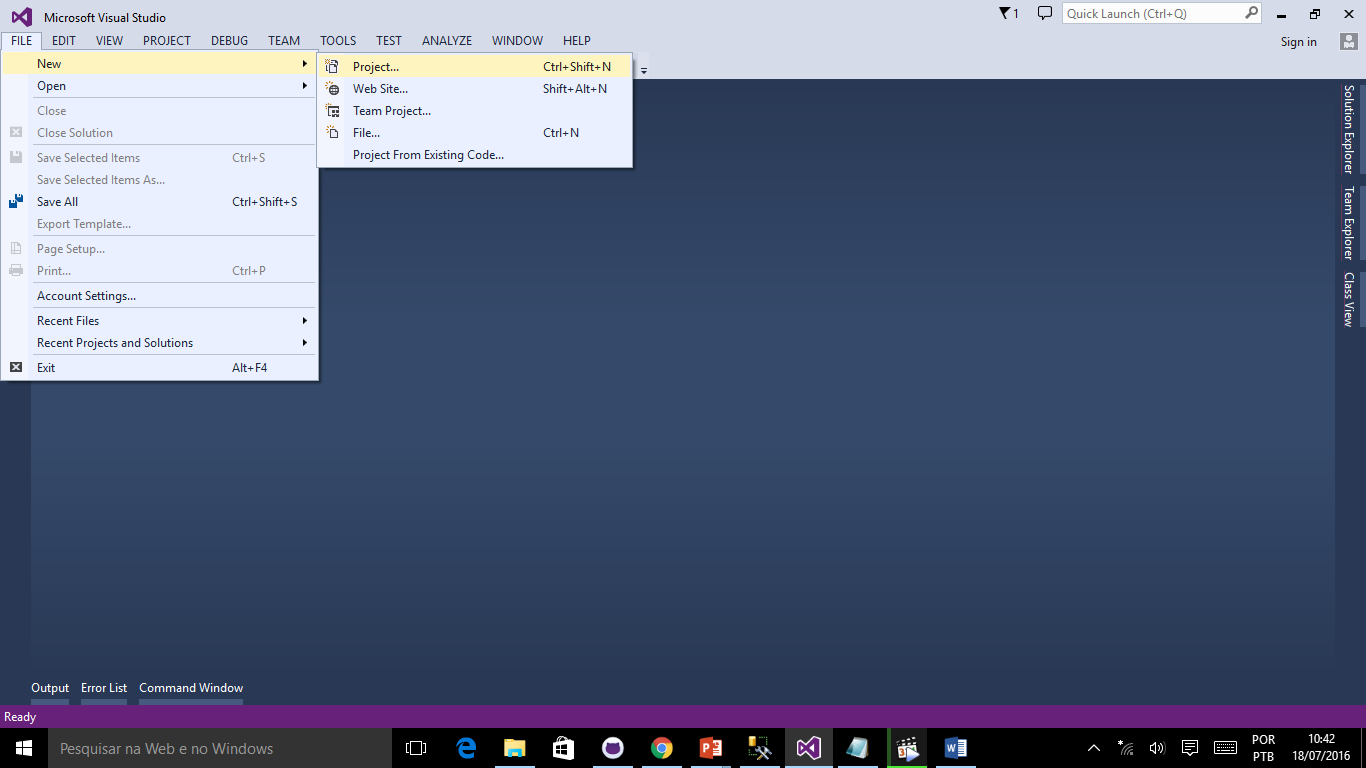
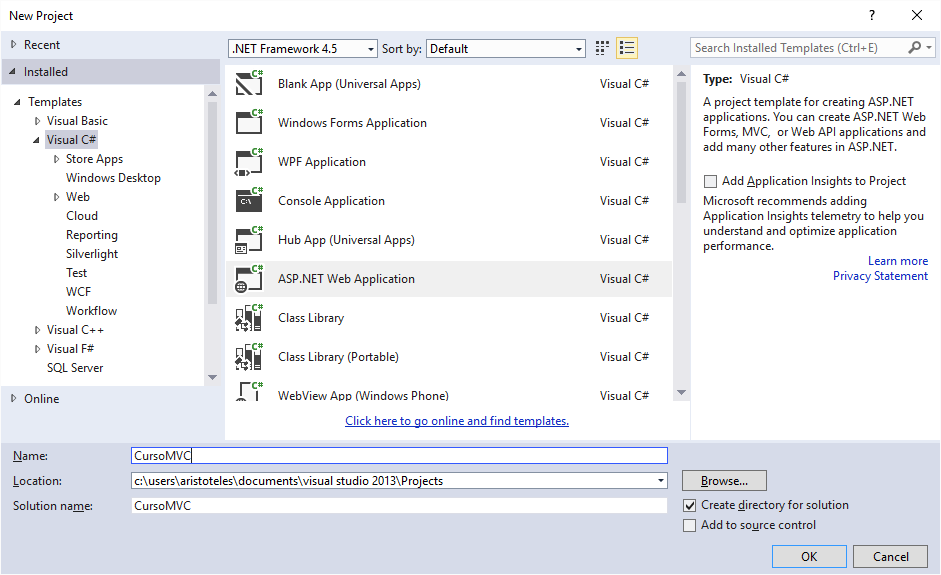
**Prática 11**

