

*Projekt*

**Krijimi i një projekti me temë "HR script SQL" duke përdorur Visual Studio**

**Lënda:C#**

**Punoi:Kiara Ymeraj**

**Gr.A**

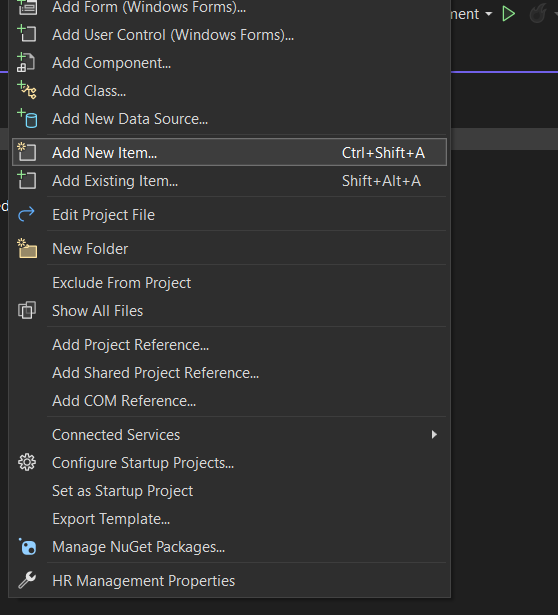


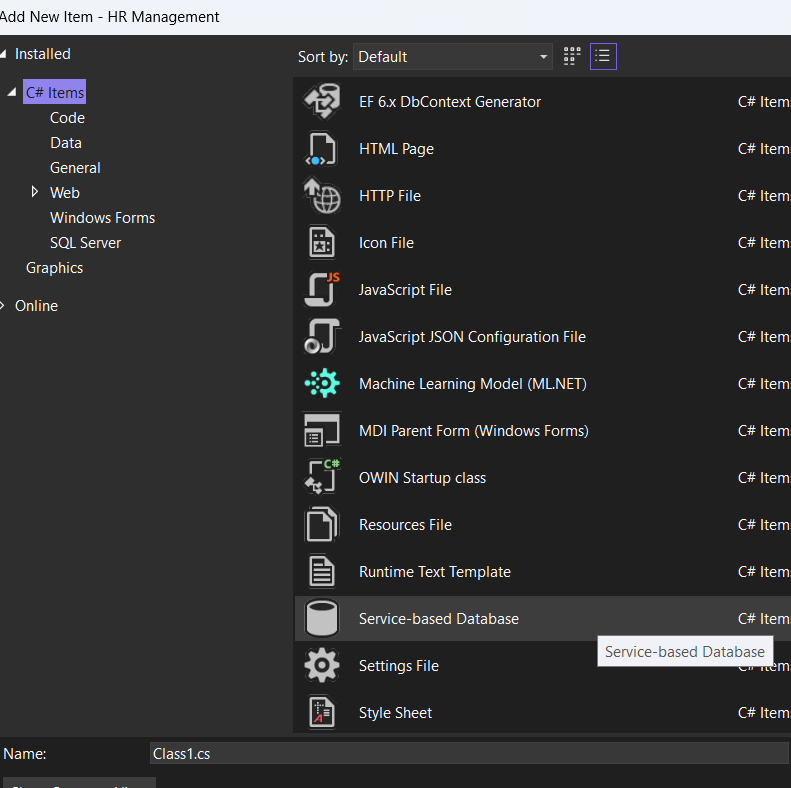
Hyrje (Qëllimi i projektit)

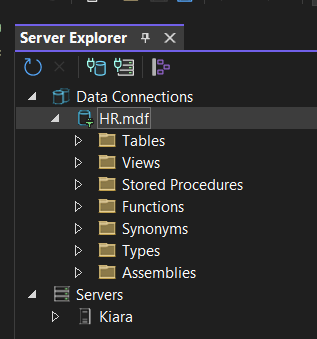
Ky projekt ka për qëllim zhvillimin e një sistemi të menaxhimit të burimeve njerëzore (HRM) duke përdorur SQL për menaxhimin e bazës së të dhënave dhe Visual Studio për zhvillimin e ndërfaqes së përdoruesit. Sistemi do të lehtësojë proceset e menaxhimit të të dhënave të punonjësve dhe departamenteve. Ky sistem do të mundësojë organizatave të ruajnë të dhëna të sakta dhe të përditësuara, të kryejnë analiza të avancuara dhe të optimizojnë menaxhimin e burimeve njerëzore.

Krijimi i projektit

Gjatë zhvillimit të projektit tim për sistemin e menaxhimit të burimeve njerëzore (HRM), fillova duke krijuar një bazë të dhënash në SQL Server. E ruajta këtë bazë të dhënash në një skedar **.mdf** për ta bërë më të lehtë menaxhimin dhe aksesimin. Pasi krijova databazën, klikova me të djathtën në emërtimin e tij në Solution Explorer dhe zgjodha Open, për të hapur tab-in Server Explorer.

