

# MVC

## LEC1

-> الفرق بين الـ **web app** والـ **desktop app**

-الـ **web server** بنعمله **hosting** عن طريق **web site**

-اي جهاز ينفع يكون **server** المهم ينزل عليه الـ **sw** اللي هيشغله كسيرفر زاي الـ **IIS** للويندوز او الـ **apache** لـ **لينكس**

-في ويب سايتس بتعمل هوستينج واحدنا بنشتري باكيدج منها ونرفع الويب سايت بتاعنا ونأخذ **domain** او الـ **url** للسايت بتاعنا

- الكلاينت عنده البراوزر وهيكتب الـ **url** بتاع السايت وكدا البراوزر هي عمل **request** للسيرفر بالـ **url** وبرد بـ **response** فيه السايت بتاعنا

- وكدا اي تعديل ف السايت هنعدله عند السيرفير اللي عملنا عليه هوستينج وكدا كل اليوزر ز نقدر نشوف التعديل لكن الديسك توب لو في تعديل لازم كل يوزر ينزل الابديت الجديد بنفسه

-> انواع الـ **development type**

-اول حاجه الـ **full web app** بنعمل الـ **back** والـ **front** في نفس الـ **project** باستخدام الـ **MVC**

-ثاني نوع بنعمل الـ **back** لوحده بالـ **API** وبيكون ليه داتا بييز ونعمل كذا فرونت للموبايل والـ **disk** توب وكله يستخدم نفس الـ **back**

-> تطور صفحات الويب

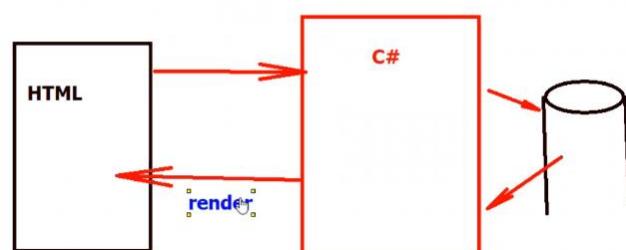
-ف البداية كانت **static pages** **html, css**

-بعد كدا **dynamic pages** باستخدام الـ **html, css, js**

-بعدين الـ **active pages** ونهحتاج داتا بييز وفي وسيط بين الداتا بييز وصفحات الـ **html** وهي الـ **C#** اللي هي **server** غن طريق الـ **EF core** او الـ **language side**

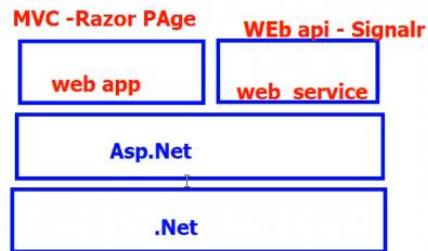
-الـ **html** بيكلم الـ **C#** وهي تكلم الداتا بييز وتأخذ الحاجه وترجع بيها للـ **html** بطريقة تفهمها الـ **html** العملية دي اسمها **rendering**

- 1) static Page (HTML + CSS)
- 2) dynamic PAge (html + css + JS)
- 3) active PAge



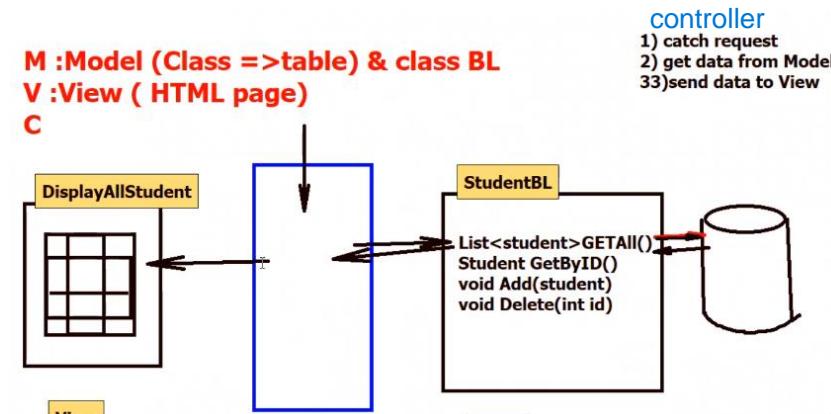
وoperating system cross platform open source frame work asp.net core يعني ينزل عليه web app يعني الاباک والفرونت او web services يعني البالک بس بيساعداني اعمل