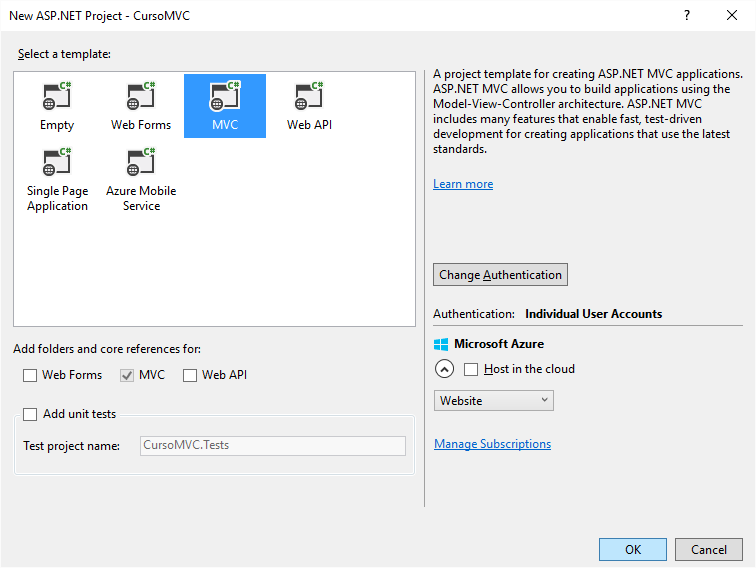
* Crie um novo item ao projeto



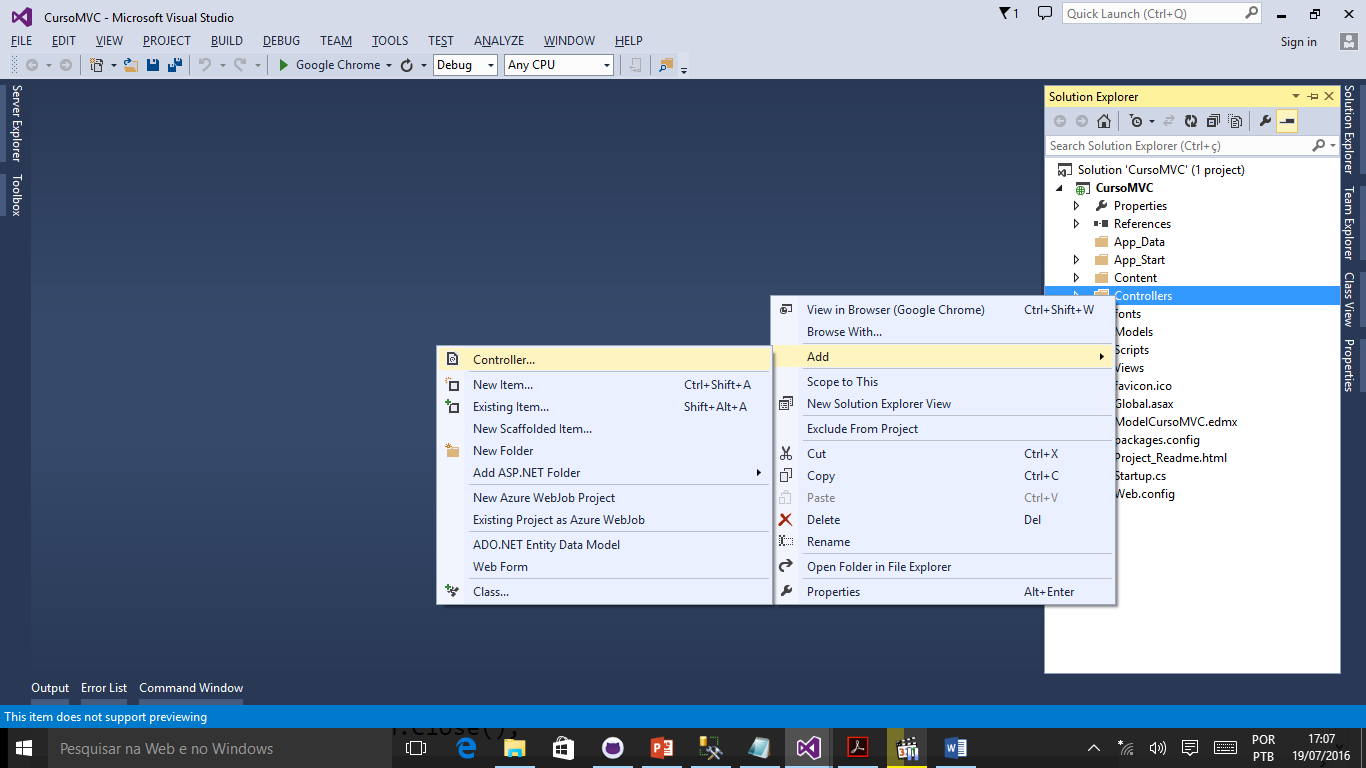
* Selecione ASP.NET Web Application e adicione o nome cursoMVC



