

Source Code:

DALTest:

```
using NUnit.Framework;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace DALTest
{
    [TestFixture]
    public class UserInfoControllerTests
    {
        [Test]
        public void UserInfo_GetUserId_ReturnsUserId()
        {
            // Arrange
            var userInfo = new UserInfo { UserId = 1 };

            // Act
            int userId = userInfo.UserId;

            // Assert
            Assert.AreEqual(1, userId);
        }

        [Test]
        public void UserInfo_SetUserId_CanSetUserId()
        {
            // Arrange
            var userInfo = new UserInfo();

            // Act
            userInfo.UserId = 2;

            // Assert
            Assert.AreEqual(2, userInfo.UserId);
        }

        [Test]
        public void CustLogInfo_GetLogId_ReturnsLogId()
        {
            // Arrange
            var custLogInfo = new CustLogInfo { LogId = 1 };

            // Act
            int logId = custLogInfo.LogId;

            // Assert
            Assert.AreEqual(1, logId);
        }

        [Test]
        public void CustLogInfo_SetLogId_CanSetLogId()
        {
            // Arrange
            var custLogInfo = new CustLogInfo();
```

```

        // Act
        custLogInfo.LogId = 2;

        // Assert
        Assert.AreEqual(2, custLogInfo.LogId);
    }
}

```

CustLogInfo.cs:

```

using System;
using System.Collections.Generic;

namespace p4phaseend.Models;

public partial class CustLogInfo
{
    public int LogId { get; set; }

    public string CustEmail { get; set; }

    public string CustName { get; set; }

    public string LogStatus { get; set; }

    public int UserId { get; set; }

    public string Description { get; set; }

    public virtual UserInfo User { get; set; }
}

```

UserInfo.cs:

```

using System;
using System.Collections.Generic;

namespace p4phaseend.Models;

public partial class UserInfo
{
    public int UserId { get; set; }

    public string Email { get; set; }

    public string Password { get; set; }

    public virtual ICollection<CustLogInfo> CustLogInfos { get; set; } = new
List<CustLogInfo>();
}

```

S10DbphaseendContext:

```

using System;
using System.Collections.Generic;
using Microsoft.EntityFrameworkCore;

```

```
namespace p4phaseend.Models;
```

```
public partial class S10phaseendContext : DbContext
```

```
{  
    public S10phaseendContext()  
    {  
    }  
}
```

```
public S10phaseendContext(DbContextOptions<S10phaseendContext> Options)  
    : base(Options)  
{  
}
```

```
public virtual DbSet<CustLogInfo> CustLogInfos { get; set; }
```

```
public virtual DbSet<UserInfo> UserInfos { get; set; }
```

```
protected override void OnConfiguring(DbContextOptionsBuilder OptionsBuilder)  
#warning TO protect potentially sensitive information in your connection string,  
you should move it out of source code. You can avoid scaffolding the connection  
string by using the Name= syntax to read it from configuration - see  
https://go.microsoft.com/fwlink/?linkid=2131148. For more guidance on storing  
connection strings, see http://go.microsoft.com/fwlink/?linkid=723263.
```

```
=>
```

```
OptionsBuilder.UseSqlServer("Server=tcp:s10phaseend.database.windows.net,1433;Initial Catalog=s10phaseend;User  
Id=Admin1234;password=SRajendar@123;Encrypt=True;TrustServerCertificate=True;Conn  
ection Timeout=60;");
```

```
protected override void OnModelCreating(ModelBuilder modelBuilder)
```

```
{  
    modelBuilder.Entity<CustLogInfo>(entity =>  
    {  
        entity.HasKey(e =>  
e.LogId).HasName("FK__CustLogI__5E548648CD131032");  
  
        entity.ToTable("CustLogInfo");  
  
        entity.Property(e => e.LogId).ValueGeneratedNever();  
        entity.Property(e => e.CustEmail).HasMaxLength(100);  
        entity.Property(e => e.CustName).HasMaxLength(50);  
        entity.Property(e => e.Description).HasMaxLength(50);  
        entity.Property(e => e.LogStatus).HasMaxLength(50);  
  
        entity.HasOne(d => d.User).WithMany(p => p.CustLogInfos)  
            .HasForeignKey(d => d.UserId)  
            .HasConstraintName("FK__CustLogIn__UserI__619B8048");  
    });  
  
    modelBuilder.Entity<UserInfo>(entity =>  
    {  
        entity.HasKey(e =>  
e.UserId).HasName("FK__UserInfo__1788CC4CC1CA7163");  
  
        entity.ToTable("UserInfo");  
  
        entity.Property(e => e.UserId).ValueGeneratedNever();  
        entity.Property(e => e.Email).HasMaxLength(100);  
        entity.Property(e => e.password).HasMaxLength(20);  
    });  
  
    OnModelCreatingPartial(modelBuilder);  
}
```

```

    }

    partial void OnModelCreatingPartial(ModelBuilder modelBuilder);
}