**asp.net core**  
**open source web framework ,build web application ,web services**



الـ **MVC** عبارة عن design pattern الذي اين باترن دا عبارة عن طريقة بنكتب فيها الكود باننا نكتب كل جزء في layer منفصله علشان ميكتش معتمد على بعض لان في الاول كان الكود بيكتب كله في صفحة واحدة ودا كان صعب علشان كدا هنسخدم الـ **mvc** اللي هي اركيتكشن او ديزاين باترن

MVC اختصار ModelViewController والmodel (هيكون فيه الكلاسيز وكل كلاس بيتم على table في الداتا بيز واسم الكلاس بنفس اسم الجدول مثلا student) و نعمل كلاس تاني فيه البيزينس لوجيك واسمه يكون آخره BL مثلا studentBL بيكون فيه الفانكتشنز اللي هنعملها على جدول الطالب) والview (فيه الواجهه او البريزينتيشن لاي اللي هي صفحات الـ html) والوسيط بينهم هو الcontroller (عن طريق انه ياخد ريكويست من الـ view ويروح للـ BL يشوف الحاجه اللي عايزة اها الـ view ويعملها ريندر ويرجع بيها للـ view تاني)



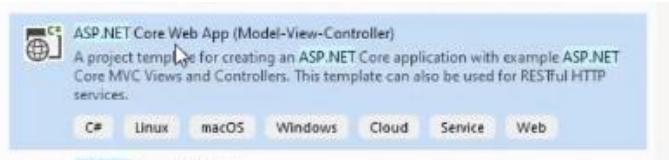
-من مميزات ال MVC: انه ي العمل separate of concern لأن كل حاجه منفصله في layer لوحدها

ثاني حاجه ال MVC بيعمل **routing map** يعني ال url مش هيكون فيه ولا extensions او paths ولكن ال url هيكون فيه اسم الكنترولير وبعد اسم الفانشن اللي هيئاديها من الكنترولير دا زاي مثلا /Emp/Index/ او /Index/Emp/ كدا ال Emp دا اسم الكنترولير و هيياميدي من جواه على ميثود اسمها index

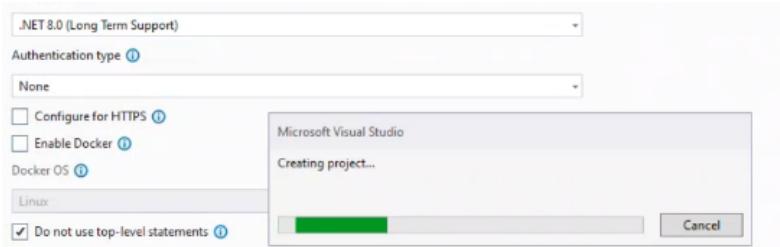
ثالث ميزة هي الـ **Razor engine** وهي ان في الـ view اقدر اكتب مع كود الـ html كود سى شارب برضو وبعدين هি�حصله ريندر ويتحول كله لـ html

## create mvc project-

-نستخدم 2022 vs نعمل create new project على asp.net ونختار اللي جمبها-  
model view-controller



وبعدين نكتب اسم الـproject والمكان وبعدين نختار الـcore ونلغي config for Https



-الفایلات الی موجود هم الفولدرز بتوع model , view, controller

- فایل الـ dependencies هيكون فيه الحاجات اللي هتخلي الـapplicaiton يرن زاي الـlibraries

- فایل الـ properties فيه json فایل بيعمل جواه هوستینج مش حقيقي علشان احنا لسه في مرحلة الـ development  
ودا لان الـ VS بینزل معاه IISExpress عن طريقه بيعمل virtual host