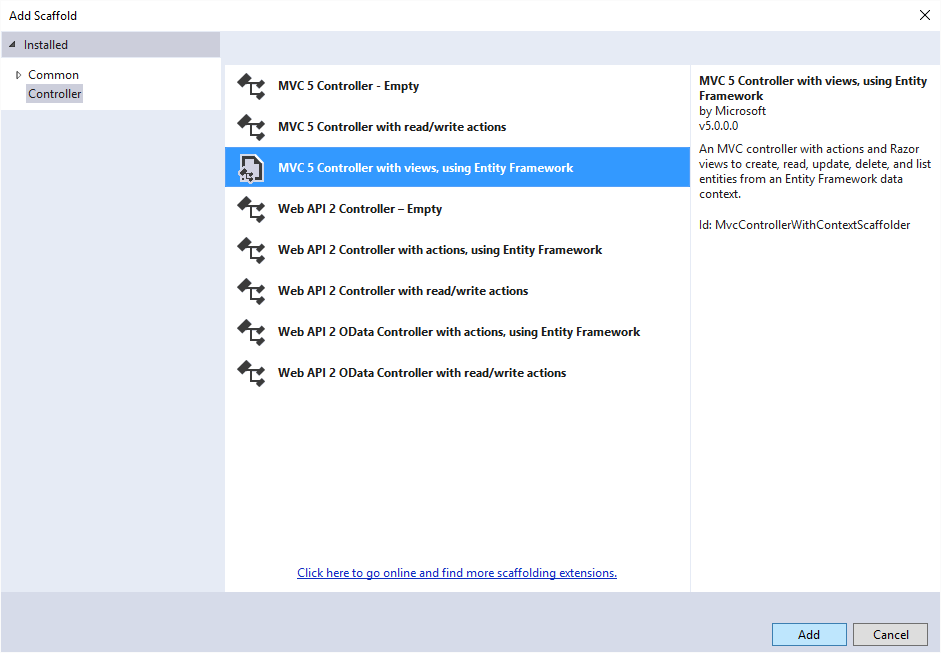
* Escolhe a opção MVC



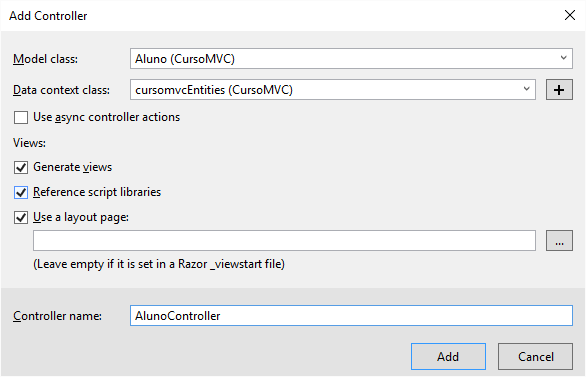
* Examine o projeto
* Adicione um ADO.NET Entity Data Model do banco de dados cursomvc
* Adicione um controller ao projeto