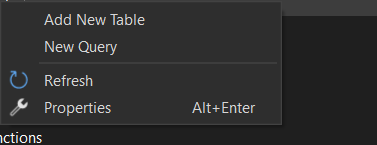


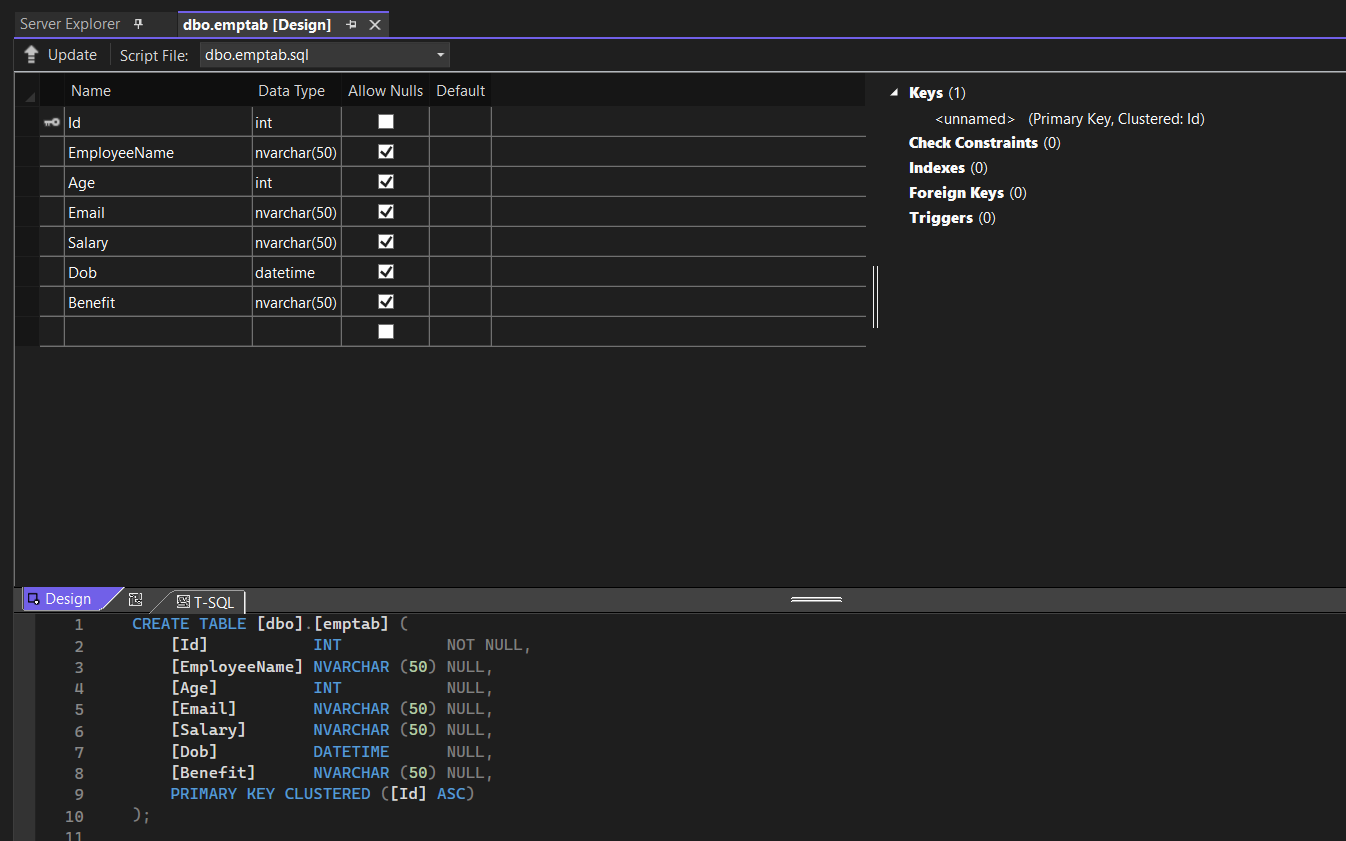


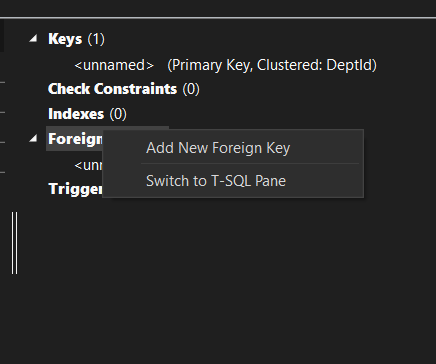


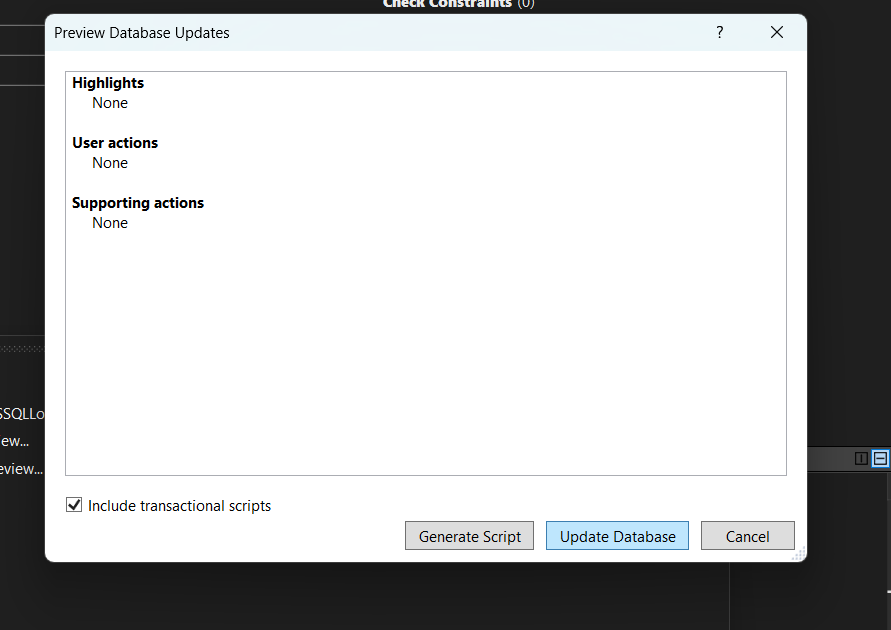
Krijimi i tabelave

Fillova duke krijuar tabelën **Employees** në bazën e të dhënave. Pasi klikova “Add New table” që e hapa duke klikuar te Tables ne skedarin .mdf shkrova një skript për krijimin e tabelës **Employees** që përfshinte kolonat **Id**, **EmployeeName**, **Age**, **Email**, **Salary**, **Dob**, dhe **Benefit**. Çelësi primar i tabelës ishte kolona **Id**, e cila siguron që çdo punonjës të ketë një identifikues unik. Pasi përfundova krijimin e tabelës, përditësova bazën e të dhënave dhe ruajta skedarin **.mdf** për të ruajtur ndryshimet.



Më pas krijova tabelën **Departments** në bazën e të dhënave. Shkrova një skript për krijimin e tabelës **Departments** që përfshinte kolonat **DeptId**, **DeptName** dhe **ManagerId**. Kolona **DeptId** u përcaktua si çelësi primar, duke siguruar që çdo departament të ketë një identifikues unik. Përveç kësaj, vendosa një çelës të huaj (foreign key) në kolonën **ManagerId**, e cila referon kolonën **Id** në tabelën **Employees**, duke krijuar një lidhje midis menaxherëve dhe departamenteve. Për të shtuar një çelës të jashtëm, klikova me të djathtin e mouse-it mbi Foreign Key dhe zgjodha nga sub-menu opsionin Add Ne Foreign Key. Pasi gjithe kodi im është i saktë e ruajta atë duke klikuar në butonin Update.



Dizenjimi i WinForms

Employee form

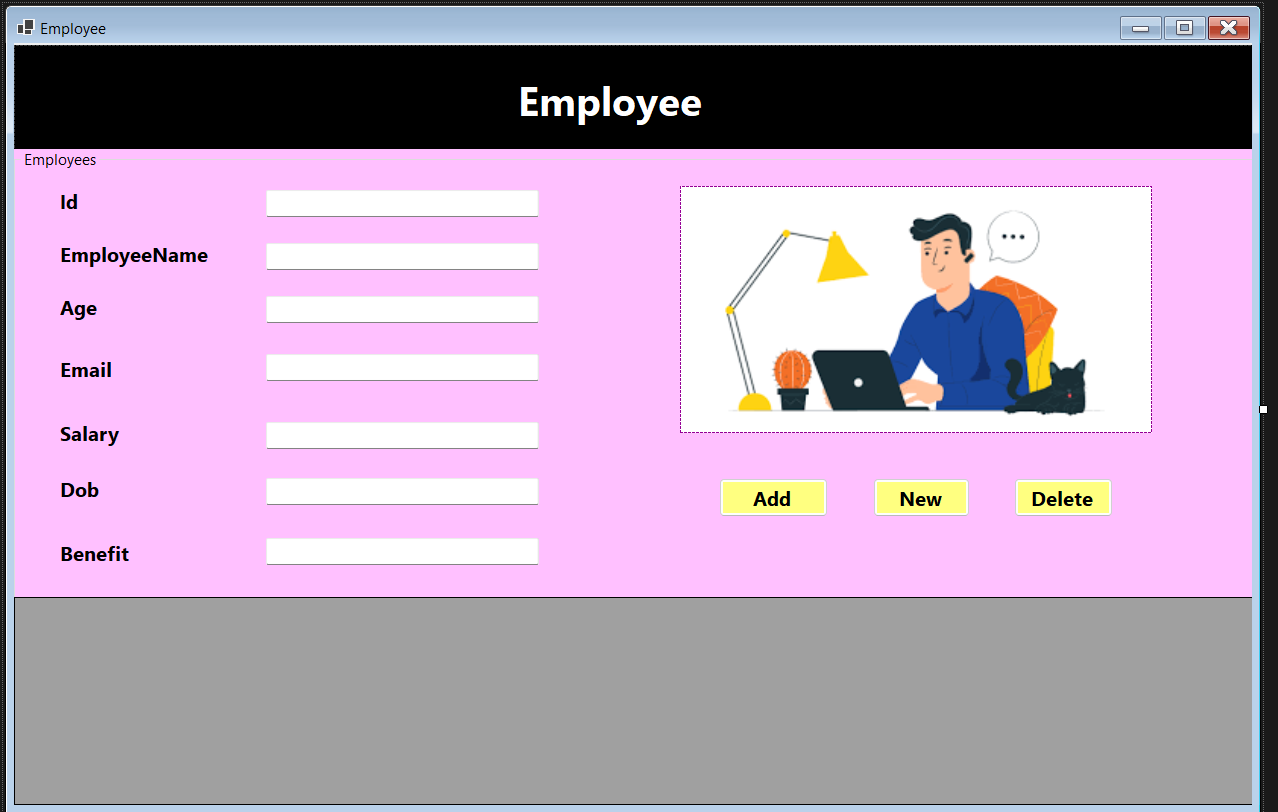
Kam krijuar një formë për të menaxhuar informacionin e punonjësve të një kompanie. Forma përfshin fusha për emrin e punonjësit, moshën, email-in, pagën, datën e lindjes dhe përfitimet. Për të manipuluar me formën, përdora butonat “Add” “New" dhe "Delete" për të kontrolluar punonjësit. Gjeta këto komanda në kutinë e veglave Toolbox të ofruara në ndërfaqen e krijuesit të formave.

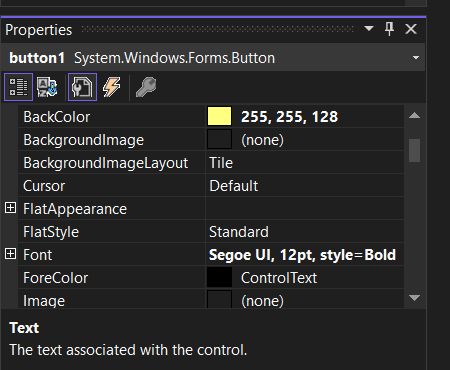
Për të krijuar formën `Employee`, përdora disa komanda dhe komponentë të ndryshëm nga kutia e veglave (Toolbox) në Visual Studio. Fillimisht, shtova komponentët `TextBox` për të hyrë emrin e punonjësit, moshën, email-in, pagën, dhe përfitimet, si dhe një `DateTimePicker` për të zgjedhur datën e lindjes. Për të afishuar dhe manipuluar të dhënat, përdora një `DataGridView` që mundëson shikimin dhe redaktimin e listës së punonjësve në kohë reale.

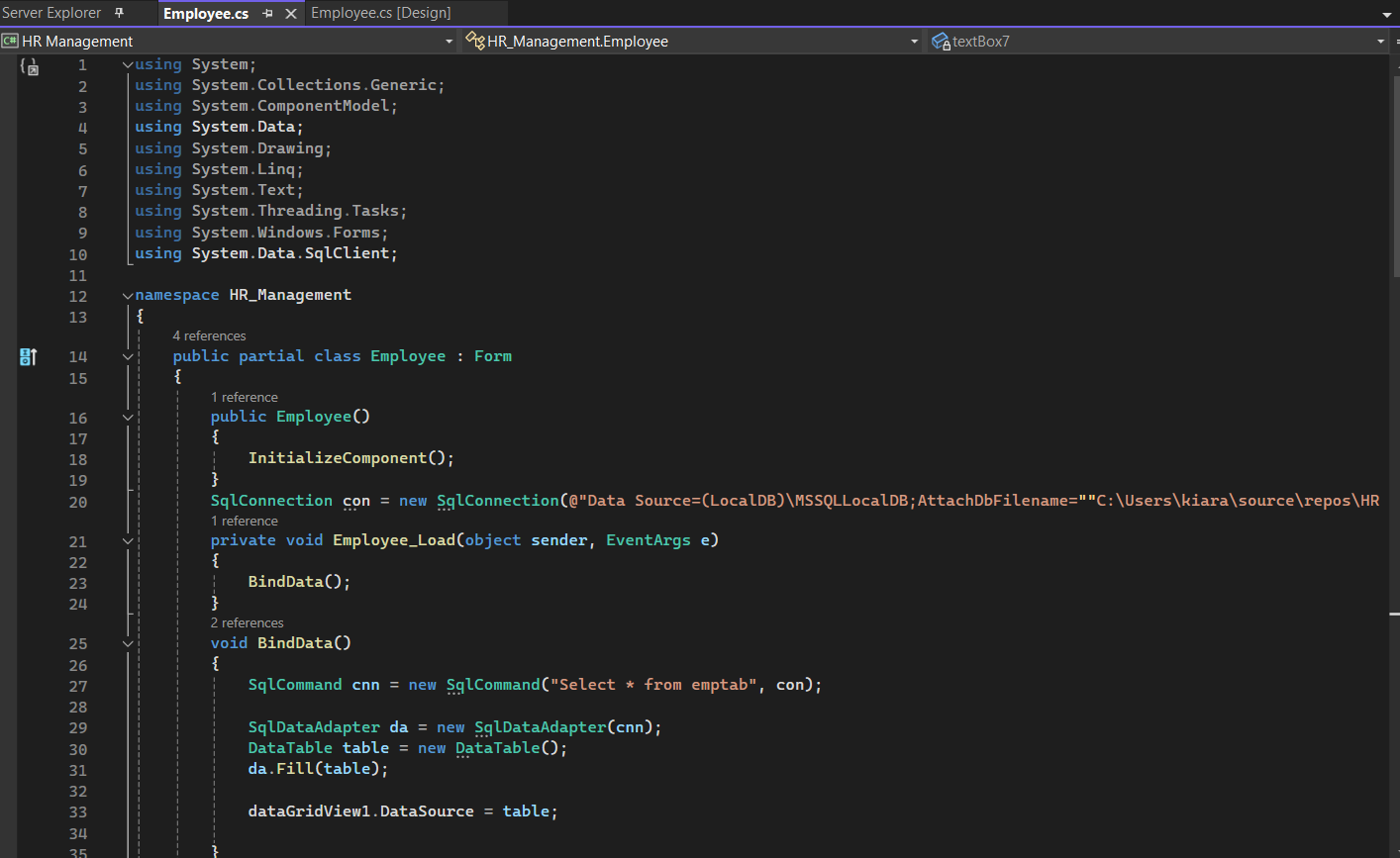
Butonat "Add New" dhe "Delete" i shtova nga kutia e veglave dhe i konfigurova me ngjarjet përkatëse (`Click events`). Për butonin "Add” New", shkrova kodin që merr të dhënat nga fushat e formës dhe i shton ato në tabelën `Employees` në bazën e të dhënave. Për butonin "Delete", shkrova kodin që fshin punonjësin e zgjedhur nga `DataGridView` dhe e heq atë nga baza e të dhënave.

