# Упражнение: Връзка между C# и база данни

## Импортирайте базата данни SoftUni

Импортирайте SoftUni DB в SQL Management Studio (ако все още не е импортиран), като **изпълните** предоставения **.sql** скрипт.

****

## Генерирайте DB-First ORM модел на база данни

Моделирайте **съществуващата база данни**, като използвате **Database First** в **Entity Framework Core**.

**Създайте ново .NET Core приложение**

Първо създайте ново празно **конзолно приложение** и след като бъде създадено, отворете **Package Manager** **конзолата**:



Ще прилича на това:



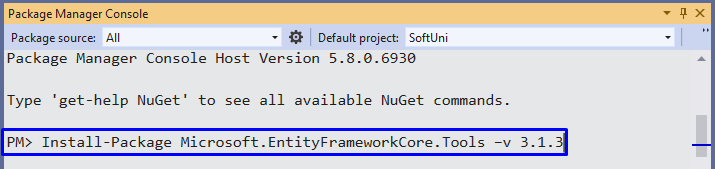
### Инсталирайте SQL Server Tools за Entity Framework

Използвайте го, за да изпълните **една по една** следните команди:

|  |
| --- |
| Install-Package Microsoft.EntityFrameworkCore.Tools -v 3.1.3 |
| Install-Package Microsoft.EntityFrameworkCore.SqlServer -v 3.1.3 |
| Install-Package Microsoft.EntityFrameworkCore.SqlServer.Design |

**Важно**: ако **Package Manager конзолата** ви даде **грешка**, докато се опитвате да **изпълните дадените команди**, опитайте да използвате следните:

|  |
| --- |
| Install-Package Microsoft.EntityFrameworkCore.Tools -Version 3.1.3 |
| Install-Package Microsoft.EntityFrameworkCore.SqlServer -Version 3.1.3 |
| Install-Package Microsoft.EntityFrameworkCore.SqlServer.Design |



Това са **пакетите**, от които ще се нуждаете, за да направите **скафоулд** на нашия **SoftUniContext** от **SoftUni** **базата данни**.

**Генерирайте EF модел на данни от съществуваща БД**

След това трябва да **изпълним** **командата** за **скафоулд** на нашия **контекстен клас**. Ще се състои от 4 неща:

* Първо, името на командата:

|  |
| --- |
| Scaffold-DbContext |

* Второ, връзката, която ще използваме (нашият connection string):

|  |
| --- |
| -Connection "Server=<ServerName>;Database=<DatabaseName>;Integrated Security=True;" |

За **ServerName** използвайте името на вашия локална инстанция на MS SQL Server или ".".

За **DatabaseName** използвайте името на базата данни, която искате да използвате, в случая – **SoftUni**.

* Трето, трябва да декларираме нашия **service provider**, ще използваме **Microsoft.EntityFrameworkCore.SqlServer**:

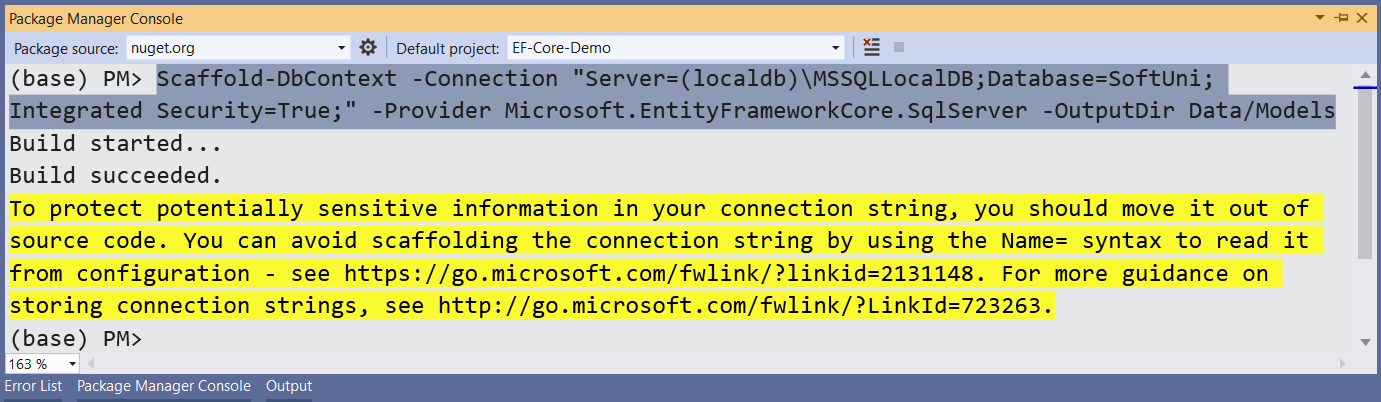
|  |
| --- |
| -Provider Microsoft.EntityFrameworkCore.SqlServer |

