

## اليوم الثالث

في الدرس السابقة تكلمنا فقط عن الكونترولر والفيو والمودل وبعض الاشياء الاساسية , اليوم سنتكلم عن كيفية التعامل مع قاعدة البيانات في MVC , وفي بداية التكلم عن هذا الشئ سوف يصادفنا مصطلح شائع وربما راه الكثير منكم وهو ( Entity Framework ) ويختصر بالحرفين ( EF ) .

سابقاً كنا نستخدم تقنية ( ADO.Net ) للتعامل مع قاعدة البيانات , حيث كنا نقوم بإنشاء اتصال مع قاعدة البيانات وننشئ Data Set التي نرجع بها البيانات من القاعدة ثم نحولها الى أوبجكت وهكذا . مايكروسوفت عملت لنا اداة تقوم باجراء هذه العمليات على القاعدة بدل عنا بصورة اوتوماتيكية سميتها ( EF ) .

تعريف آخر لـ EF تعني اننا سوف نتعامل مع الكلاسات بالطبقة العليا , وليس مع الجداول والكائنات الاخرى للقاعدة بأقل كود ممكن . وأنها سوف تجعلنا نستغني عن كتابة الكود الغير ضروري في طبقة الوصول للبيانات ( Data Access layer ) .

ماذا يعني ( First Code ) تعني ان قاعدة البيانات سوف تنشئ من الكود (يقصد كود السي شارب) أو بالعكس الذي نقوم بكتابته , وهناك ثلاث طرق لها :

أولاً – **Database First** : وهذا يعني أننا سوف نقوم بإنشاء قاعدة البيانات وتفصيلها (الجداول , العلاقات , PK... الخ ) أولاً ثم يقوم ( EF ) بإنشاء وبصورة اوتوماتيكية كلاسات تحوي كود سي شارب خاص بهذه القاعدة (هذا الكلاسات ستكون نوعين , الاول نسميه بطبقة الوصول للبيانات – Data Access Layer والثاني هو Business Entity ) وكل كائنات القاعدة .

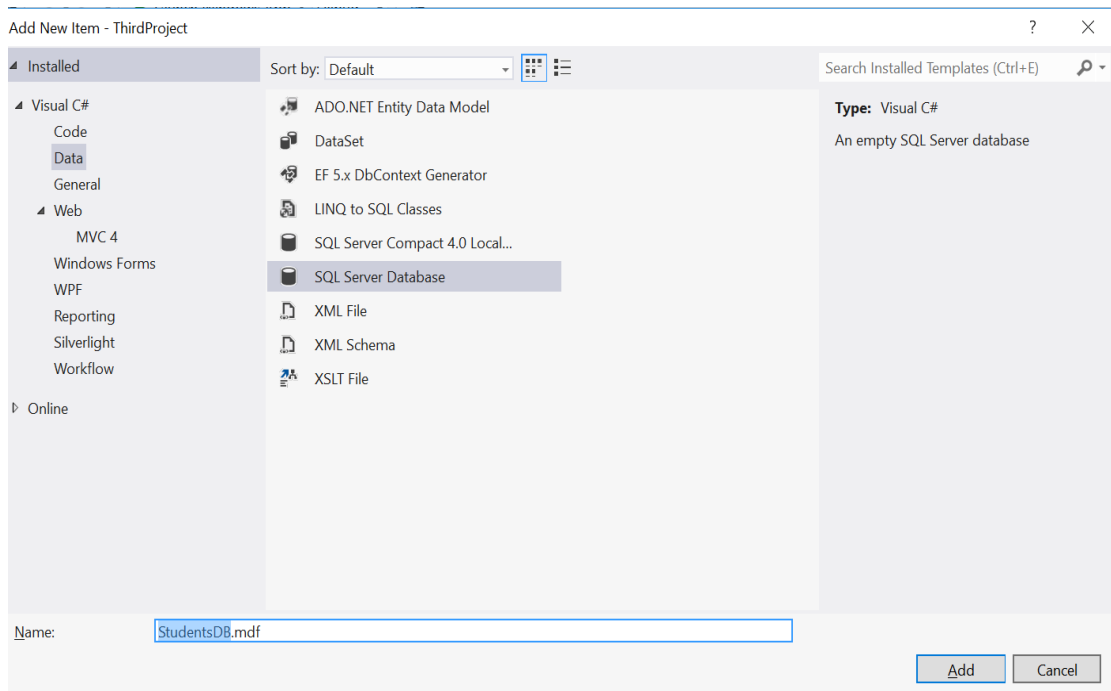
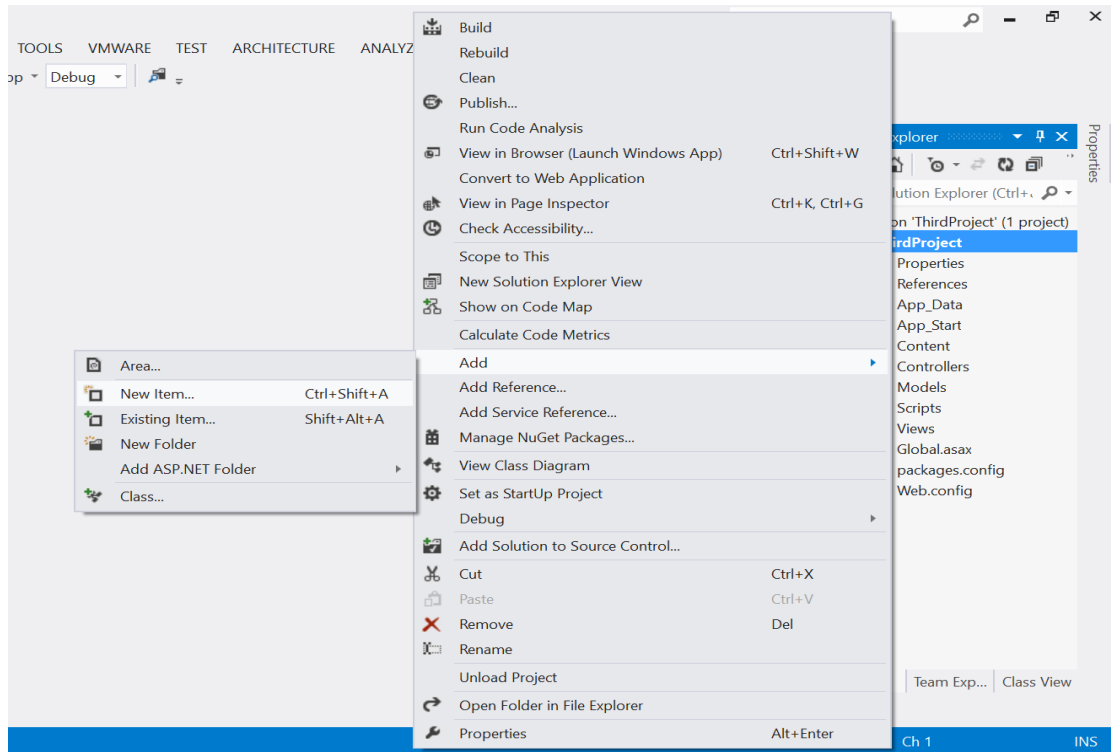
ثانياً – **Model First** : في هذا الطريقة سوف نستعمل مصمم Model (عبارة عن أداة داخل الفيجوال ستوديو تساعدنا على انشاء كلاسات المودل بسهولة ) ثم يأتي دور EF الذي سيقوم بإنشاء قاعدة البيانات اعتماداً على هذه الكلاسات التي قمنا بإنشائها ( كل كلاس سوف يمثل جدول بالقاعدة ) .