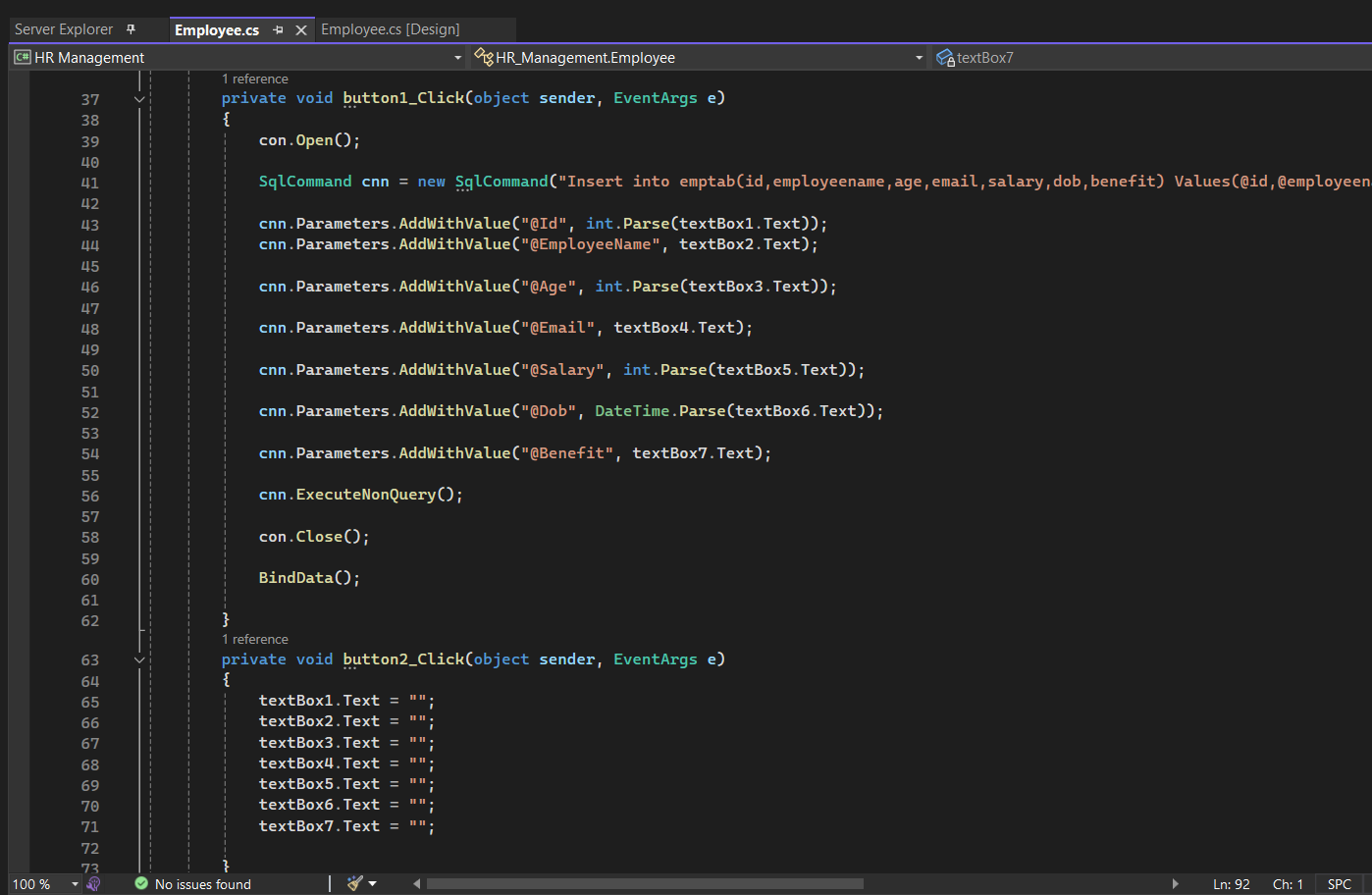
Kjo formë e krijuar është shumë e dobishme për menaxhimin e informacionit të rëndësishëm të punonjësve në kompani, duke ofruar një mënyrë të lehtë dhe efikase për shtimin, përditësimin dhe fshirjen e të dhënave të punonjësve.

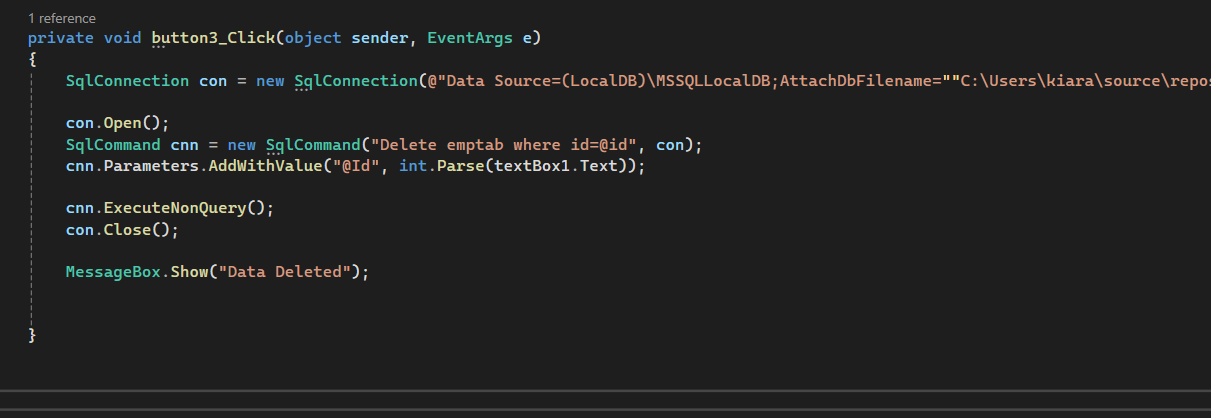
Kam bërë gjithashtu ndryshime në madhësinë e shkrimit dhe ngjyrat për të përmirësuar pamjen dhe përdorshmërinë e formës. Për të ndryshuar madhësinë e shkrimit, përdora pronën `Font` të komponentëve `TextBox`, `Label`, dhe `Button`, duke vendosur madhësi më të madhe për të lehtësuar leximin. Për të ndryshuar ngjyrat, përdora pronën `ForeColor` për të ndryshuar ngjyrën e tekstit dhe pronën `BackColor` për të ndryshuar ngjyrën e sfondit të butonave dhe kontrolleve të tjera. Këto përmirësime vizuale e bëjnë formën më tërheqëse dhe të lehtë për t'u përdorur.



