

## Database Connectivity Program

### Userdemo.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.ComponentModel.DataAnnotations;
namespace WebApplication4.Models
{
    public class Userdemo
    {
        [Required]
        public int UserdemoId { get; set; }
        [Required]
        [Display(Name = "username:")]
        [DataType(DataType.EmailAddress)]
        public String Username { get; set; }
        [Required]
        [Display(Name="password:")]
        [DataType(DataType.Password)]
        public String Password { get; set; }
    }
}
```

### Userdemocontext.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Data.Entity;

namespace WebApplication4.Models
{
    public class Userdemocontext : DbContext
    {
        public Userdemocontext()
            : base("mycon")
        {
            DropCreateDatabaseIfModelChanges<Userdemocontext> d = new
DropCreateDatabaseIfModelChanges<Userdemocontext>();
            Database.SetInitializer(d);
        }
        public DbSet<Userdemo> usersdemo { get; set; }
    }
}
```

### Userdemocontroller.cs

```
using System;
```

```

using System.Collections.Generic;
using System.Data;
using System.Data.Entity;
using System.Linq;
using System.Net;
using System.Web;
using System.Web.Mvc;
using WebApplication4.Models;

namespace WebApplication4.Controllers
{
    public class UserdemoController : Controller
    {
        private Userdemocontext db = new Userdemocontext();

        // GET: /Userdemo/
        public ActionResult Index()
        {
            return View(db.usersdemo.ToList());
        }

        // GET: /Userdemo/Details/5
        public ActionResult Details(int? id)
        {
            if (id == null)
            {
                return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
            }
            Userdemo userdemo = db.usersdemo.Find(id);
            if (userdemo == null)
            {
                return HttpNotFound();
            }
            return View(userdemo);
        }

        // GET: /Userdemo/Create
        public ActionResult Create()
        {
            return View();
        }

        // POST: /Userdemo/Create
        // To protect from overposting attacks, please enable the specific properties you
        // want to bind to, for
        // more details see http://go.microsoft.com/fwlink/?LinkId=317598.
        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult Create([Bind(Include="UserdemoId,Username,Password")]
            Userdemo userdemo)
        {
            if (ModelState.IsValid)
            {
                db.usersdemo.Add(userdemo);
                db.SaveChanges();
                return RedirectToAction("Index");
            }
        }
    }
}

```

```

        return View(userdemo);
    }

    // GET: /Userdemo/Edit/5
    public ActionResult Edit(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        Userdemo userdemo = db.usersdemo.Find(id);
        if (userdemo == null)
        {
            return HttpNotFound();
        }
        return View(userdemo);
    }

    // POST: /Userdemo/Edit/5
    // To protect from overposting attacks, please enable the specific properties you
    want to bind to, for
    // more details see http://go.microsoft.com/fwlink/?LinkId=317598.
    [HttpPost]
    [ValidateAntiForgeryToken]
    public ActionResult Edit([Bind(Include="UserdemoId,Username,Password")] Userdemo
userdemo)
    {
        if (ModelState.IsValid)
        {
            db.Entry(userdemo).State = EntityState.Modified;
            db.SaveChanges();
            return RedirectToAction("Index");
        }
        return View(userdemo);
    }

    // GET: /Userdemo/Delete/5
    public ActionResult Delete(int? id)
    {
        if (id == null)
        {
            return new HttpStatusCodeResult(HttpStatusCode.BadRequest);
        }
        Userdemo userdemo = db.usersdemo.Find(id);
        if (userdemo == null)
        {
            return HttpNotFound();
        }
        return View(userdemo);
    }

    // POST: /Userdemo/Delete/5
    [HttpPost, ActionName("Delete")]
    [ValidateAntiForgeryToken]
    public ActionResult DeleteConfirmed(int id)
    {
        Userdemo userdemo = db.usersdemo.Find(id);
        db.usersdemo.Remove(userdemo);
    }