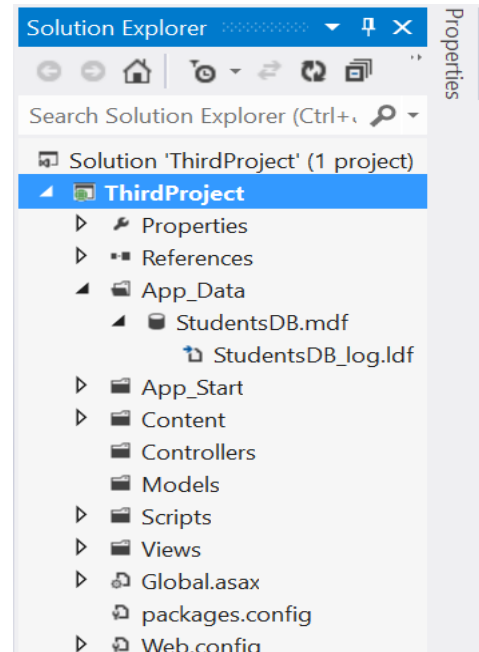
ثالثاً- **Code First** : وهذه الطريقة الاكثر استخداماً حيث أننا سوف ننشئ مجموعة من كلاسات بالكود ( POCO ) وعندما ينفذ التطبيق اول مرة سوف يقوم بإنشاء طبقة ( Data Access layer ) و قاعدة البيانات والجداول .

الـ ( POCO ) تعني كلاسات عادية وهي مختصر لـ ( Plain Old CLR ) , في الامثلة السابقة كنا قد عملنا الكثير منها .

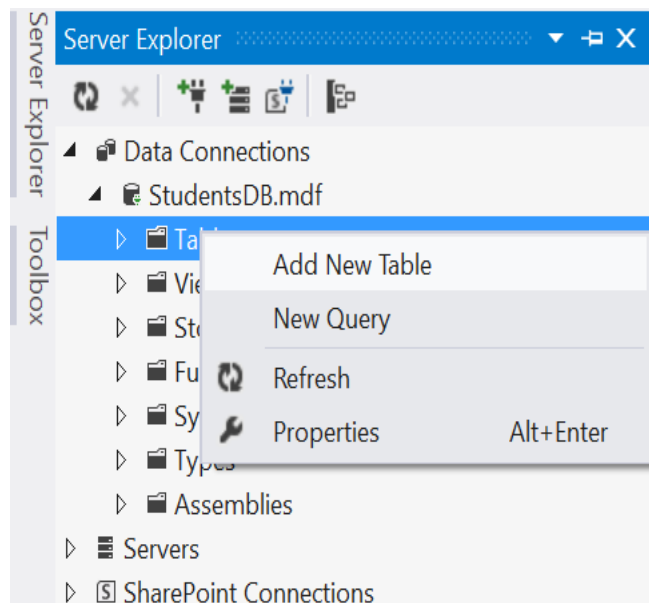
لنعمل مشروع جديد بأسم ( ThirdProject ) :

وننشئ داخله قاعدة البيانات بأسم (StudentsDB) من خلال ( Right Click ) على المشروع :





هذه القاعدة خاصة بالطلاب , وسوف نعمل داخلها جدول بسيط عن درجات الطلاب , سوف نجدها بالمجلد ( App\_Data ) نقرتين على القاعدة حتى تفتح في Server Explorer ثم ( Right Click ) على Tables ونضيف جدول جديد بأسم ( ScoresTb ) :