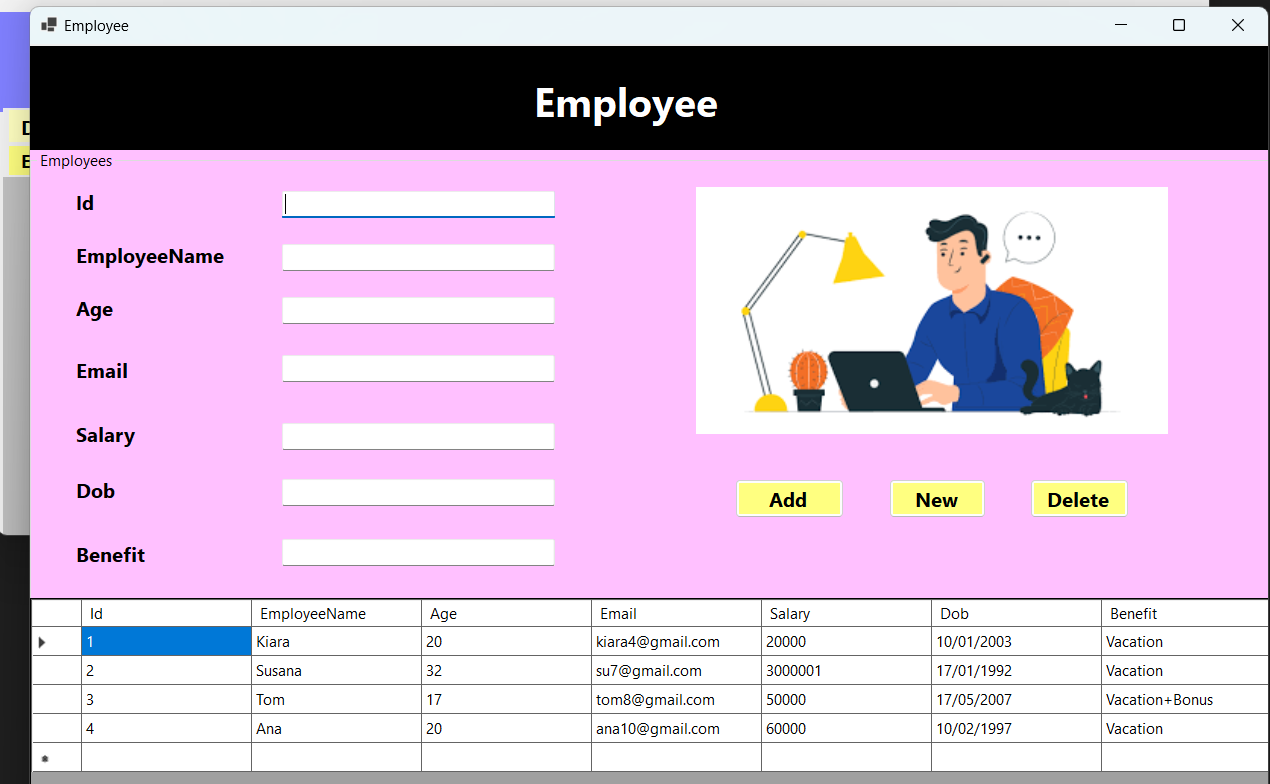




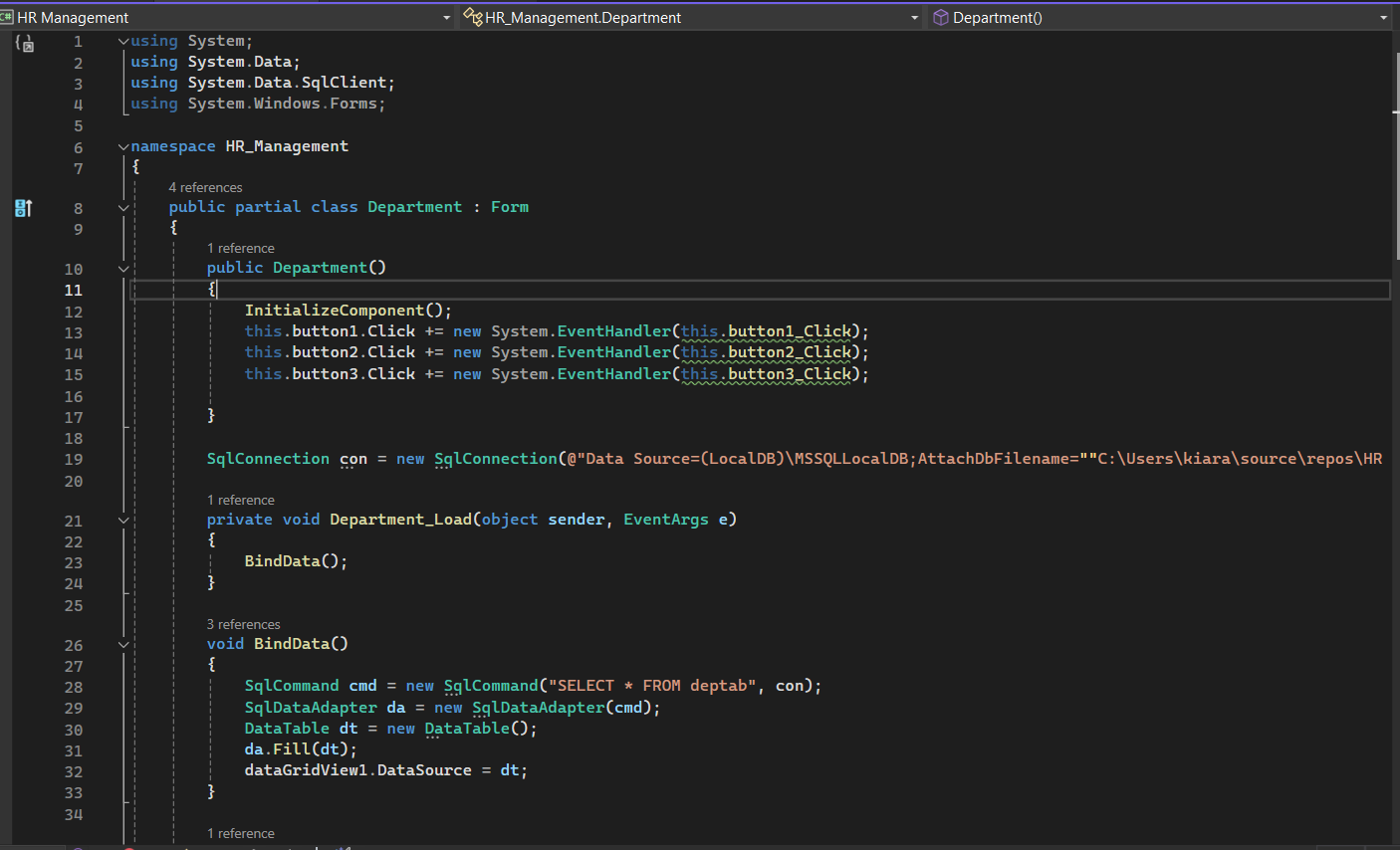


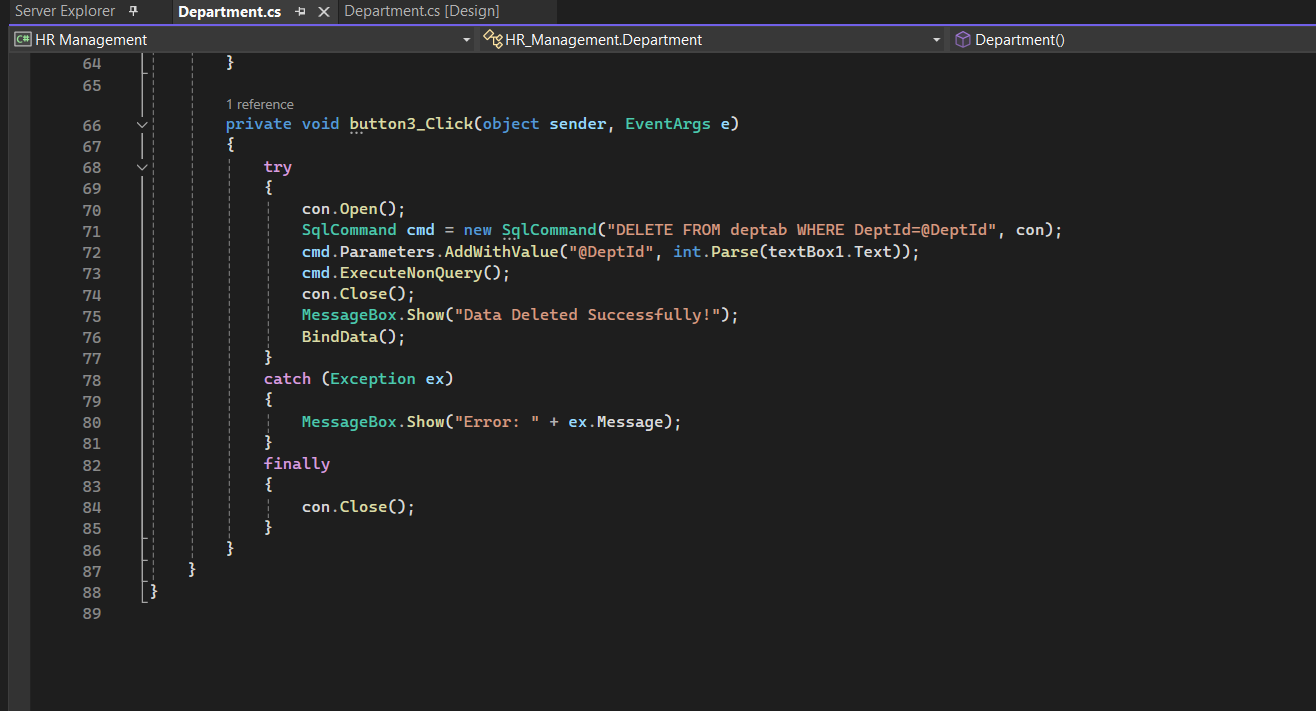
Ky kod i mësipërm është një aplikacion Windows Forms për menaxhimin e informacionit të punonjësve në një kompani, i ndërtuar duke përdorur C# dhe SQL Server. Në kod, lidhja me bazën e të dhënave është realizuar përmes një SqlConnection që përdor skedarin HR.mdf. Kur forma Employee ngarkohet, funksioni BindData ekzekuton një komandë SQL për të marrë të dhënat nga tabela emptab dhe i shfaq ato në një DataGridView. Funksioni BindControls bën lidhjen (binding) e TextBox-ave me DataGridView për të lehtësuar manipulimin e të dhënave. Butoni "Add” (button1) shton një punonjës të ri në tabelë duke marrë të dhënat nga TextBox-at dhe duke i futur ato në tabelën emptab përmes një SqlCommand. Komandat con.Open() dhe con.Close() përdoren për të hapur dhe mbyllur lidhjen me bazën e të dhënave për çdo operacion. Butoni "Clear" (button2) pastron të gjitha fushat e tekstit për të përgatitur formën për të dhëna të reja. Butoni "Delete" (button3) fshin punonjësin e zgjedhur nga baza e të dhënave duke përdorur Id të futur në textBox1. Edhe këtu, komandat con.Open() dhe con.Close() sigurojnë që lidhja me bazën e të dhënave të menaxhohet saktë. Komponentët kryesorë të kodit janë:

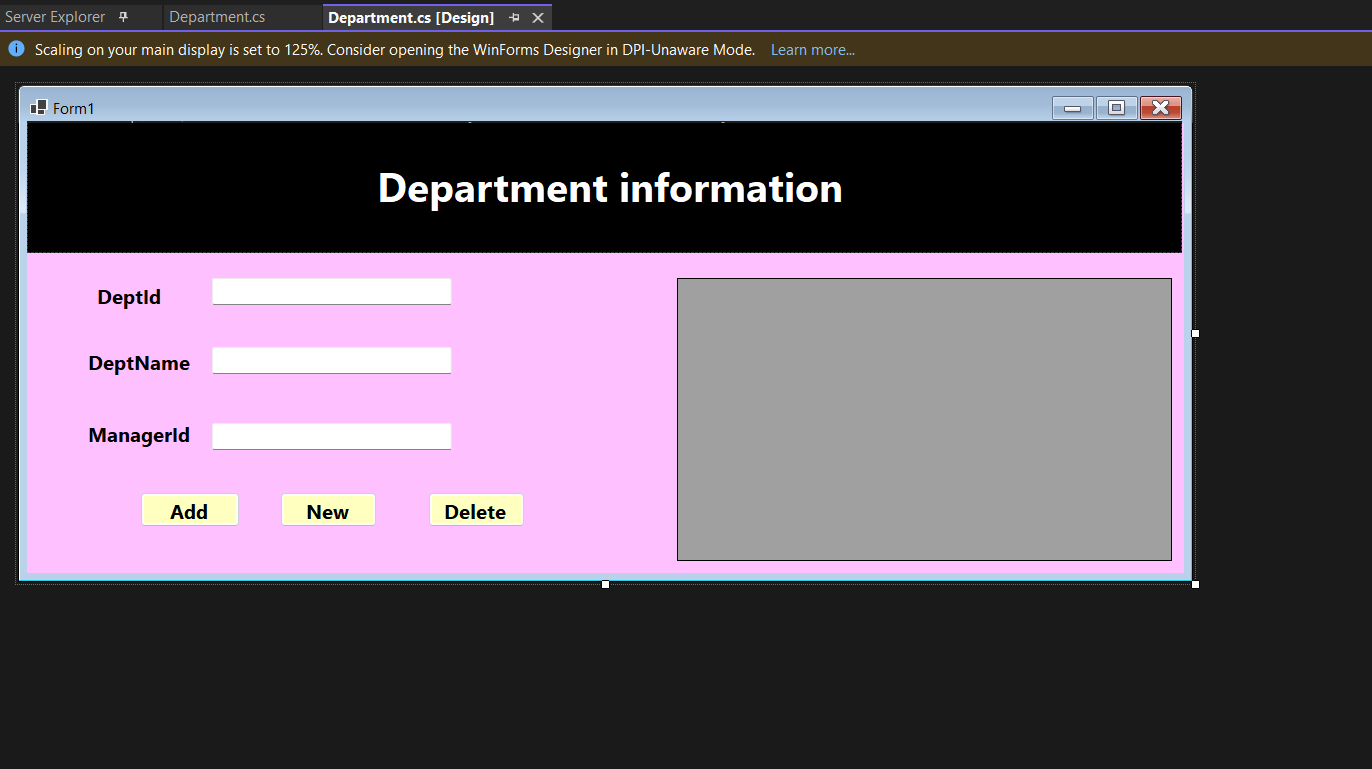
* SqlConnection: Përdoret për të vendosur lidhjen me bazën e të dhënave.
* SqlCommand: Përdoret për të ekzekutuar komandat SQL (p.sh., SELECT, INSERT, DELETE).
* SqlDataAdapter dhe DataTable: Përdoren për të mbushur DataGridView me të dhëna nga baza e të dhënave.
* DataGridView: Kontrolli i formës që shfaq të dhënat e tabelës emptab.
* BindingSource: Përdoret për të lidhur të dhënat me kontrollet e formës.

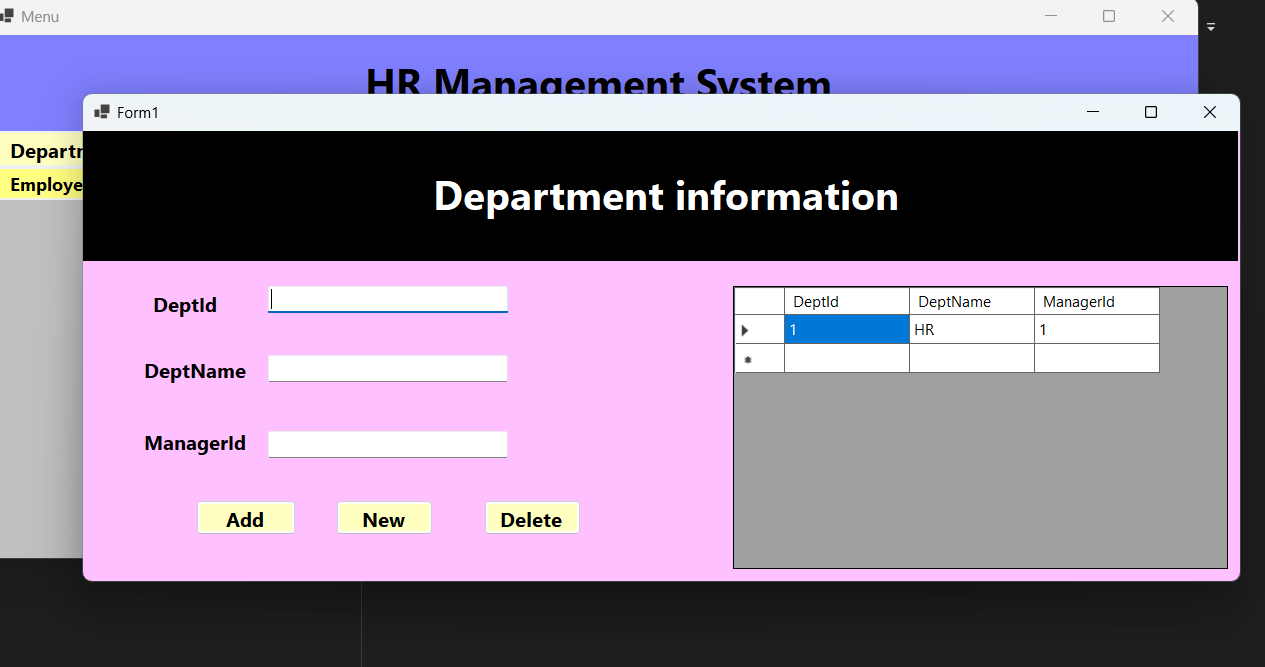
Department form

Ky kod është një aplikacion Windows Forms që menaxhon një tabelë departamenti në një bazë të dhënash. Kur formulari ngarkon, ai merr të dhëna nga tabela e deptab dhe i shfaq ato në një DataGridView. Butoni "Shto" fut një departament të ri në tabelë me ID-në e departamentit, emrin dhe ID-në e menaxherit. Butoni "I ri" fshin kutitë e tekstit dhe butoni "Fshi" fshin një departament me ID-në e specifikuar. Kodi përdor ADO.NET për të bashkëvepruar me bazën e të dhënave dhe përfshin blloqe try-catch për të trajtuar çdo përjashtim që mund të ndodhë gjatë operacioneve të bazës së të dhënave. Metoda BindData përdoret për të rifreskuar DataGridView pas shtimit ose fshirjes së të dhënave. Kur krijohet forma, metoda InitializeComponent() vendos paraqitjen dhe kontrollet e formularit. Pas kësaj, kodi shton mbajtës të ngjarjeve për ngjarjet Kliko të tre butonave (butoni1, butoni2 dhe butoni3). Këta mbajtës të ngjarjeve përcaktohen si metoda në të njëjtën klasë dhe ato do të ekzekutohen kur përdoruesi të klikojë butonat përkatës. Duke përdorur operatorin +=, kodi po i shton këta mbajtës të ngjarjeve në listën ekzistuese të mbajtësve të ngjarjeve për secilin buton. Kjo lejon që formulari t'i përgjigjet të dhënave të përdoruesit dhe të kryejë veprime specifike kur klikohen butonat.