- فایل الـ wwwroot بيكون فيه كل الحاجات الاستاتيك وفایلات الـcss والبوتستрап والصور برضو علشان لو اتحطت  
برا مش هتنضاف

connection string دا json فایل والسي شارب بنتكتب فيه وتقرأ منه وبعدين هنحط فيه الـ appsetting

-آخر فایل هو program.cs دا فيه الـ main اللي جواها هي عمل build للـapplicaiton علشان يرن

--->لما نعمل كنترولير جديد لازم اسم الكلاس يكون اخره كلمة Controller وبيورث من كلاس Controller  
اللي بيخللي الكنترولير اللي بنعمله يعرف يستقبل ريكويست او بيعت ريسبونس ودا لا كلاس كنترولير نفسه وارث  
من كلاس ثاني اسمه ControllerBase ودا جواه HttpContext اللي بيتخط جواه الـHttpContextBase اللي بعنه الكلاينت  
والـRespones

```
public class HomeController : Controller
{
```

---> لما نعمل ميثود او اكشن جوا الـcontroller لازم تكون public وماينفعش تكون static او overload

-> الاكشن يقدر يرجع string و كل واحده عملنالها string, views, json,file, notfound  
و هكذا viewResult هيكون view وال ContentResult هيكون

```
//Action Return type
//String--> ContentREsult
//View -->ViewResulthtml
//Json -->JSonResult
//File -->FileResult
//notfound ->NotFoundREsult
//unauthor ->UnAuthoriResult
```

- علشان نرجع حاجه declare لازم نعمل contentResult بعدين initialize بأننا نكتب السترينج في  
return result.content

```
public ContentResult ShowMsg()
{
    //declare
    ContentResult result = new ContentResult();
    //initial
    result.Content = "HelloWolrd";
    //return
    return result;
}
```

- نفس الكلام للviewResult بس بنستخدم viewName و تأخذ اسم ال view اللي هترجعه

```
public ViewResult ShowView()
{
    //logic
    ViewResult result = new ViewResult();
    //declare
    //initl
    result.ViewName = "View1";
    //return
    return result;
}
```

- علشان يدور على view1 دي هيدور في فولدر views على فولدر باسم الكنترولر بتاعي اللي هو Home ويدور  
جواه على فайл اسمه view1 ولو مفيش دول هيدور في فولدر ال shared اللي جوا ال views ولو مفيش هيطلع  
صفحة اكسبيشن, لكن لو كتبت ال url غلط هيطلع صفحة notfound مش اكسبيشن

-> كدا كل حاجه ليها ريتيرن تايب مختلف ودا مش هينفع لأن في حالات ممكن الآيند بوينت او الاكشن يكون هيرجع  
انواع مختلف علي حسب الكونديشن

- علشان كدا كل الريتيرن تايب بيورثو من حاجه اسمها ActionResult اللي بت implementActionResult  
علشان كدا اي ريتيرن تايب هنعمله

- لو الاكشن بتاخذ باراميترز لما نبعتها ف ال url بنعمل ؟ بعد اسم الاكشن ونكتب الابار امسترز بينهم &

```
//home/showmix/1 just send id
//home/showmix/1?name=hassan
//home/showmix?id=1&name=hassan query string arrange not must
important
```

```

//Home>ShowMix?id=1&name=ahmed
public IActionResult ShowMix(int id, string name)
{
    if (id % 2 == 0)
    {
        ViewResult result = new ViewResult();
        //declare
        //initl
        result.ViewName = "View1";
        //return
        return result;
    }
    else
    {
        //declare
        ContentResult result = new ContentResult();
        //initial
        result.Content = "HelloWorld";
    }
}

```

-بدل م نكر اسطر ال declare و intial و return في ميثودز جاهزه زاي View بتأخد بس اسم ال view وهي  
هتعمل كل دا و ContentResult stringContent وترجع Json وفي contentResult و هكذا

```

//home/showmix/1 just send id
//home/showmix/1?name=hassan
//home/showmix?id=1&name=hassan query string arrange not must important
//Home>ShowMix?id=1&name=ahmed
public IActionResult ShowMix(int id, string name)
{
    if (id % 2 == 0)
    {
        return View("View1");
    }
    else
    {
        return Content("Hello");
    }
}