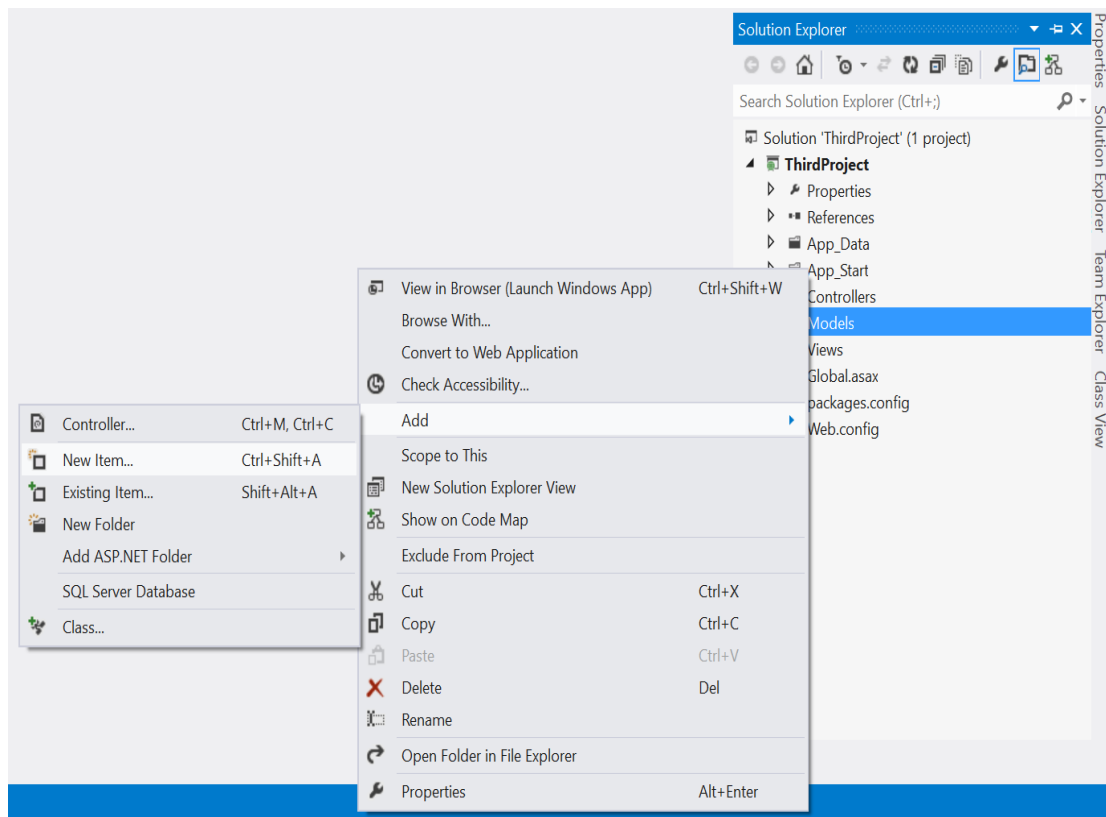
ويكون كود الجدول كالتالي :

```
CREATE TABLE [dbo].[ScoresTb]
(
    [StudentId] INT NOT NULL PRIMARY KEY IDENTITY,
    [StudentName] NVARCHAR(50) NULL,
    [StudentScore] TINYINT NULL
)
```

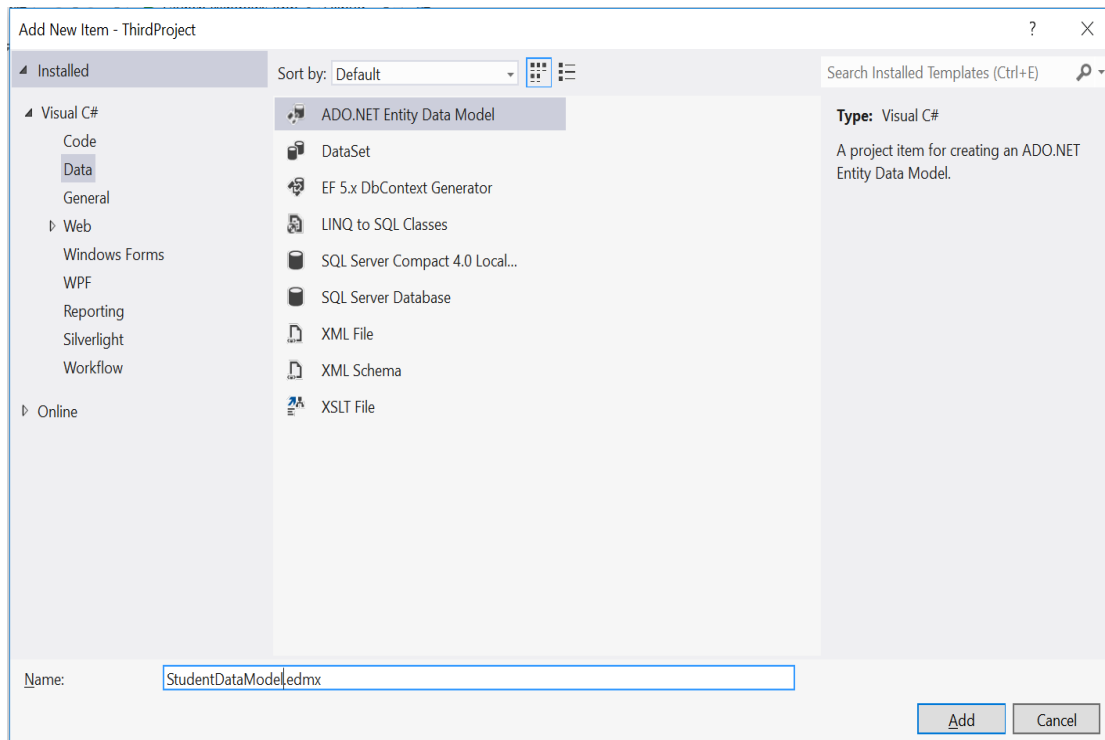
Name	Data Type	Allow Nulls	Default
StudentId	int	<input type="checkbox"/>	
StudentName	nvarchar(50)	<input checked="" type="checkbox"/>	
StudentScore	tinyint	<input checked="" type="checkbox"/>	
		<input type="checkbox"/>	