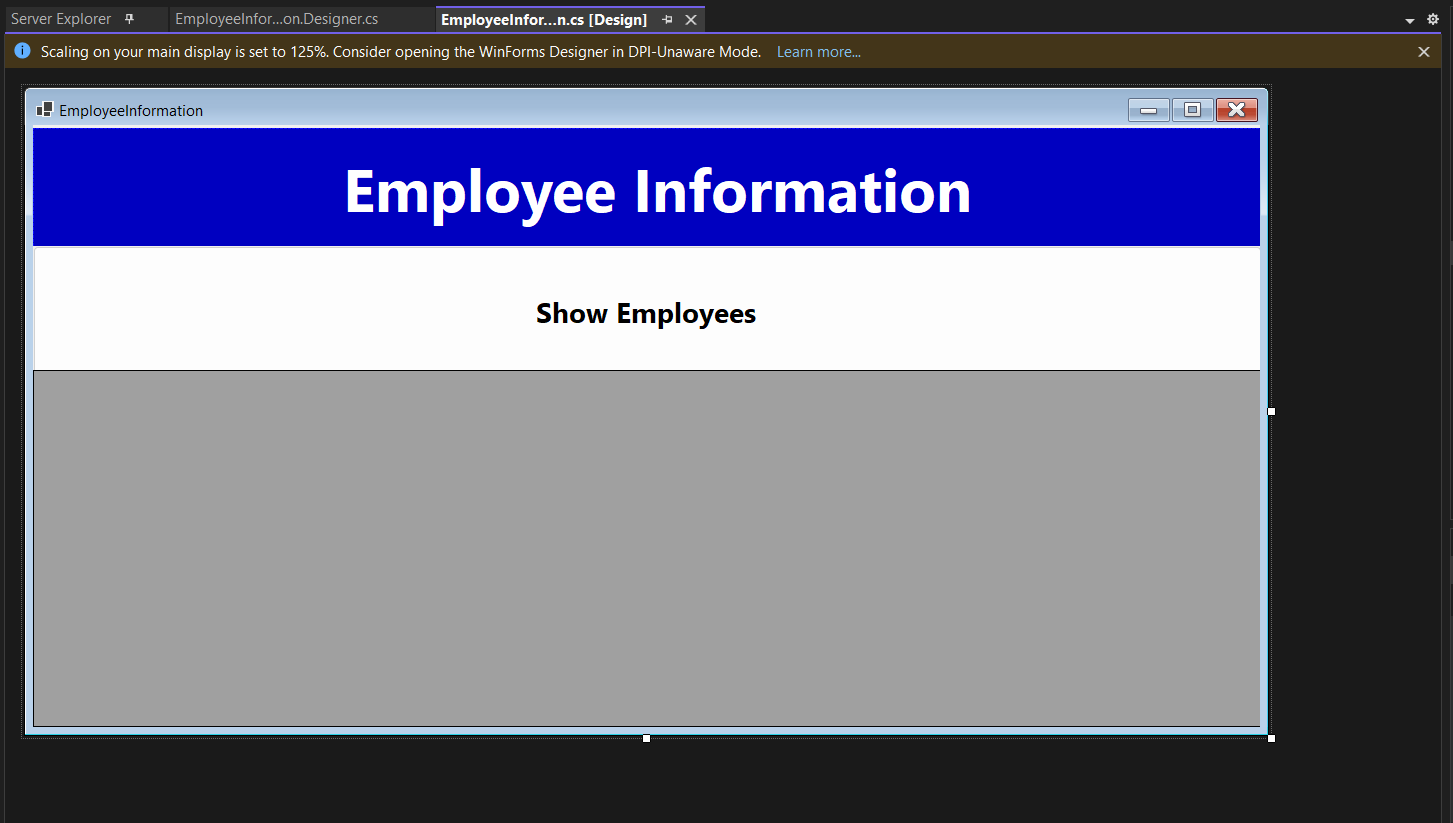


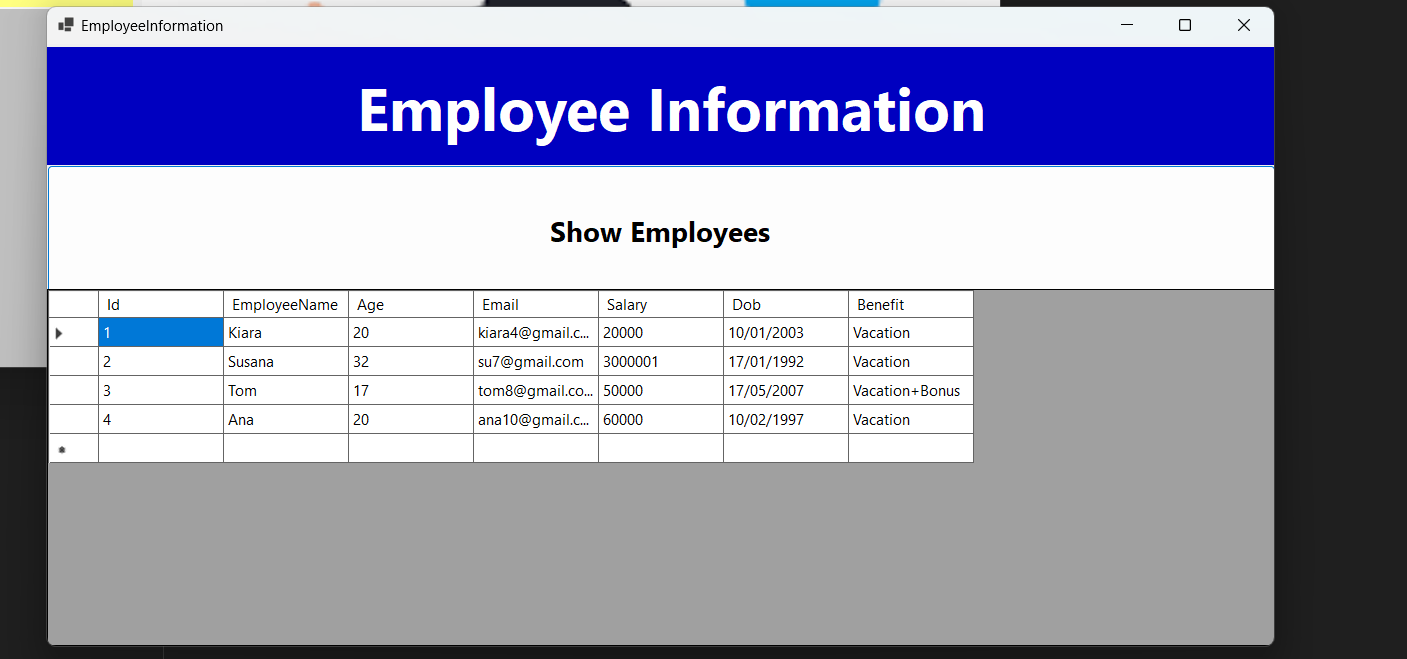


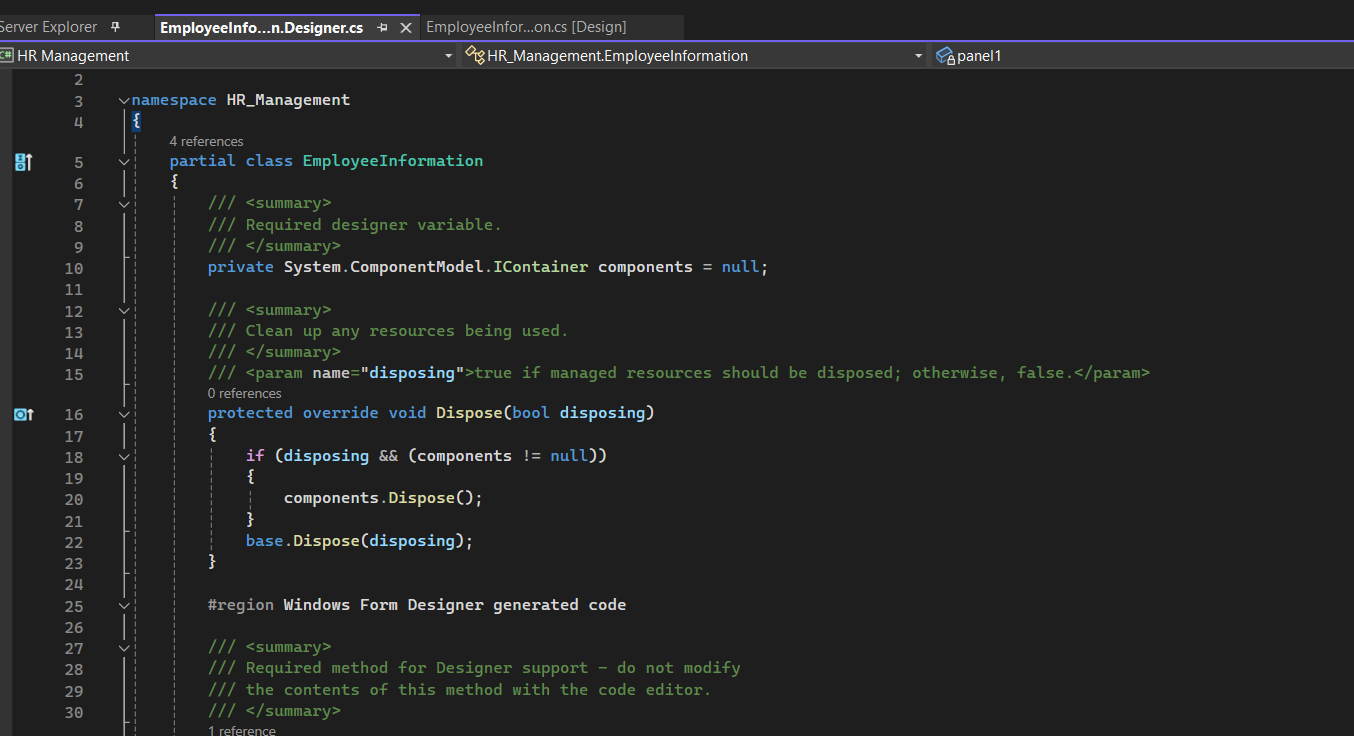


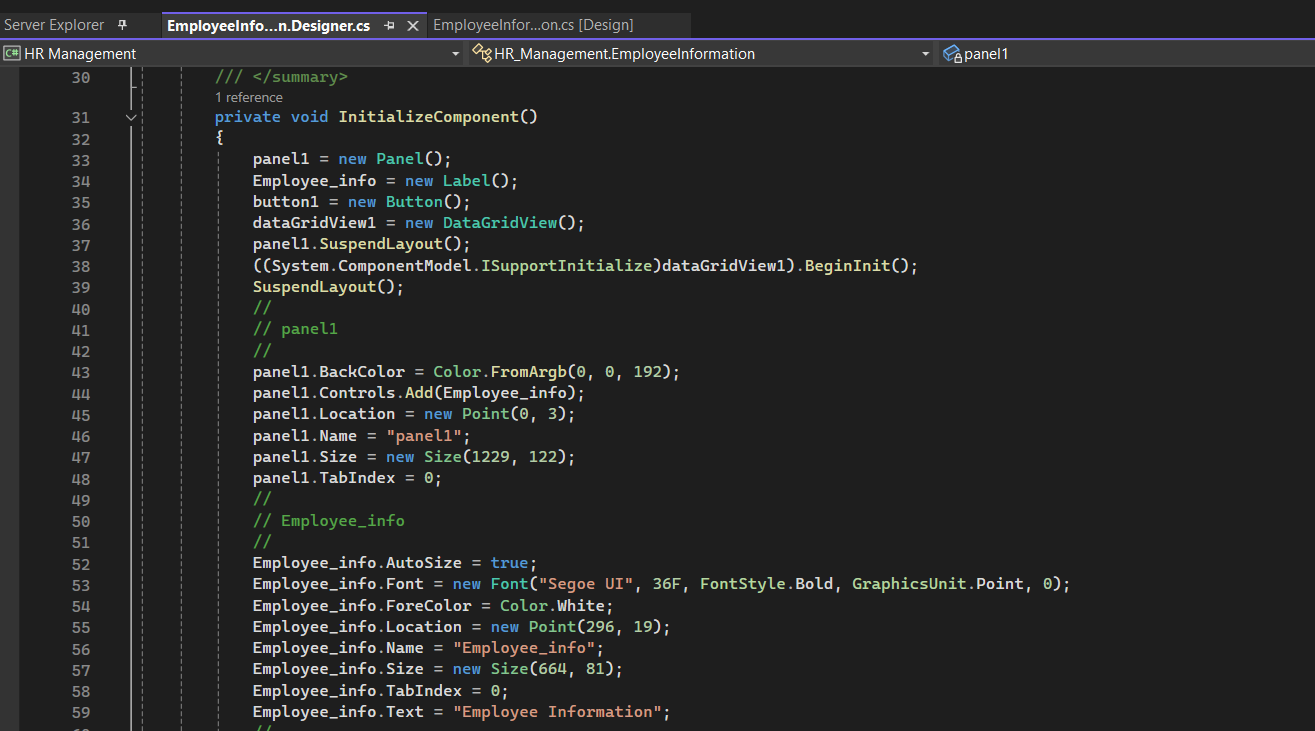
Employee Information form

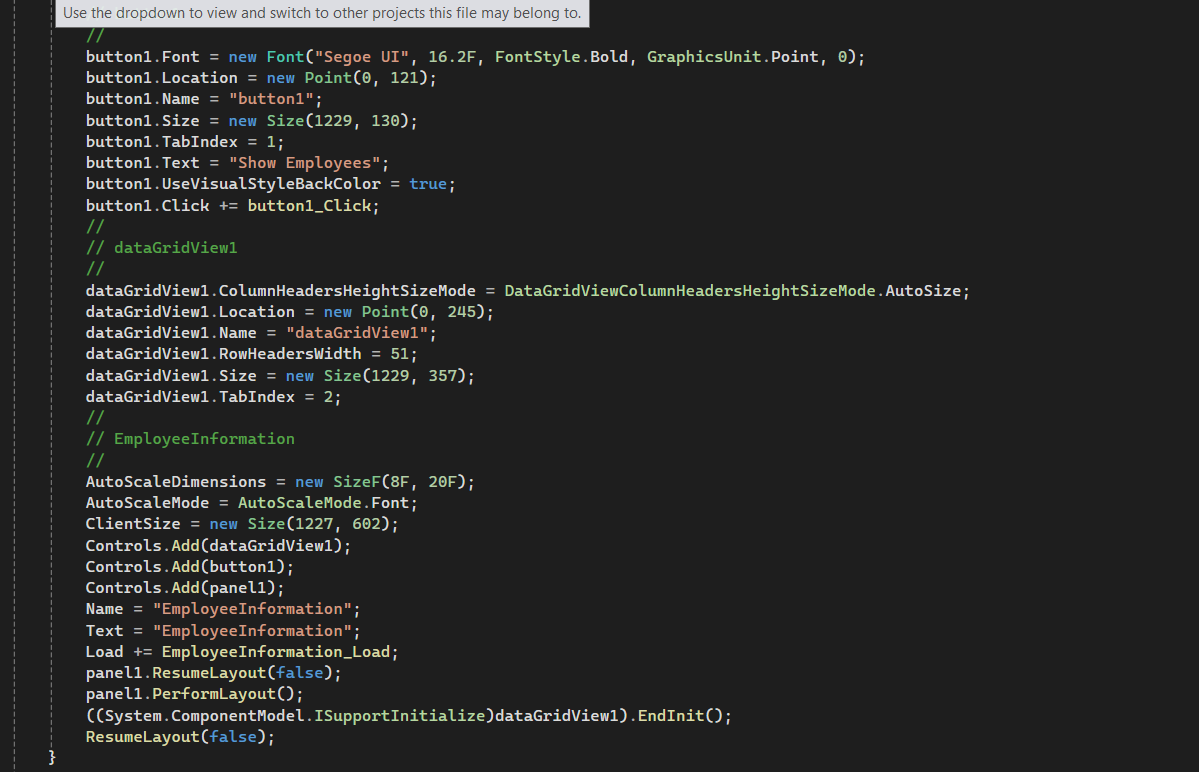
Ky kod është një klasë e pjesshme për një aplikacion Windows Forms që shfaq informacionin e punonjësve. Formulari ka një panel në krye me një etiketë që shfaq titullin "Informacioni i punonjësve". Poshtë panelit, ka një buton të emërtuar "Trego punonjësit" që, kur klikohet, do të shfaqë një listë të punonjësve në një DataGridView. DataGridView ndodhet në fund të formularit dhe do të shfaqë të dhënat e punonjësve. Formulari përfshin gjithashtu një metodë të quajtur InitializeComponent() që konfiguron paraqitjen dhe kontrollet e formularit, dhe një metodë të quajtur EmployeeInformation\_Load() që ekzekutohet kur formulari ngarkohet. Formulari përfshin gjithashtu një metodë Dispose() që pastron çdo burim që përdoret nga formulari kur mbyllet.











Menu form

Ky kod është një klasë C# për një aplikacion Windows Forms që shërben si një menu për një sistem të menaxhimit të burimeve njerëzore. Klasa quhet Menu dhe trashëgon nga klasa Form, e cila është klasa bazë për të gjitha format e Windows. Klasa fillon me disa deklarata të përdorimit që importojnë hapësirat e emrave që përdoren në klasë. Këto përfshijnë System, System.Data, System.Data.SqlClient dhe System.Windows.Forms.