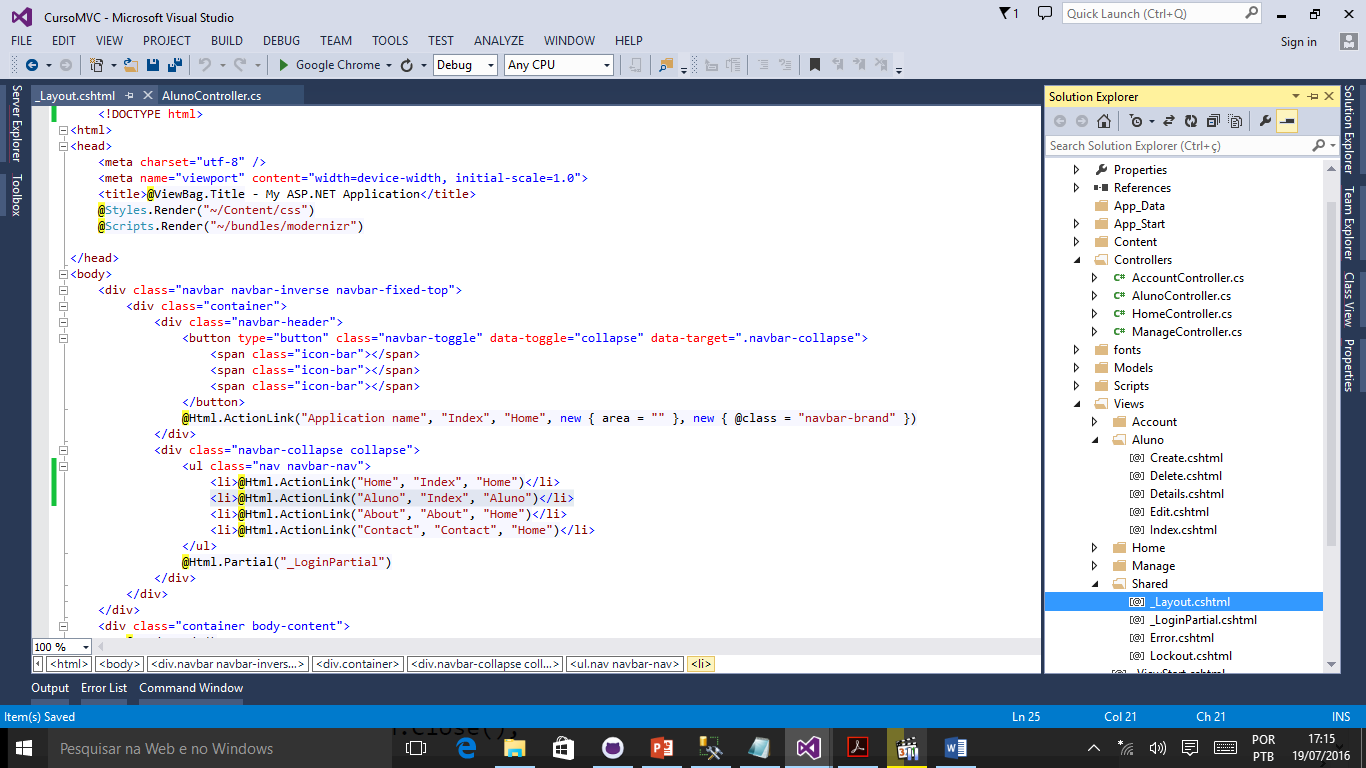
* Selecione a opção MVC 5 Controller with views, using Entity Framework



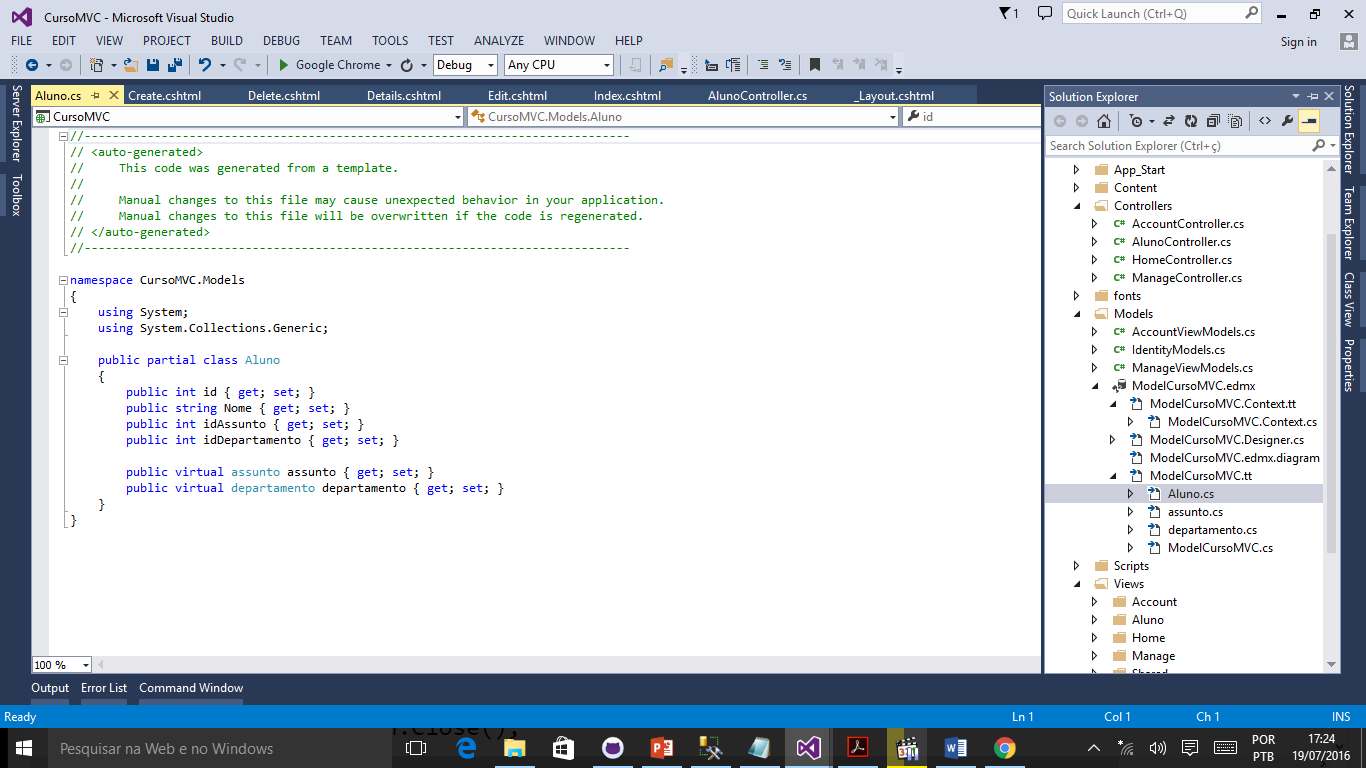
* Informe o nome o Model Class, Data contect class e Controller name