ثم ننقر على Update من أجل حفظ الجدول .

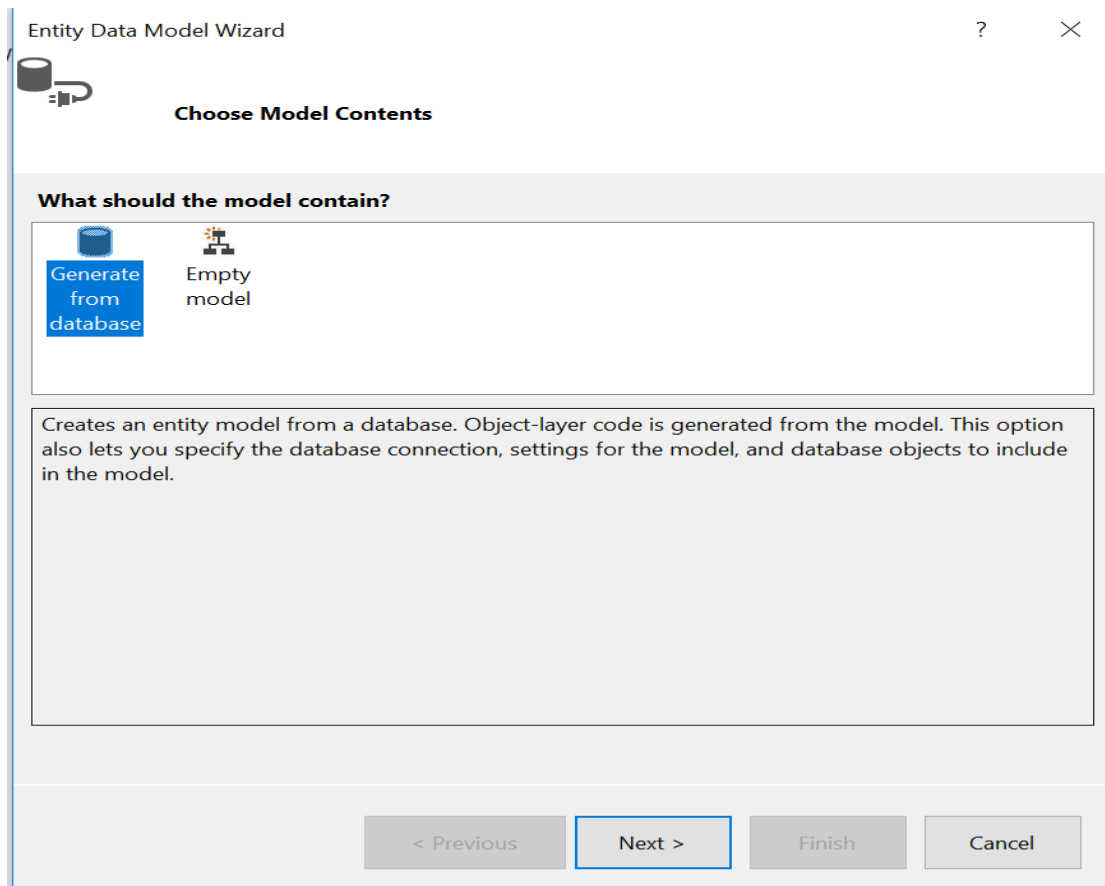
الآن بما أننا أنشأنا قاعدة البيانات الخطوة التالي والمتبعة دائماً هي إنشاء المودل لهذه القاعدة , مايكروسوفت جهزت لنا أداة تسمى ( ORM ) وهي اختصار ( Object Relational Mapping ) فائدتها إنشاء كلاسات من قاعدة البيانات بصورة تلقائية , لنستخدمها ! من خلال ( Right Click ) على المجلد ( Models ) ونختار إضافة عنصر جديد ونختار ( ADO.Net Entity Data Model ) .



نسميها ( StudentDataModel )



الآن ستظهر لنا النافذة التالية تطلب منا ماذا سيكون محتوى هذا النموذج , نختار **Generate from database** ثم ننقر على **Next** :



الآن تظهر لنا نافذة ثانية تطلب منا اختيار الاتصال بالقاعدة ننقر على Next كذلك :

Entity Data Model Wizard

Choose Your Data Connection

**Which data connection should your application use to connect to the database?**

StudentsDB.mdf New Connection...

This connection string appears to contain sensitive data (for example, a password) that is required to connect to the database. Storing sensitive data in the connection string can be a security risk. Do you want to include this sensitive data in the connection string?

No, exclude sensitive data from the connection string. I will set it in my application code.

Yes, include the sensitive data in the connection string.