* И четвъртото нещо, което ще направим, е да му дадем **директория**, където ще отидат всички наши **модели** (напр. **Models**):

|  |
| --- |
| -OutputDir Data/Models |

Цялата команда ще изглежда така:

|  |
| --- |
| Scaffold-DbContext -Connection "Server=(localdb)\MSSQLLocalDB;Database=SoftUni;Integrated Security=True;" -Provider Microsoft.EntityFrameworkCore.SqlServer -OutputDir Data/Models |



Изпълнете **цялата команда** на **един ред**.

**Променяне на генерирания EF модел на данни**

Entity Framework Core успешно **мапна схемата на базата данни към C# класове**. Но не е достатъчно добър с имената – всички класове са в **множествено число**.

* Използвайте **Solution Explorer** във Visual Studio, за да преместите класа **SoftUniContext** от **Models** в папката **Data** и да преименувате правилно всичките ни класове.
* Използвайте **right click →** [**Rename**] или [**F2**] и натиснете [**OK**] в този **изскачащ прозорец** след всеки клас:



По този начин Visual Studio ще **преименува** **класовете навсякъде**, където са използвани.

Резултатът трябва да изглежда така:



Не забравяйте да коригирате namespace-a на **SoftUniContext**, след като го преместите, и добавете референция към namespace-а на **Models**:

Уверете се, че вашите **namespace-и** са **същите** като тези:

|  |
| --- |
| SoftUni  SoftUni.Data  SoftUni.Models |

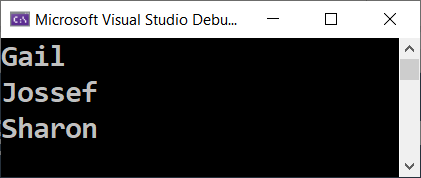
### Премахнете ненужните NuGet пакети

И накрая, можем да **изчистим** пакетите, които вече няма да използваме, от **GUI** на **package manager-а** или като изпълним тези **команди**:

|  |
| --- |
| Uninstall-Package Microsoft.EntityFrameworkCore.Tools -r  Uninstall-Package Microsoft.EntityFrameworkCore.SqlServer.Design -RemoveDependencies |

## Намерете служители с длъжност

Създайте **public static string FindEmployeesWithJobTitle(SoftUniContext context)** в класа StartUp, за да отпечатате **First Name** на всички служители с **Job Title**, равна на **“Design Engineer”**.



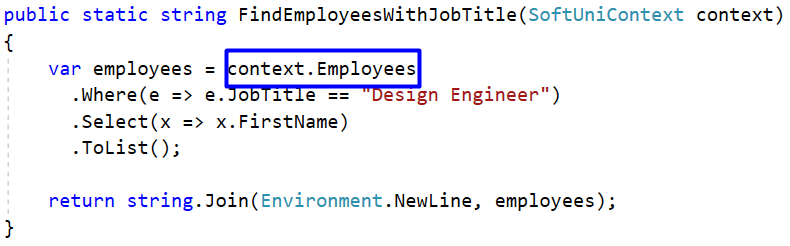
### Подсказка

**Структурата** на вашия **клас** трябва да следва предоставения скелет на проекта, като този:

Graphical user interface, text, application, email

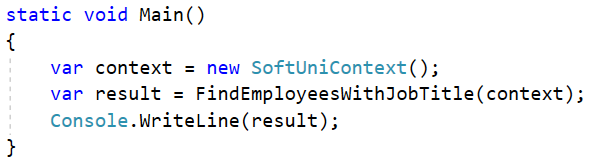
Description automatically generated

Вземете всички служители и ги **филтрирайте** с помощта на **context.Employees**. След това изберете само **First Name** на всеки служител и използвайте **String.Join()**, за да върнете масива от имена като стринг към метода.