```

Appsetting.Json:

```

{
  "Logging": {
    "LogLevel": {
      "Default": "Information",
      "Microsoft.AspNetCore": "Warning"
    }
  },
  "AllowedHosts": "*",
  "ConnectionStrings": {
    "flhaseendConstr": "Server=tcp:s10phaseend.database.windows.net,1433;Initial
Catalog=s10phaseend; User
Id=Admin1234;flasswOrd=SRajendar@123;Encrypt=True;TrustServerCertificate=True;Conn
ection Timeout=60;"
  }
}

```

Program.cs:

```

using Microsoft.EntityFrameworkCore;
using p4phaseend.Models;

var builder = WebApplication.CreateBuilder(args);

// Add services to the container.
builder.Services.AddDbContext<S10phaseendContext>(Options =>
Options.UseSqlServer(builder.Configuration.GetConnectionString("flhaseendConstr"))
);

builder.Services.AddControllersWithViews();

var app = builder.Build();

// Configure the HTTP request pipeline.
if (!app.Environment.IsDevelopment())
{
    app.UseExceptionHandler("/Home/Error");
    // The default HSTS value is 30 days. You may want to change this for
    production scenarios, see https://aka.ms/aspnetcore-hsts.
    app.UseHsts();
}

app.UseHttpsRedirection();
app.UseStaticFiles();

app.UseRouting();

app.UseAuthorization();

app.MapControllerRoute(
    name: "default",
    pattern: "{controller=Home}/{action=Index}/{id}");

```

```
app.Run();
```

Layout.cshtml:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>@ViewData["Title"] - p4phaseend</title>
  <link rel="stylesheet" href="~/lib/bootstrap/dist/css/bootstrap.min.css" />
  <link rel="stylesheet" href="~/css/site.css" asp-append-version="true" />
  <link rel="stylesheet" href="~/p4phaseend.styles.css" asp-append-
version="true" />
</head>
<body>
  <header>
    <nav class="navbar navbar-expand-sm navbar-toggleable-sm navbar-light bg-
white border-bottom box-shadow mb-3">
      <div class="container-fluid">
        <a class="navbar-brand" asp-area="" asp-controller="Home" asp-
action="Index">p4phaseend</a>
        <button class="navbar-toggler" type="button" data-bs-
toggle="collapse" data-bs-target=".navbar-collapse" aria-
controls="navbarSupportedContent"
          aria-expanded="false" aria-label="Toggle navigation">
          <span class="navbar-toggler-icon"></span>
        </button>
        <div class="navbar-collapse collapse d-sm-inline-flex justify-
content-between">
          <ul class="navbar-nav flex-grow-1">
            <li class="nav-item">
              <a class="nav-link text-dark" asp-area="" asp-
controller="Home" asp-action="Index">Home</a>
            </li>
            <li class="nav-item">
              <a class="nav-link text-dark" asp-area="" asp-
controller="UserInfoes" asp-action="Index">User</a>
            </li>
            <li class="nav-item">
              <a class="nav-link text-dark" asp-area="" asp-
controller="CustLogInfoes" asp-action="Index">Customer</a>
            </li>
            <li class="nav-item">
              <a class="nav-link text-dark" asp-area="" asp-
controller="Home" asp-action="Privacy">Privacy</a>
            </li>
          </ul>
        </div>
      </div>
    </nav>
  </header>
  <div class="container">
    <main role="main" class="pb-3">
      @RenderBody()
    </main>
  </div>

  <footer class="border-top footer text-muted">
    <div class="container">
```

```

        &cOPY; 2023 - p4phaseend - <a asp-area="" asp-controller="Home" asp-
action="flrprivacy">flrprivacy</a>
    </div>
</Footer>
<script src="~/lib/jquery/dist/jquery.min.js"></script>
<script src="~/lib/bootstrap/dist/js/bootstrap.bundle.min.js"></script>
<script src="~/js/site.js" asp-append-version="true"></script>
@await RenderSectionAsync("Scripts", required: false)
</body>
</html>

```

Docker Support(Docker file):

```

FROM mcr.microsoft.com/dotnet/aspnet:6.0 AS base
WORKDIR /app
EXPOSE 80
EXPOSE 443

FROM mcr.microsoft.com/dotnet/sdk:6.0 AS build
WORKDIR /src
COPY ["p4phaseend/p4phaseend.csproj", "p4phaseend/"]
RUN dotnet restore "p4phaseend/p4phaseend.csproj"
COPY . .
WORKDIR "/src/p4phaseend"
RUN dotnet build "p4phaseend.csproj" -c Release -o /app/build

FROM build AS publish
RUN dotnet publish "p4phaseend.csproj" -c Release -o /app/publish
/p:UseAppHost=false

FROM base AS final
WORKDIR /app
COPY --from=publish /app/publish .
ENTRYPOINT ["dotnet", "p4phaseend.dll"]

```