Entity connection string:

```
metadata=res://*/Models.StudentDataModel.csdl|res://*/Models.StudentDataModel.ssdl|
res://*/Models.StudentDataModel.msl;provider=System.Data.SqlClient;provider connection
string="data source=(LocalDB)\v11.0;attachdbfilename=|DataDirectory|
\StudentsDB.mdf;integrated security=True;MultipleActiveResultSets=True;App=EntityFramework"
```


Save entity connection settings in Web.Config as:

StudentsDBEntities

< Previous **Next >** Finish Cancel

ومن النافذة أدناه سوف نختار جدولنا الذي قمنا بإنشاءه سابقا :

Entity Data Model Wizard ? X

 Choose Your Database Objects and Settings

Which database objects do you want to include in your model?

- Tables
  - dbo
    - ScoresTb
  - Views
  - Stored Procedures and Functions

Pluralize or singularize generated object names

Include foreign key columns in the model

Import selected stored procedures and functions into the entity model

Model Namespace:

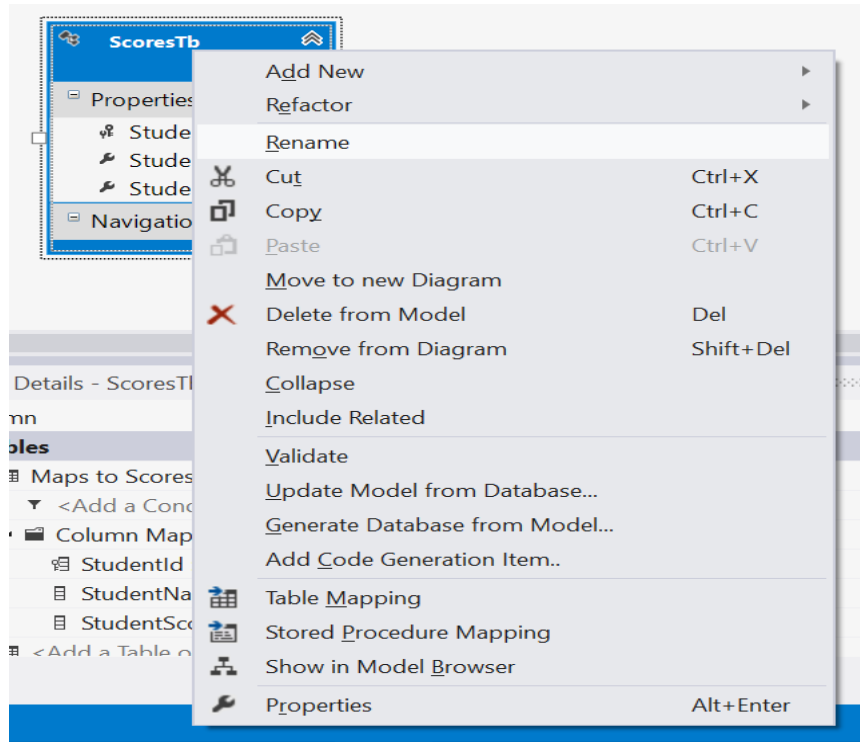
< Previous Next > Finish Cancel

يجب ان تكون Model Namespace هي Models

ثم أنقر على Finish .

الان وكما تلاحظ قد قام بإنشاء ( Entity ) واحد فقط يمثل جدول الدرجات .

لو أردنا تغيير أسم هذا الكلاس نستطيع ذلك من خلال ( Right Click ) عليه ونختار ( Rename ) .



سوف أسميه ( Score ) .

حتى الان أنتهينا من انشاء المودل ! اليس هذا سهل ؟

أعمل Build Solution .

الخطوة التالية هي أنشاء الكونترولر , أضف كونترولر جديد بأسم ( HomeController ) .

Add Controller

Controller name:  
HomeController

Scaffolding options

Template:  
MVC controller with read/write actions and views, using Entity Framework

Model class:  
Score (ThirdProject.Models)

Data context class:  
StudentDBEntities (ThirdProject.Models)

Views:  
Razor (CSHTML)

Advanced Options...

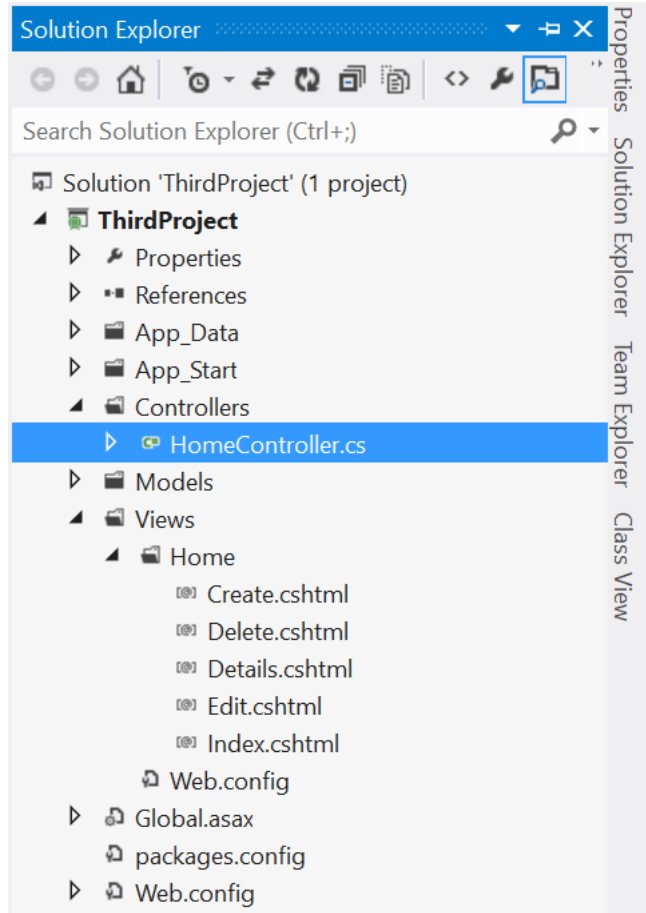
Add Cancel



كما تلاحظ بالصورة اعلاه اننا اخترنا قالب ( Entity Framework ) .والمودل والذاتا كونتكس كلاس .