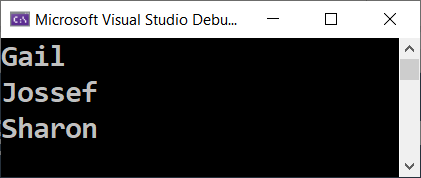


### Изпълнете кода си на конзолата

Извикайте метода **FindEmployeesWithJobTitle(SoftUniContext context)** от входната точка на приложението Main():



Натиснете **[Ctrl+F5]**, за да стартирате приложението. Проверете дали резултатът на конзолата е правилен:



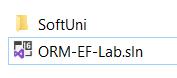
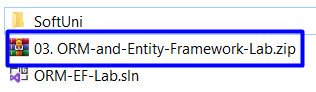
### Качете своя код в Judge

**Запазете вашите файлове** във Visual Studio. Отворете source папката на вашия проект:

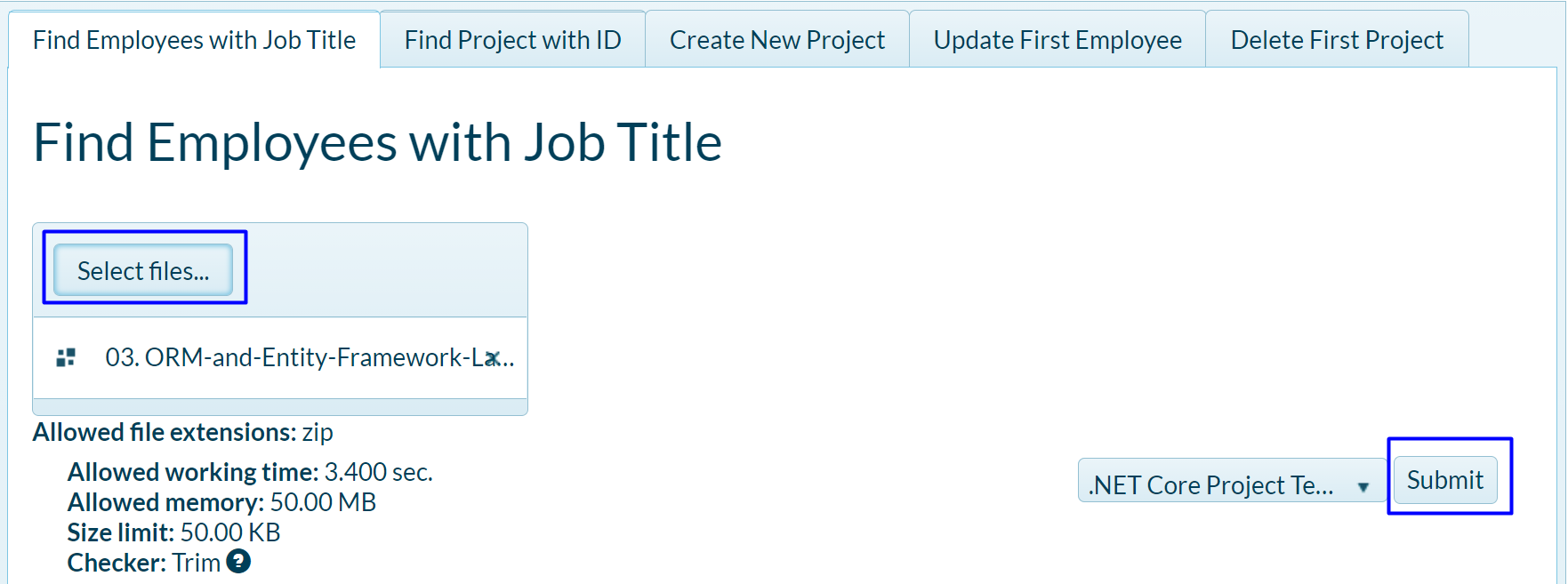
Graphical user interface, application

Description automatically generated

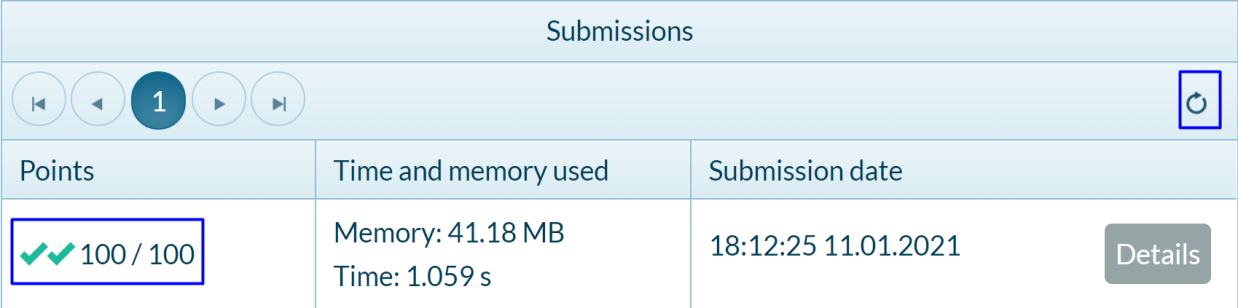
Изтрийте папките **“**bin**”** и **“**obj**”** от папката на **SoftUni** и създайте **ZIP архив** на вашето решение:

Изпратете ZIP архивния файл в **SoftUni Judge**:

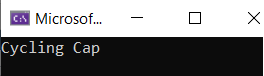


Трябва да получите 100/100 резултат:

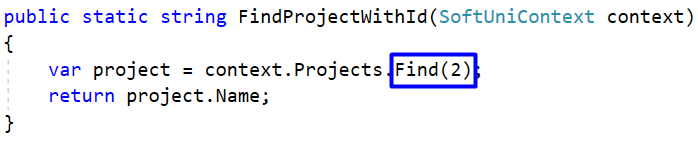


