FirstName" />

</div>

<div>

<label>LastName: </label>

<input type="text" name="LastName" />

</div>

<div>

<label>Username: </label>

<input type="text" name="Username" />

</div>

<div>

<label>Email: </label>

<input type="text" name="Email" />

</div>

<div>

<label>Password: </label>

<input type="text" name="Password" />

</div>

<div>

<button type="submit" name="action" value="Create">Create</button>

<button type="submit" name="action" value="Update">Update</button>

<button type="submit" name="action" value="Delete">Delete</button>

</div>

}

<table class="table">

<thead>

<tr>

<th>ID</th>

<th>FirstName</th>

<th>LastName</th>

<th>Username</th>

<th>Email</th>

<th>Password</th>

</tr>

</thead>

<tbody>

@foreach (var user in Model)

{

<tr>

<td>@user.ID</td>

<td>@user.FirstName</td>

<td>@user.LastName</td>

<td>@user.Username</td>

<td>@user.Email</td>

<td>@user.Password</td>

</tr>

}

</tbody>

</table>