أنقر على Add .

بصورة أوتوماتيكية الفيچوال ستوديو أضاف لنا خمسة فيو , وبحزد ثمانية أكشنات داخل الكونترولر :



```

HomeController.cs  Index.cshtml  Details.cshtml  Create.cshtml  Edit.cshtml  Delete.cshtml
ThirdProject.Controllers.HomeController  db
using System.Data.Entity;
using System.Linq;
using System.Web;
using System.Web.Mvc;
using ThirdProject.Models;

namespace ThirdProject.Controllers
{
    public class HomeController : Controller
    {
        private StudentDBEntities db = new StudentDBEntities();

        // ...

        public ActionResult Index()...

        // ...

        public ActionResult Details(int id = 0)...

        // ...

        public ActionResult Create()...

        // ...

        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult Create(Score score)...

        // ...

        public ActionResult Edit(int id = 0)...

        // ...

        [HttpPost]
        [ValidateAntiForgeryToken]
        public ActionResult Edit(Score score)...

        // ...

        public ActionResult Delete(int id = 0)...

        // ...

        [HttpPost, ActionName("Delete")]
        [ValidateAntiForgeryToken]
        public ActionResult DeleteConfirmed(int id)...

        protected override void Dispose(bool disposing)...
    }
}

```

- ❖ Index – هذا الاكشن الافتراضي , وعادة يعرض كل محتويات الجدول .
  - ❖ Details – يقوم بعرض تفاصيل عنصر معين .
  - ❖ Create – يقوم بعرض فورم خاصة بإضافة عنصر جديد.
  - ❖ Create(Score) – يضيف عنصر جديد الى قاعدة البيانات.
  - ❖ Edit(Id) – يعرض فورم التعديل على عنصر معين .
  - ❖ Edit(Score) – يحدد بيانات عنصر معين في القاعدة .
  - ❖ Delete(Id) – يعرض فورم حذف عنصر معين .
  - ❖ DeleteConfirm(Id) – تأكيد حذف عنصر من القاعدة .
- الان لنعدل على الكود الموجود في الأكشن ( Create(Score) ) .

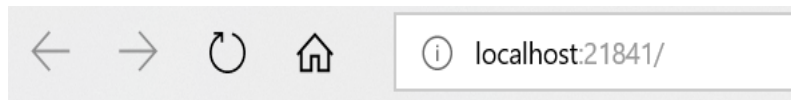
```

[HttpPost]
[ValidateAntiForgeryToken]
public ActionResult Create([Bind(Exclude="StudentId")] Score score)
{
    if (!ModelState.IsValid)
        return View();
    try
    {
        db.Scores.Add(score);
        db.SaveChanges();
        return RedirectToAction("Index");
    }
    catch
    {
        return View();
    }
}

```

الأوبجكت ( db ) قد عرف في بداية الكونترولر وهو من نوع ( StudentDBEntity ) . في الكود أعلاه كما تلاحظ اننا أسستيننا ( StudentId ) من الاضافة لكونه ( PK ) ويأخذ قيمته بصورة تلقائية ويزداد بمقدار واحد , ثم تحققنا من حالة المودل (نقصد التحقق من الفورم الادخال ) بعدها أضفنا البيانات المدخلة الى db حفظنا التغيرات التي حصلت (حفظ في القاعدة) و عملنا اعادة توجيه الى الاكشن الافتراضي ( Index ) .

لنجرب أن ننفذ المشروع :

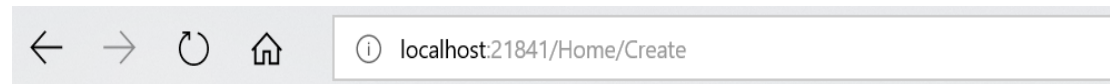


## Index

[Create New](#)

StudentId StudentName StudentScore

أذا تلاحظ اننا لم نكتب في الشريط أي كونترولر وبصورة مباشرة عرض لنا الكونترولر , لان ( IIS ) يعتبر أي كونترولر بأسم ( Home ,Default ) افتراضي ويعرضه بدون الحاجة الى كتابته بالشريط مثل الاكشن ( Index ) , الان أنقر على ( Create ) .  
قد يظهر لك الخطأ كما في الصورة .



## Server Error in '/' Application.

### Compilation Error

**Description:** An error occurred during the compilation of a resource required to service this request. Please review the following specific error details and modify your source code appropriately.