## Намерете проект с ID

Отново използвайте **context** и вземете всички проекти от него. Използвайте метода **.Find()**, за да намерите проекта с **ID 2** и да върнете **името** на проекта.



### Подсказка

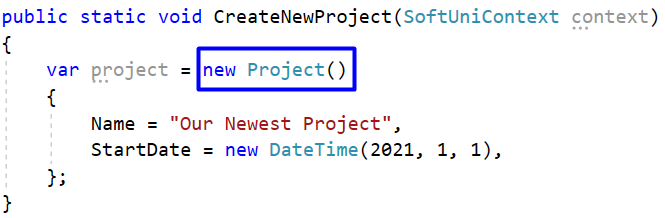


## Създайте нов проект

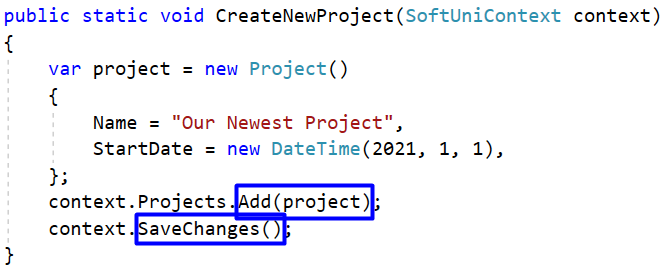
Вашата задача е да създадете **нов проект** в таблицата с **Projects**.

### Подсказка

За да създадете нов **ред** в база данни, използвайте метода **.Add()** на съответния **DbSet**. Първо създайте нов **обект** на **проекта** и дайте стойности на пропъртитата **Name** и **StartDate**.



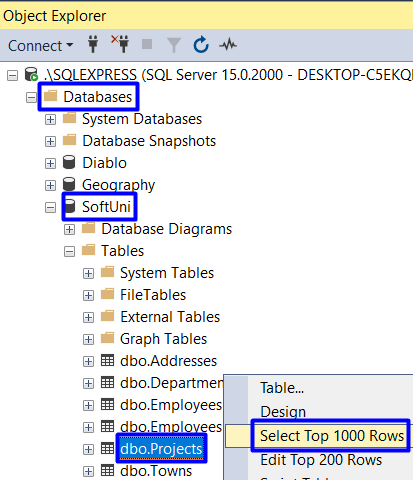
След това добавете обекта към **DbSet** и не забравяйте да **запазите промените** по следния начин:



**Стартирайте** приложението. На конзолата **не се показва** резултат.

### Проверете резултата в БД

За да проверите резултата, отидете на **SQL Server Management Studio -> Object Explorer -> Databases -> SoftUni -> dbo.Projects**. Натиснете с **десния бутон** върху него и изберете **Select Top 1000 Rows**.



