In this video we will discuss editing a model in mvc. Please [watch Part 16](http://csharp-video-tutorials.blogspot.com/2013/05/part-16-difference-between-updatemodel.html), before proceeding.

**Step 1:** Copy and paste the following **"Edit"** controller action method in "EmployeeController.cs" file.  
[HttpGet]  
public ActionResult Edit(int id)  
{  
    EmployeeBusinessLayer employeeBusinessLayer =   
           new EmployeeBusinessLayer();  
    Employee employee =   
           employeeBusinessLayer.Employees.Single(emp => emp.ID == id);  
              
    return View(employee);  
}  
  
**Please note:**  
**1.** This method is decorated with [HttpGet] attribute. So this method only responds to HTTP get request when editing data.  
**2.** The **"Edit"** action method also receives "id" of the employee that is being edited. This "id" is used to retrieve the employee details.  
**3.** The employee object is passed to the view  
  
**Step 2:**Add "Edit" view  
**a)** Right click on the "Edit" controller action method, and select "Add view" from the context menu  
**b)** Set  
    View name = Edit  
    View engine = Razor  
    Select "Create a strongly-typed view" check box  
    Model class = "Employee"  
    Scaffold template = "Edit"  
    Finally click "Add" button  
**c)** This should add "Edit.cshtml" to "Employee" folder in "Views" foolder  
**d)** Delete the following scripts section that is present at the bottom of "Edit.cshtml" view  
@section Scripts   
{  
    @Scripts.Render("~/bundles/jqueryval")  
}  
  
Run the application and navigate to http://localhost/MVCDemo/Employee/Index. This page should list all the employees. Click on**"Edit"**link. The "Edit" page should display the details of the **"Employee"** being edited. Notice that, by default **"textboxes"** are used for editing. It is ideal to have a dropdownlist for gender rather than a textbox. To achieve this. make the following changes to **"Edit.cshtml"**  
  
**REPLACE THE FOLLOWING CODE**  
@Html.EditorFor(model => model.Gender)  
@Html.ValidationMessageFor(model => model.Gender)  
  
**WITH**  
@Html.DropDownList("Gender", new List<SelectListItem>  
    {  
    new SelectListItem { Text = "Male", Value="Male" },  
    new SelectListItem { Text = "Female", Value="Female" }  
    }, "Select Gender")  
@Html.ValidationMessageFor(model => model.Gender)  
  
Run the application. Edit an employee, and notice that a DropDownList is used for gender as expected. Post the form by clicking on "Save" button. We will get an error stating - The resource cannot be found. We will discuss fixing this in our next video.