* Adicione a opção Aluno no menu



* Compile e teste
* Abra o Arquivo Alunos que se encontra na pasta Model



* Adicione as Data Annotations para alterar o título dos campos, antes importe a biblioteca DataAnnotation

using System.ComponentModel.DataAnnotations;

public partial class Aluno

{

public int id { get; set; }

[Display(Name="Nome")]

[Required]

public string Nome { get; set; }

[Display(Name="Código do Assunto")]

[Required]

public int idAssunto { get; set; }

[Display(Name="Código do Departamento")]

[Required]

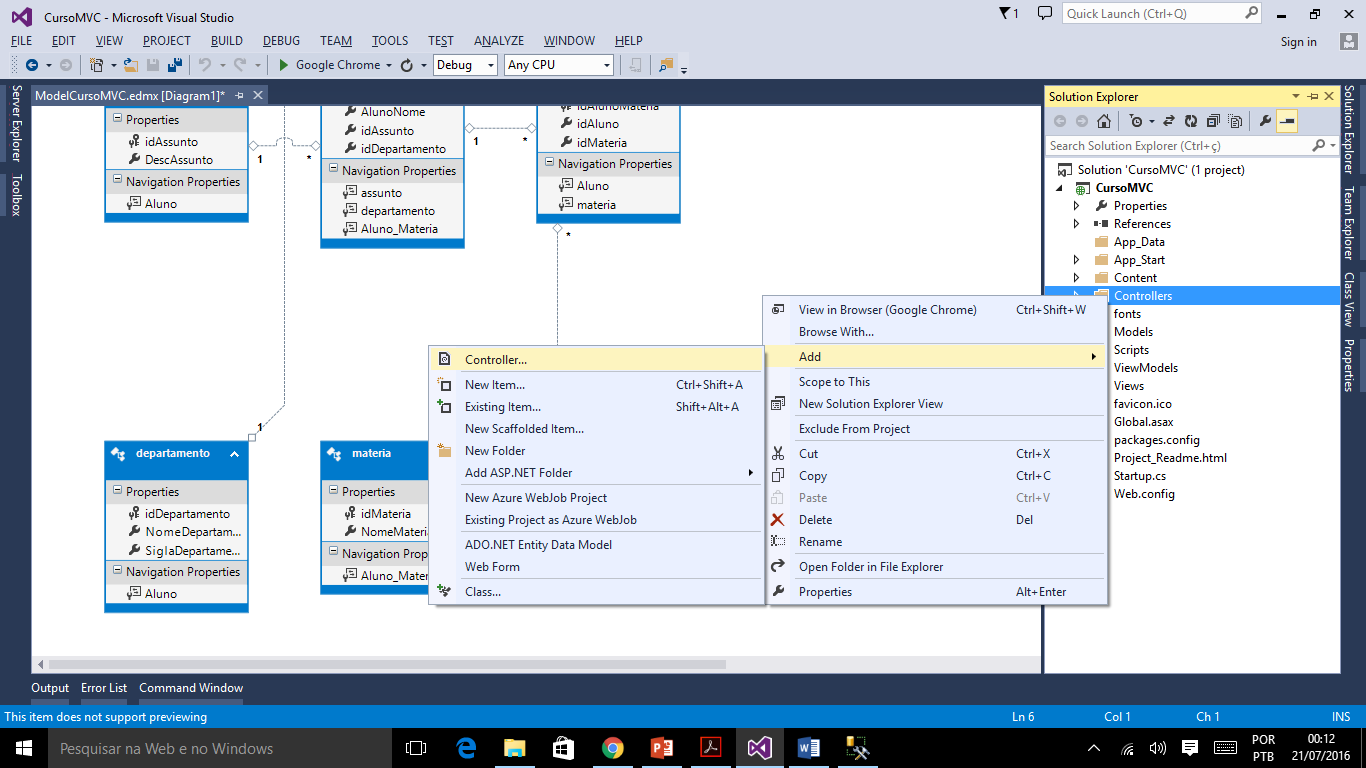
public int idDepartamento { get; set; }

public virtual assunto assunto { get; set; }

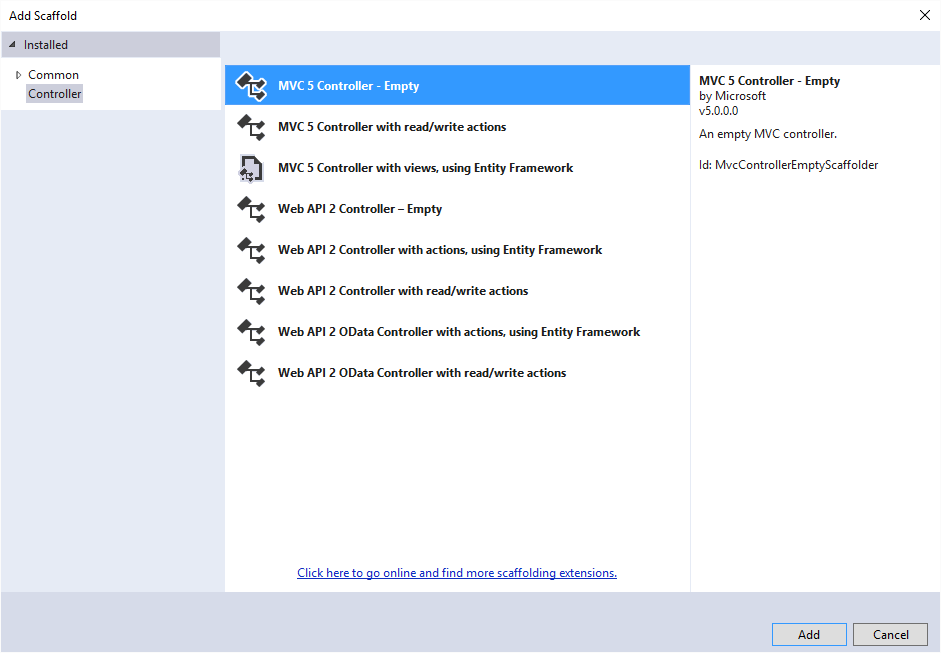
public virtual departamento departamento { get; set; }

}

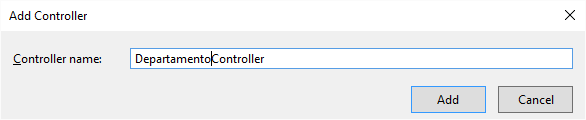
* Compile e teste
* Crie o Controller para Departamento



* Escolha a opção **MVC 5 Controller – Empty** para criar um Controller vazio



* Adicione o nome DepartamentoController



* Será exibida a classe DepartamentoController

namespace CursoMVC.Controllers

{

public class DepartamentoController : Controller

{

// GET: Departamento

public ActionResult Index()

{

return View();

}

}

}

* Crie um objeto do entity criado e no método Index() consulte e retorne a lista de departamentos

using CursoMVC.Models;

namespace CursoMVC.Controllers

{

public class DepartamentoController : Controller

{

cursomvcEntities1 db = new cursomvcEntities1();

// GET: Departamento

public ActionResult Index()

{

var departamento = db.departamento.ToList();