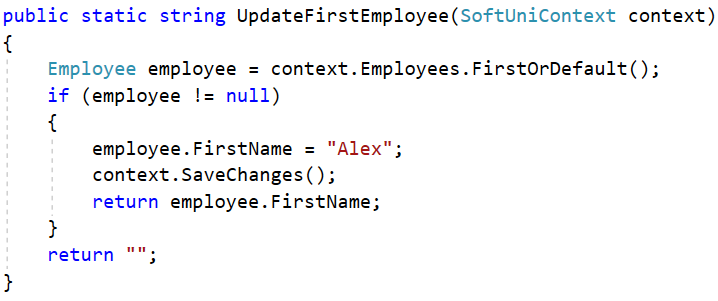
Скролнете надолу до **последния обект**. Трябва да е този, който добавихме с помощта на C# команда във Visual Studio.



## Променете първия служител

Вземете **първия служител** с помощта на метода **.FirstOrDefault()** и променете **първото му име** на **“Alex”**. Не забравяйте да **запазите промените**! В случай, че няма служители, върнете **празен стринг** към метода, в противен случай върнете промененото име на служителя.

### Подсказка



### Проверете резултатите в БД

Това е обектът на базата данни **преди** изпълнението на кода:



След изпълнението на кода обектът **трябва да бъде** **променен**:

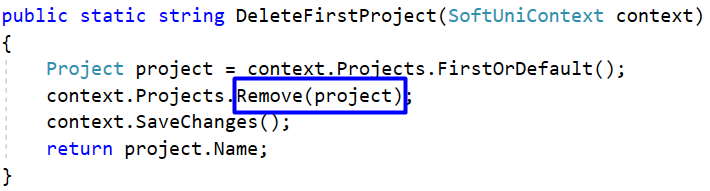


## Изтрийте първия проект

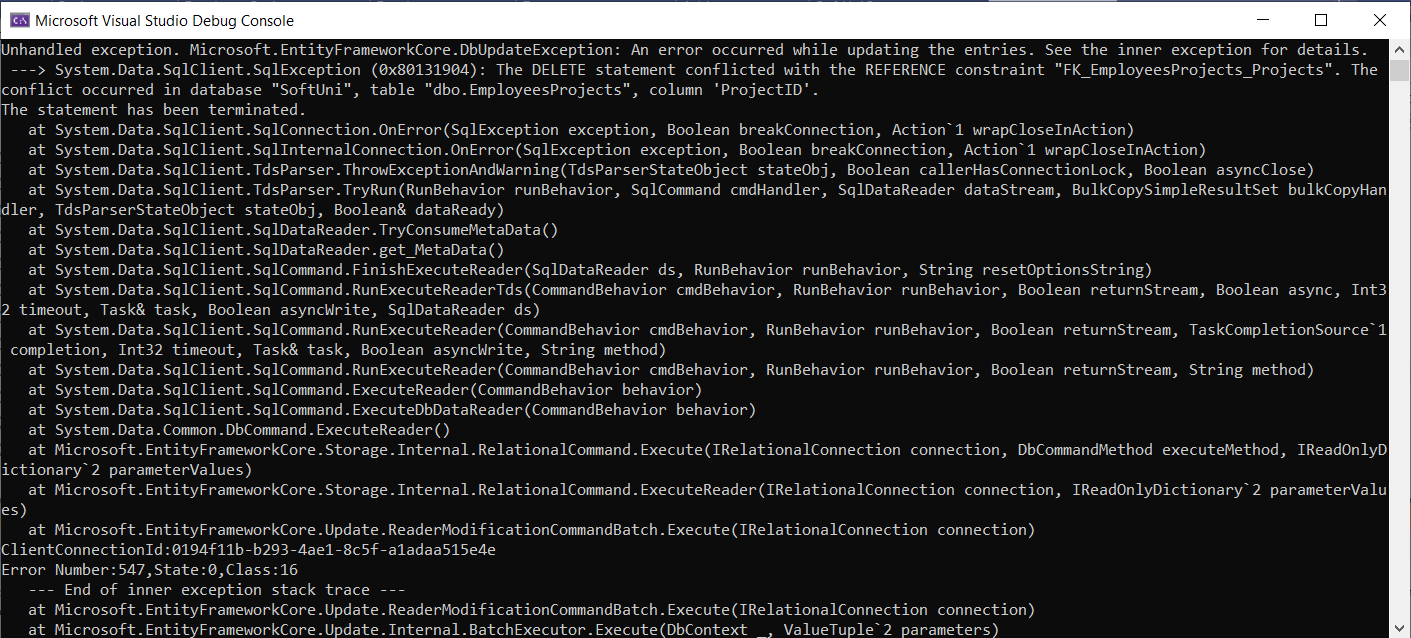
Вземете **първия проект** и го изтрийте с помощта на метода **.Remove()**. Не забравяйте да **запазите промените**! Обектът, който трябва да премахнем, е следният:



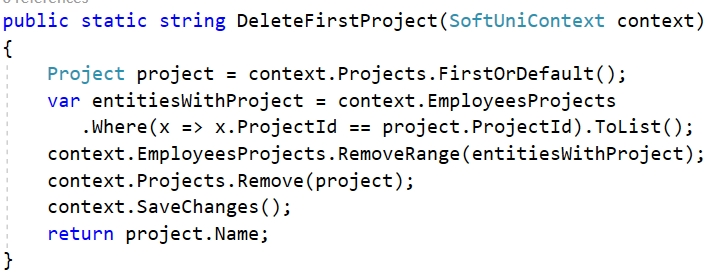
### Подсказка



Въпреки това, когато програмата се изпълни, се появява **съобщение** **за** **грешка**.



Причината за грешката е, че таблицата **EmployeesProjects** в SoftUni БД съдържа колона **ProjectID**. Така че обектите от таблицата Projects **не могат да бъдат изтрити** по този начин, тъй като някои обекти в таблицата **EmployeesProjects** съдържат **Id** на обекта на проекта, който искаме да изтрием. За да разрешим този проблем, можем първо да **изтрием** всички обекти от таблицата **EmployeesProjects**, които съдържат нашия **ProjectId** (в нашия случай с ProjectId=**1**). Командата е следната:

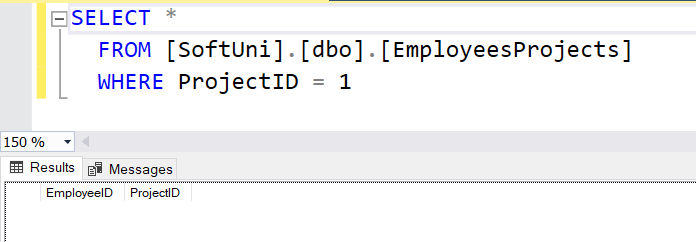


### Проверете резултатите в БД

Изпълнете програмата и вижте резултата в таблицата **Projects** в базата данни SoftUni.

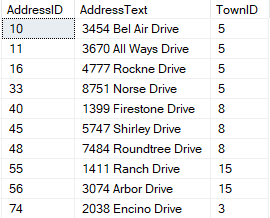
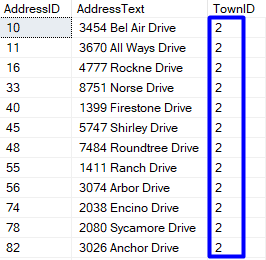


Можете също да проверите таблицата **EmployeesProjects**. Сега тя не съдържа обекти с **ProjectId = 1**.

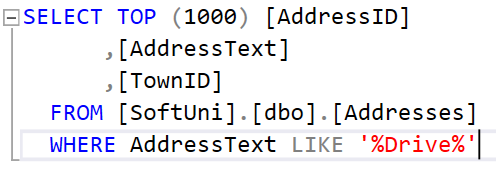


## Променете адресите

Напишете **метод** за променяне на **TownId** на **2** за всички **адреси** с **AddressText**, съдържащ думата **“Drive”**.

Можете да проверите резултата в **SoftUni БД** с тази команда:



Методът **UpdateAddresses(SoftUniContext context)** трябва да върне **броя** на **променените** адреси, преобразувани в **string**.