Klasa ka një konstruktor të paracaktuar, Menu(), që thërret metodën InitializeComponent(). Kjo metodë, e cila nuk tregohet në kod, është përgjegjëse për konfigurimin e ndërfaqes së përdoruesit të formularit, duke përfshirë krijimin dhe pozicionimin e shiritit të menysë dhe artikujve të menysë.

Klasa ka tre mbajtës ngjarjesh për artikujt e menysë: Employees\_Click, EmployeeInformation\_Click dhe DepartmentToolStripMenuItem\_Click. Këta mbajtës të ngjarjeve thirren kur klikohet artikulli përkatës i menysë. Çdo mbajtës i ngjarjeve krijon një shembull të ri të një formulari (punonjës, informacion punonjës ose departament) dhe thërret metodën Show() për të shfaqur formularin.

Klasa ka gjithashtu një mbajtës të ngjarjeve për artikullin e menysë "Exit", Exit\_Click, i cili thërret metodën Application.Exit() për të mbyllur aplikacionin.

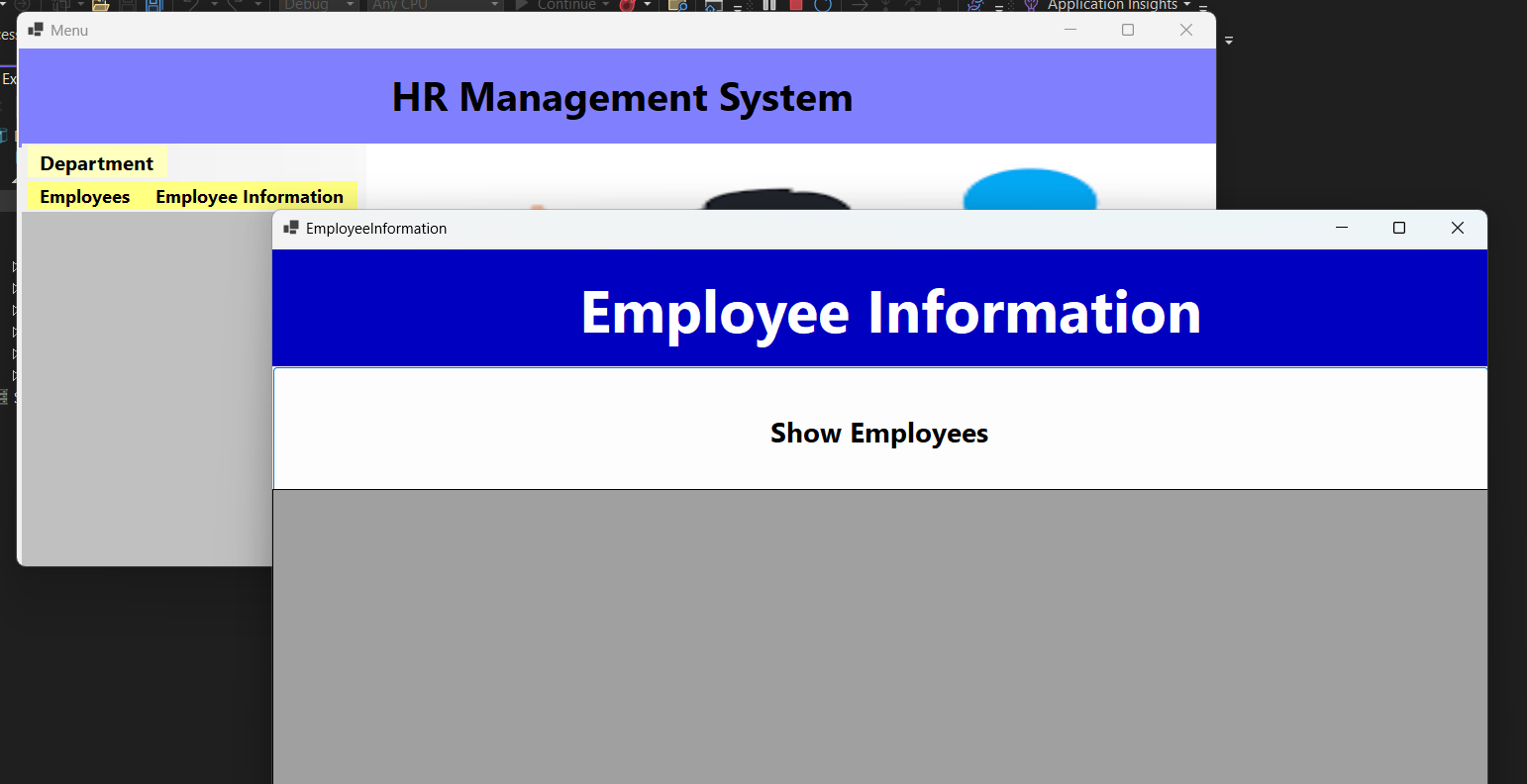
Klasa ka gjithashtu një mbajtës të ngjarjeve Load, Menu\_Load, i cili thirret kur ngarkohet formulari. Ky mbajtës i ngjarjeve është aktualisht bosh, por mund të përdoret për të kryer çdo inicializim të nevojshëm kur formulari është i ngarkuar.