**Compiler Error Message:** CS0103: The name 'Scripts' does not exist in the current context

**Source Error:**

```
Line 49:  
Line 50: @section Scripts {  
Line 51:     @Scripts.Render("~/bundles/jqueryval")  
Line 52: }
```

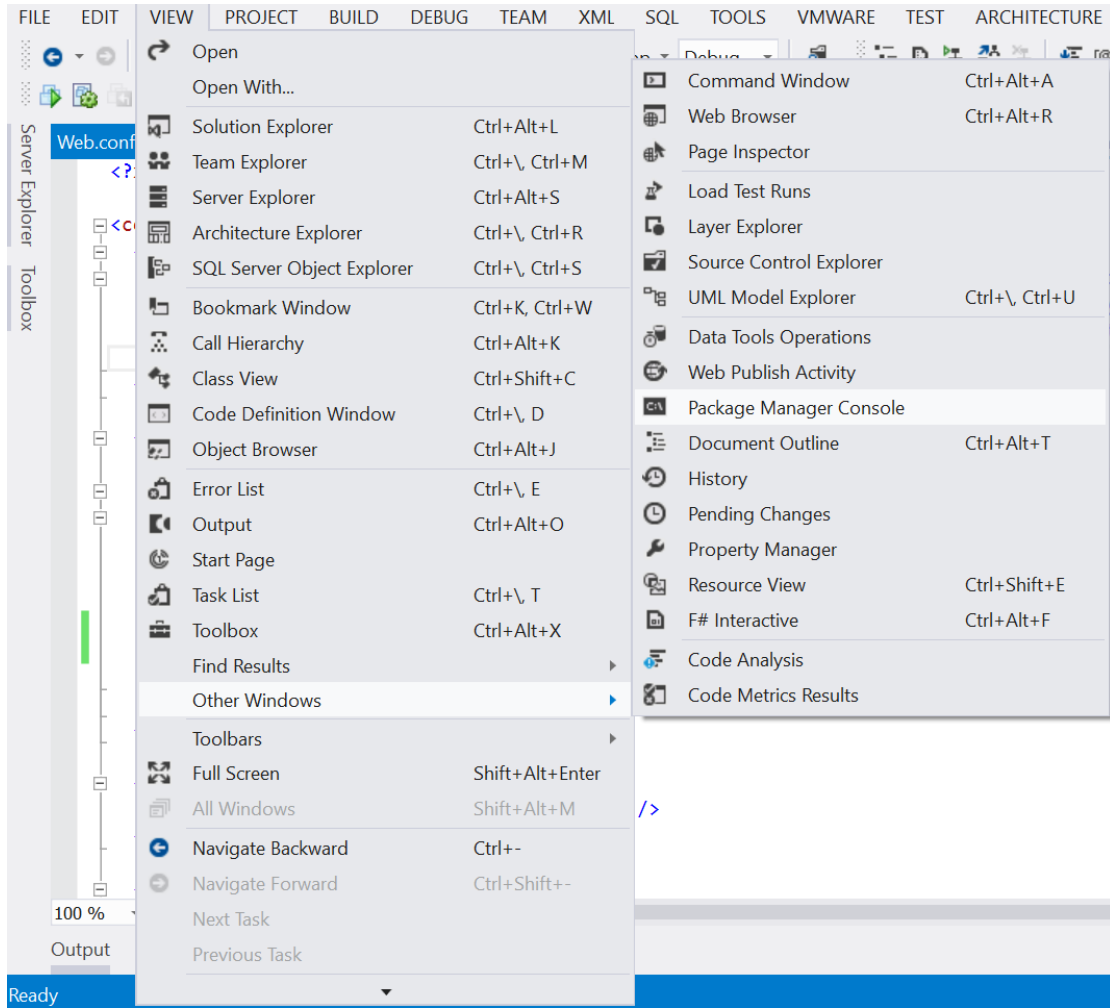
**Source File:** d:\MVC Course\Projects\ThirdProject\ThirdProject\Views\Home\Create.cshtml Line: 51

[Show Detailed Compiler Output:](#)

[Show Complete Compilation Source:](#)

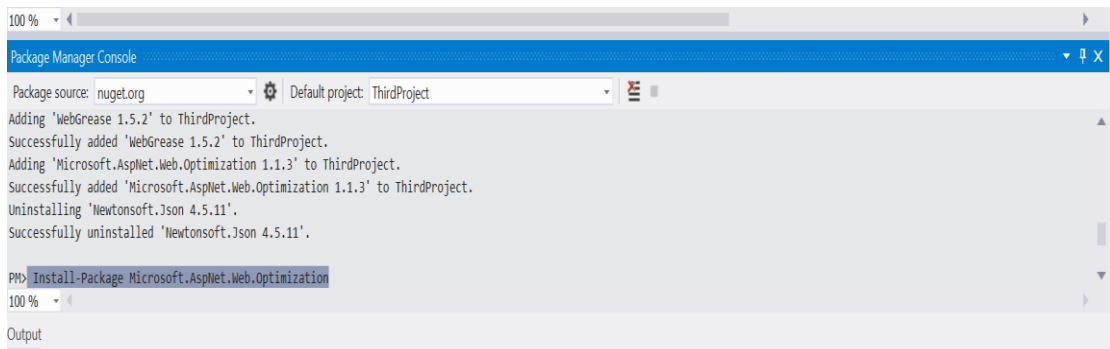
**Version Information:** Microsoft .NET Framework Version:4.0.30319; ASP.NET Version:4.7.2556.0

هذا الخطأ بسكربت الجي كويري المستدعى , المهم الان ان نرجع الى الفيچوال ستوديو ومن ( View ) نختار ( Other Windows ) ثم ( Package Manager Console ) :