```

```

        db.SaveChanges();
        return RedirectToAction("Index");
    }

    protected override void Dispose(bool disposing)
    {
        if (disposing)
        {
            db.Dispose();
        }
        base.Dispose(disposing);
    }
}

```

## Web.config

```

<?xml version="1.0" encoding="utf-8"?>
<!--
    For more information on how to configure your ASP.NET application, please visit
    http://go.microsoft.com/fwlink/?LinkId=301880
-->
<configuration>
    <configSections>
        <!-- For more information on Entity Framework configuration, visit
        http://go.microsoft.com/fwlink/?LinkId=237468 -->
        <section name="entityFramework"
            type="System.Data.Entity.Internal.ConfigFile.EntityFrameworkSection, EntityFramework,
            Version=6.0.0.0, Culture=neutral, PublicKeyToken=b77a5c561934e089"
            requirePermission="false" />
    </configSections>
    <connectionStrings>
        <add name="DefaultConnection" connectionString="Data
Source=(LocalDb)\v11.0;AttachDbFilename=|DataDirectory|\aspnet-WebApplication4-
20190130023535.mdf;Initial Catalog=aspnet-WebApplication4-20190130023535;Integrated
Security=True"
            providerName="System.Data.SqlClient" />

        <add name="mycon" connectionString="Data Source=(localdb)\Projects;Initial
Catalog=demodb;Integrated Security=True"
            providerName="System.Data.SqlClient" />
    </connectionStrings>
    <appSettings>
        <add key="webpages:Version" value="3.0.0.0" />
        <add key="webpages:Enabled" value="false" />
        <add key="ClientValidationEnabled" value="true" />
        <add key="UnobtrusiveJavaScriptEnabled" value="true" />
    </appSettings>
    <system.web>
        <authentication mode="None" />
        <compilation debug="true" targetFramework="4.5" />
        <httpRuntime targetFramework="4.5" />
    </system.web>
    <system.webServer>
        <modules>
            <remove name="FormsAuthenticationModule" />
        </modules>
    </system.webServer>

```

```

<runtime>
  <assemblyBinding xmlns="urn:schemas-microsoft-com:asm.v1">
    <dependentAssembly>
      <assemblyIdentity name="System.Web.Helpers" publicKeyToken="31bf3856ad364e35" />
      <bindingRedirect oldVersion="1.0.0.0-3.0.0.0" newVersion="3.0.0.0" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="System.Web.Mvc" publicKeyToken="31bf3856ad364e35" />
      <bindingRedirect oldVersion="1.0.0.0-5.0.0.0" newVersion="5.0.0.0" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="System.Web.WebPages" publicKeyToken="31bf3856ad364e35" />
      <bindingRedirect oldVersion="1.0.0.0-3.0.0.0" newVersion="3.0.0.0" />
    </dependentAssembly>
    <dependentAssembly>
      <assemblyIdentity name="WebGrease" publicKeyToken="31bf3856ad364e35" />
      <bindingRedirect oldVersion="1.0.0.0-1.5.2.14234" newVersion="1.5.2.14234" />
    </dependentAssembly>
  </assemblyBinding>
</runtime>
<entityFramework>
  <defaultConnectionFactory
type="System.Data.Entity.Infrastructure.SqlConnectionFactory, EntityFramework" />
  <providers>
    <provider invariantName="System.Data.SqlClient"
type="System.Data.Entity.SqlServer.SqlProviderServices, EntityFramework.SqlServer" />
  </providers>
</entityFramework>
</configuration>

```

## Rootconfig.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using System.Web.Routing;

namespace WebApplication4
{
    public class RouteConfig
    {
        public static void RegisterRoutes(RouteCollection routes)
        {
            routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

            routes.MapRoute(
                name: "Default",
                url: "{controller}/{action}/{id}",
                defaults: new { controller = "Userdemo", action = "Index", id =
UrlParameter.Optional }
            );
        }
    }
}

```