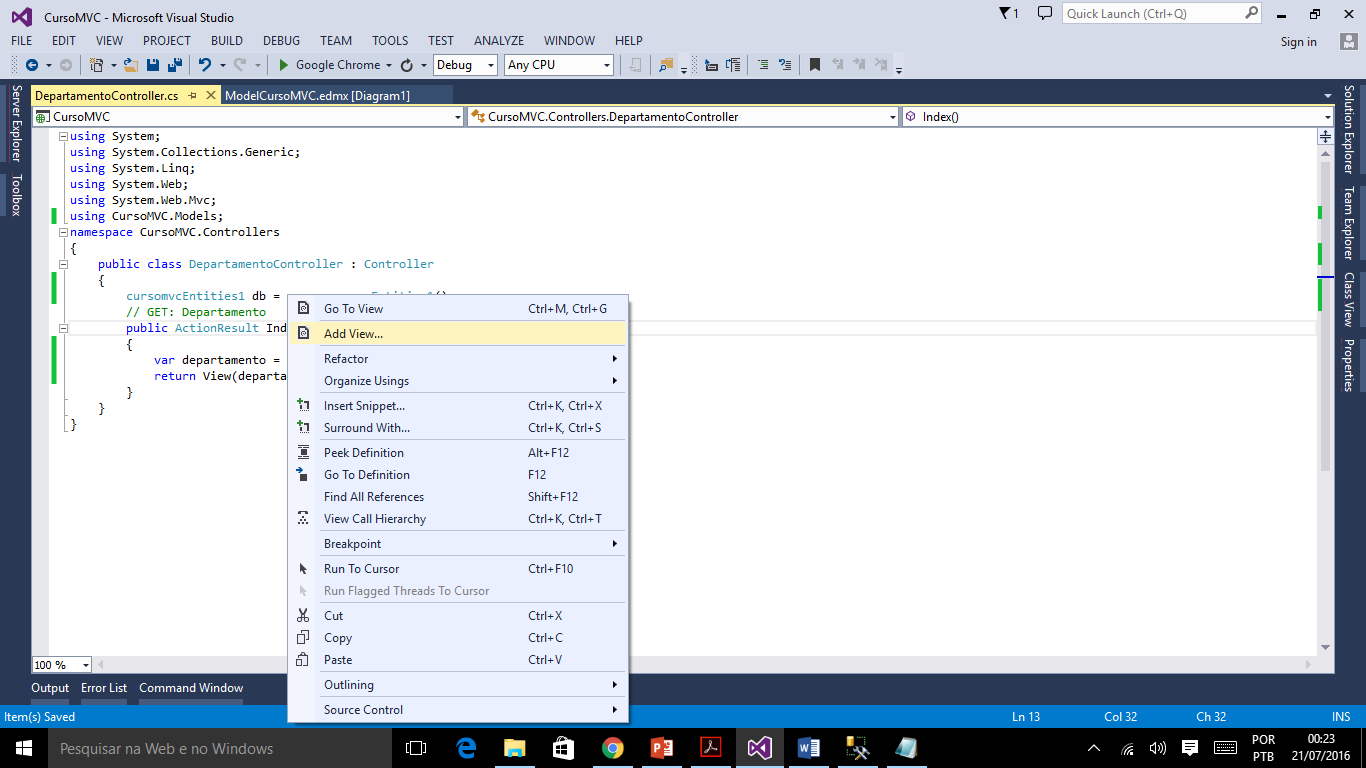
return View(departamento);

}

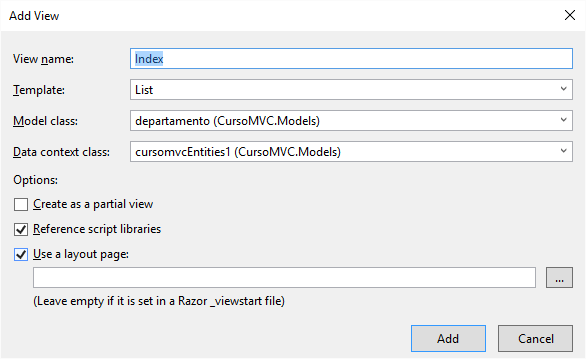
}

}

* Adicione uma view de Index a partir do Controller



* Informe os dados da view



* Será exibida as informações da view

@model IEnumerable<CursoMVC.Models.departamento>

@{

ViewBag.Title = "Index";

}

<h2>Index</h2>

<p>

@Html.ActionLink("Create New", "Create")

</p>

<table class="table">

<tr>

<th>

@Html.DisplayNameFor(model => model.NomeDepartamento)

</th>

<th></th>

</tr>

@foreach (var item in Model) {

<tr>

<td>

@Html.DisplayFor(modelItem => item.NomeDepartamento)

</td>

<td>

@Html.ActionLink("Edit", "Edit", new { id=item.idDepartamento }) |

@Html.ActionLink("Details", "Details", new { id=item.idDepartamento }) |

@Html.ActionLink("Delete", "Delete", new { id=item.idDepartamento })

</td>

</tr>

}

</table>

* Agora crie o método Edit na classe DepartamentoController

public ActionResult Edit(int id)

{

var departamento = db.departamento.Find(id);

return View(departamento);

}

* Adicione a view para Edit
* Na classe DepartamentoController adicione o método para fazer a alteração

[HttpPost]

public ActionResult Edit(departamento depart)

{

if (ModelState.IsValid)

{

db.Entry(depart).State = EntityState.Modified;

db.SaveChanges();

return RedirectToAction("Index");

}

return View(depart);

}

* Agora crie o método Delete na classe DepartamentoController

public ActionResult Edit(int id)

{

var departamento = db.departamento.Find(id);

return View(departamento);

}

* Adicione a view para Edit
* Na classe DepartamentoController adicione o método para fazer a remoção

[HttpPost, ActionName("Delete")]

public ActionResult DeleteConfirmed(int id)