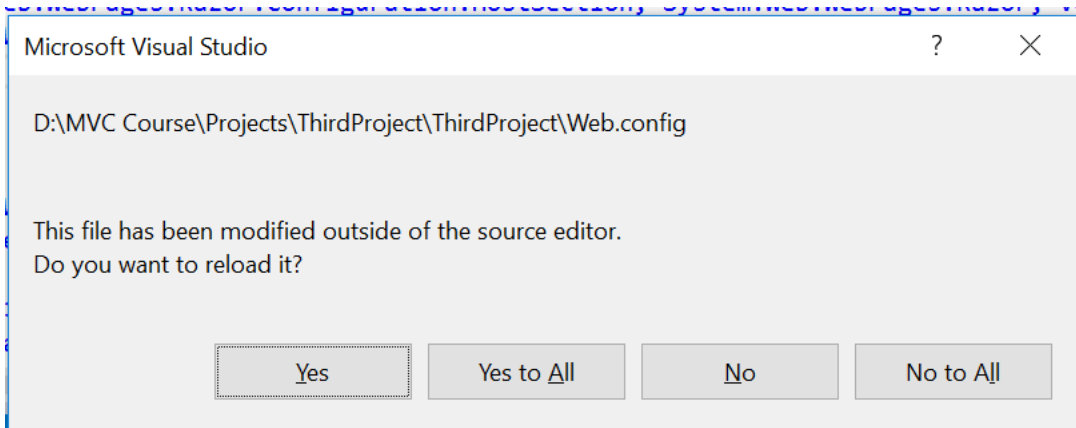


لتنصيب مكتبة ( Optimization ) نكتب الأمر التالي :

```
Install-Package Microsoft.AspNet.Web.Optimization
```

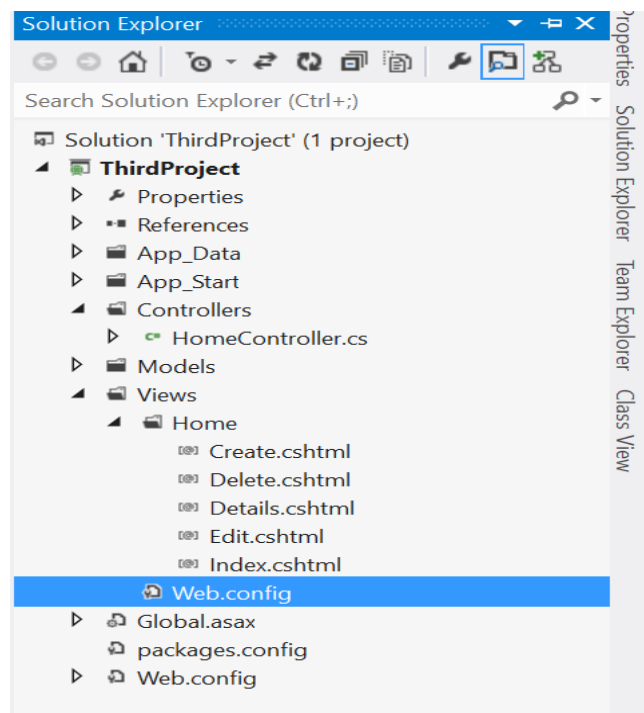


بعدها ننقر Enter , ثم تظهر لنا النافذة أدناه :



نختار ( yes to All) .

نذهب الى الملف ( Web.Config ) الموجود في مجلد ( View ) ونضيف فيه استدعاء مكتبة ( Optimization)

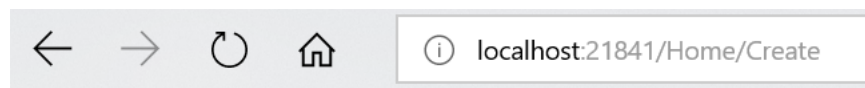


ونستدعي المكتبة من خلال السطر التالي :

```
<add namespace="System.Web.Optimization"/>
```

```
Web.config → X HomeController.cs Index.cshtml Details.cshtml Create.cshtml Edit.cshtml Delete.cshtml Web.config X
<?xml version="1.0"?>
<configuration>
  <configSections>
    <sectionGroup name="system.web.webPages.razor" type="System.Web.WebPages.Razor.Configuration.RazorWebSectionGroup, System.Web.WebPages.Razor, Version=2.0.0.0, Culture=neutral, PublicKeyToken=31BF3856AD364E35" />
    <section name="host" type="System.Web.WebPages.Razor.Configuration.HostSection, System.Web.WebPages.Razor, Version=2.0.0.0, Culture=neutral, PublicKeyToken=31BF3856AD364E35" />
    <section name="pages" type="System.Web.WebPages.Razor.Configuration.RazorPagesSection, System.Web.WebPages.Razor, Version=2.0.0.0, Culture=neutral, PublicKeyToken=31BF3856AD364E35" />
  </configSections>
  <system.web.webPages.razor>
    <host factoryType="System.Web.Mvc.MvcWebRazorHostFactory, System.Web.Mvc, Version=4.0.0.0, Culture=neutral, PublicKeyToken=31BF3856AD364E35" />
    <pages pageBaseType="System.Web.Mvc.WebViewPage">
      <namespaces>
        <add namespace="System.Web.Mvc" />
        <add namespace="System.Web.Mvc.Ajax" />
        <add namespace="System.Web.Mvc.Html" />
        <add namespace="System.Web.Routing" />
        <add namespace="System.Web.Optimization"/>
      </namespaces>
    </pages>
  </system.web.webPages.razor>
  <appSettings>
    <add key="webpages:Enabled" value="false" />
  </appSettings>
</configuration>
</system.web>
```

نفذ المشروع من جديد وأنقر على ( Create New ) :



## Create

Score

StudentId

StudentName

StudentScore

[Back to List](#)



## Index

[Create New](#)

StudentId	StudentName	StudentScore	
1	Mustafa	58	<a href="#">Edit</a>   <a href="#">Details</a>   <a href="#">Delete</a>