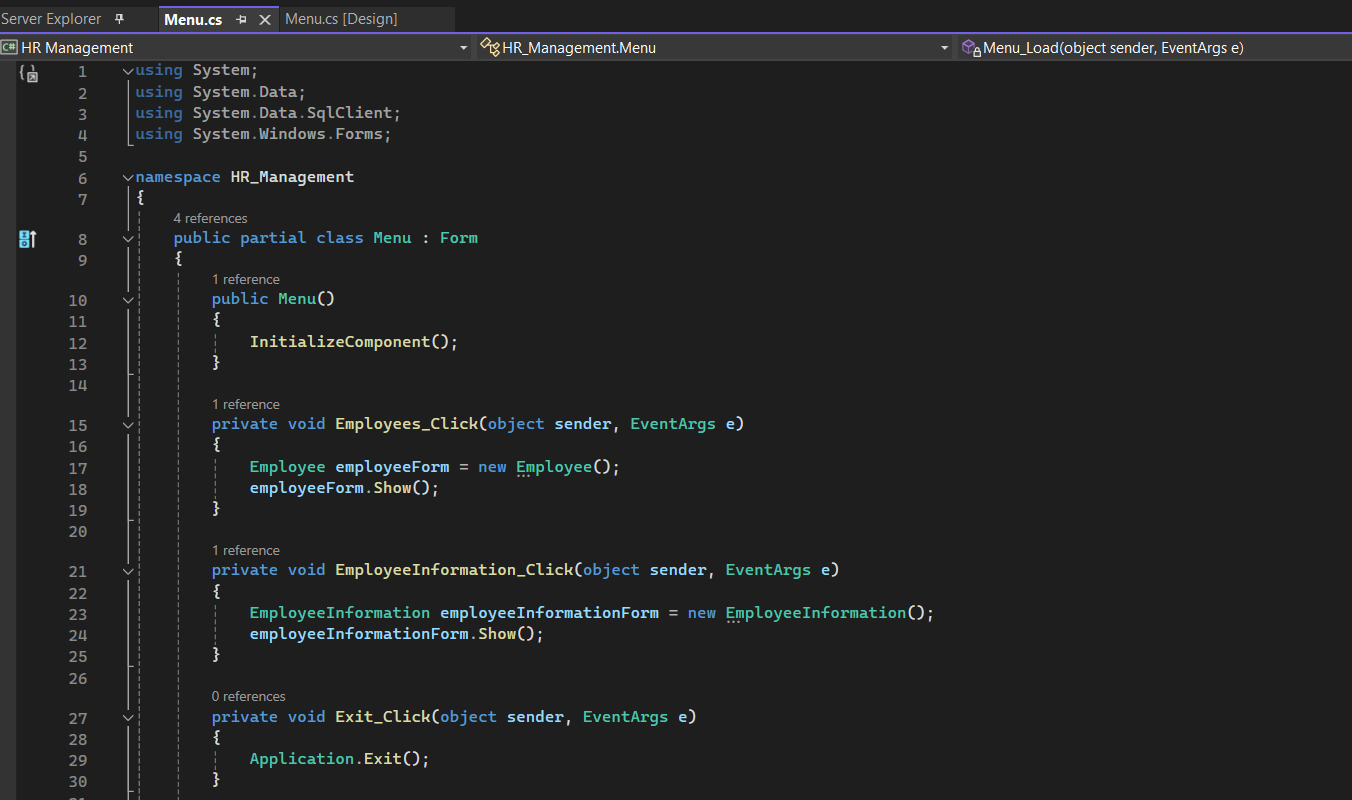
Klasa ka gjithashtu një mbajtës ngjarjesh për ngjarjen menuStrip2\_ItemClicked, menuStrip2\_ItemClicked, i cili nuk përdoret në këtë kod.

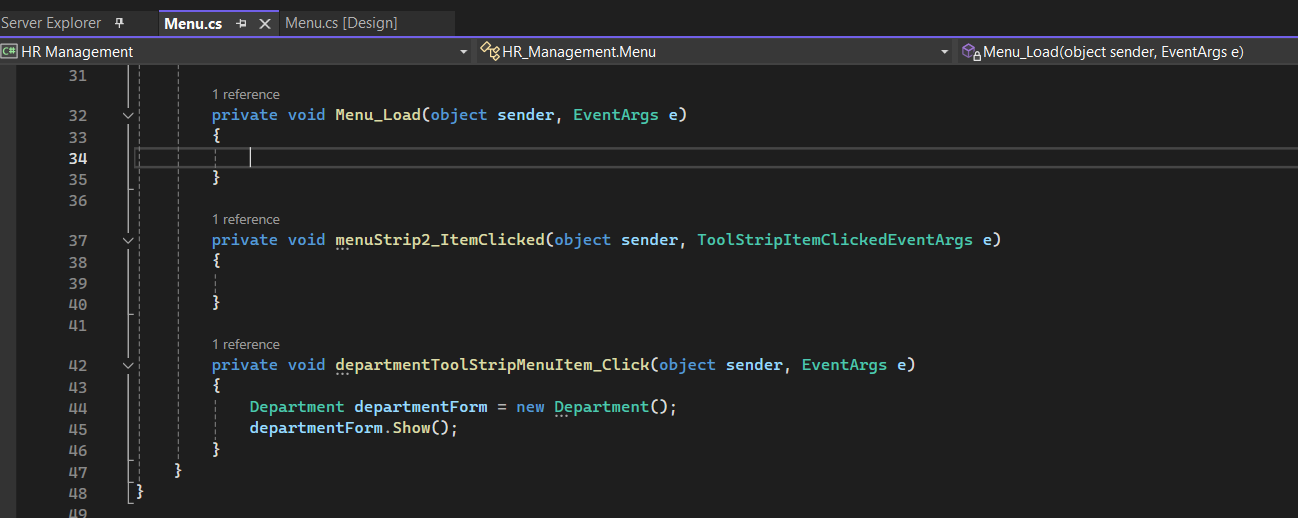
Klasa ka gjithashtu disa fusha private për kontrollet e formularit, të tilla si shiriti i menysë dhe artikujt e menysë, të cilat përdoren në mbajtësit e ngjarjeve për të aksesuar dhe manipuluar kontrollet.

Në përgjithësi, kjo klasë ofron një meny të thjeshtë për një sistem të menaxhimit të burimeve njerëzore, duke i lejuar përdoruesit të lundrojë në forma të ndryshme për të parë dhe menaxhuar informacionin e punonjësve dhe departamentit.









Login form

Ky kod është një klasë C# për një aplikacion Windows Forms që shërben si një formë identifikimi për një sistem të menaxhimit të burimeve njerëzore. Klasa quhet Login dhe trashëgon nga klasa Form, e cila është klasa bazë për të gjitha Format e Windows.

Klasa fillon me disa deklarata të përdorimit që importojnë hapësirat e emrave që përdoren në klasë. Këto përfshijnë System, System.ComponentModel dhe System.Windows.Forms.

Klasa ka një komponentë të fushës private të tipit Icontainer, e cila përdoret nga Windows Forms Designer për të ruajtur informacione rreth komponentëve të formularit.

Klasa ka një metodë Dispose që lëshon burimet e pamenaxhuara të përdorura nga formulari dhe liron opsionalisht burimet e menaxhuara.

Klasa ka një mbajtës të ngjarjeve Load, Login\_Load, i cili thirret kur ngarkohet formulari. Ky mbajtës i ngjarjeve është aktualisht bosh, por mund të përdoret për të kryer çdo inicializim të nevojshëm kur formulari është i ngarkuar.

Klasa ka gjithashtu disa fusha private për kontrollet e formularit, të tilla si etiketat, kutitë e tekstit dhe butonat, të cilat përdoren në ndërfaqen e përdoruesit të formularit.

Klasa ka një metodë InitializeComponent, e cila thirret nga konstruktori i formularit për të inicializuar ndërfaqen e përdoruesit të formularit. Kjo metodë është përgjegjëse për krijimin dhe pozicionimin e kontrolleve të formularit, të tilla si etiketat, kutitë e tekstit dhe butonat.

Klasa ka gjithashtu mbajtës të ngjarjeve për kontrollet e formularit, si button\_login\_Click, button\_clear\_Click dhe button\_exit\_Click, të cilët thirren kur klikohet kontrolli përkatës. Këta mbajtës të ngjarjeve mund të përdoren për të kryer veprime të tilla si regjistrimi, pastrimi i formularit dhe dalja nga aplikacioni.

Në përgjithësi, kjo klasë ofron një formë të thjeshtë hyrjeje për një sistem të menaxhimit të burimeve njerëzore, duke i lejuar përdoruesit të fusë emrin e përdoruesit dhe fjalëkalimin e tij për të hyrë në sistem.