{

var departamento = db.departamento.Find(id);

db.departamento.Remove(departamento);

db.SaveChanges();

return RedirectToAction("Index");

}

* Agora crie o método Details na classe DepartamentoController

public ActionResult Details(int id)

{

var departamento = db.departamento.Find(id);

return View(departamento);

}

* Adicione a view para Details
* Agora crie o método Create na classe DepartamentoController

public ActionResult Create()

{

return View();

}

* Adicione a view para Create
* Na classe DepartamentoController adicione o método para fazer a inserção

[HttpPost]

public ActionResult Create([Bind(Include = "idDepartamento, NomeDepartamento, SiglaDepartamento")] departamento departamento)

{

if (ModelState.IsValid)

{

db.departamento.Add(departamento);

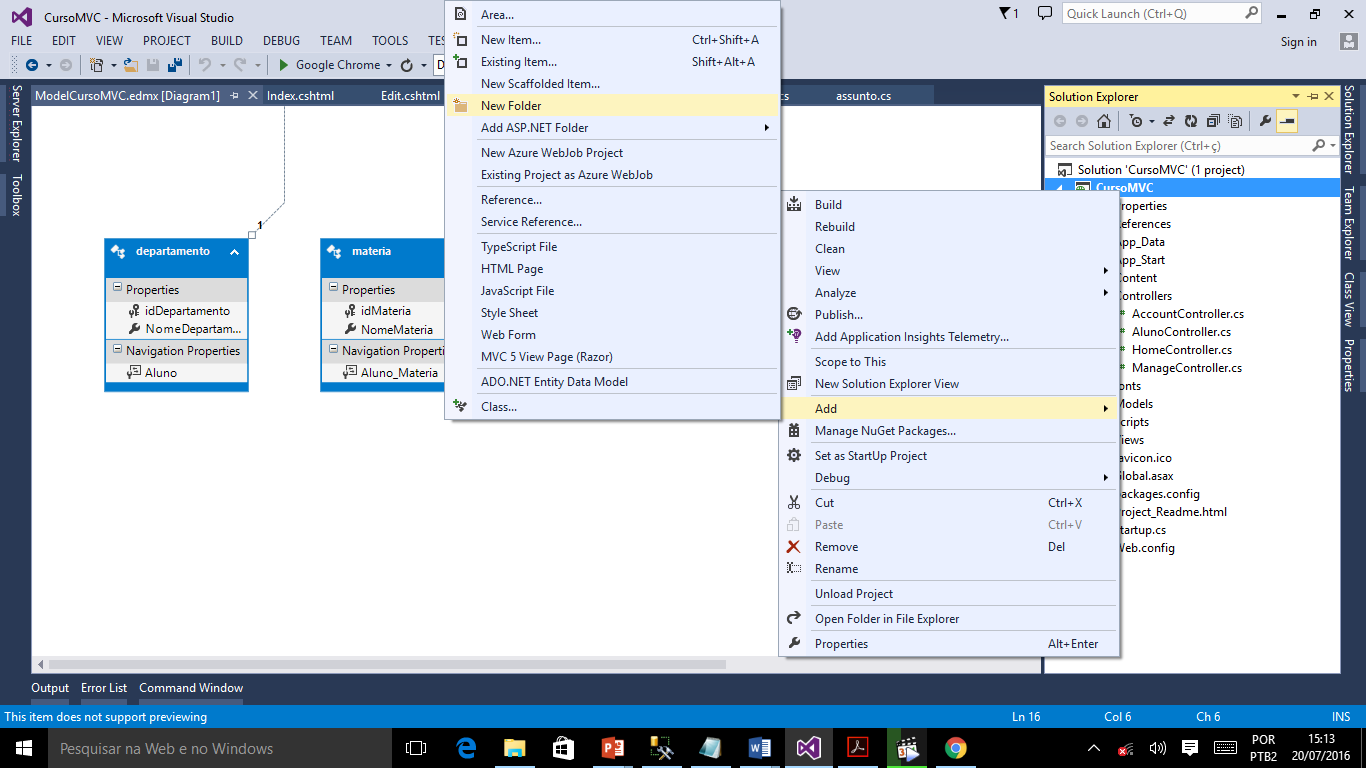
db.SaveChanges();

}

return RedirectToAction("Index");

}

* Crie a pasta ViewModels



* Crie a classe AlunoMateriaViewModel

using CursoMVC.Models;

using System;

using System.Collections.Generic;

using System.Linq;

using System.Web;

namespace CursoMVC.ViewModels

{

public class AlunoMateriaViewModel

{

public Aluno aluno { get; set; }

public Aluno\_Materia aluno\_Materia { get; set; }

}

}

* Abra o Arquivo Alunos que se encontra na pasta Model