```

---> كدا خلصنا الكنترولر و هنشوف ال view

ال view بنعمل فايل الاكتشن بناعه cshtml . لان ممكن نكتب فيه سي شارب او html

علشان نكتب css جوا فايل ال view بنفتح تاج script ولل تاج style

لكن علشان نكتب سي شارب:

اول طريقة ال block code باننا نعمل @ {c# code} وبنكتب جواها كود سي شارب كامل والبلوك دا هيتنفذ عند السيرفر بس لكن مش هيظهر مع كود ال html

تاني طريقة ال inline statement ودي بتشاف في ال html

```

<script>
    console.log("Hello");
</script>
<!--Razor Engin--&gt;
<!--Block Code--&gt;
@{
    int x = 10 ;
    int y = x +10;
}

<!--Inline Statement--&gt;
&lt;h2&gt;@x&lt;/h2&gt;
@y
</pre>


arrow



```

@if(x>y)
{ <h1>x @x greater than y @y</h1>
else{<h1>x @x less than y @y</h1> }

```


```

@for(int x...){}

@(x+5)

نقدر نعمل برضو {} او @if{}@else{}@for{}@for{}

<--- كذا فاضل الـ Model: هنف على فolder الـ right ونعمل Add ونختار model ونكتب اسمه  
ودا هنكتب فيه المعلومات الأساسية

```
namespace Q3DotNetAssiut.Models
{
    public class Student
    {
        public int Id { get; set; }
        public string Name { get; set; }
        public string ImageURL { get; set; }
        public string Title { get; set; }
    }
}
```

بعد كذا نعمل كلاس تاني للبيزنس نعمل فيه الاكتشنز واسمها يكون اخره BL

```
public class StudentBL
{
    List<Student> students;
    public StudentBL()
    {
        students = new List<Student>();

        students.Add(new Student() { Id = 1, Name = "Ahmed", ImageURL = "1.png" });
        students.Add(new Student() { Id = 2, Name = "Mohamed", ImageURL = "1.png" });
        students.Add(new Student() { Id = 3, Name = "REwida", ImageURL = "2.jpg" });
        students.Add(new Student() { Id = 4, Name = "Alaa", ImageURL = "2.jpg" });
    }

    public List<Student> GetAll()
    {
        return students;
    }

    public Student GetById(int id)
    {
        return students.FirstOrDefault(s => s.Id == id);
    }
}
```

<--- دلوقتي هنحتاج نعمل controller لـ student اسميه controller

-هنعمل جواه ميثود او ايندبوينت اسمها showAll بترجع حاجه نوعها IActionResult وجوهاها عايزة يعرض صفة فيها كل الطلبة عن طريق ان الكونترولر هيبيت يجيده من الـ model وبيتعه للـ view وهي تلف على الـ list و تعرض لهم

-في الاول هنحتاج نعمل obj من الـ BL بتأتى الـ student عن طريقه اقدر انادي على ميثود الـ getAll اللي عملتها واخزنها في ليست عندي وبعدين عشان اعمل ريتيرن قولنا هنستخدم الفانكشن الجاهز اللي اسمها View ودي كانت بتاخذ اسم الـ view عايزة نعرضه و هتاخذ بارميتر تانى فيه اسم الليست اللي هتلف عليها تعرضاها

```
public class StudentController : Controller
{
    //Student>ShowAll
    public IActionResult ShowAll()
    {
        StudentBL studentBL = new StudentBL();
        List<Student> StudentListModel = studentBL.GetAll();//
        return View("ShowAll", StudentListModel); //Moedl List<student>
    }
}
```

-> علشان نعمل view هنعمل الاول فولدر اسمه student باننا نقف جوا اي ميثود ف الكنترولر زاي ونعمل view right clk ونختار add view و هو هي عمل فولدر باسم الكنترولر وبعدين نعمل فايل لل view اسمه showAll

- المفروض كنا بعثنا ليست مع اسم ال view وهي هستقبلها في بروبيرتي اسمها Model

- ال Model دا التايب بتاعه dynamic ودا مشكلته بيعدى اي حاجه في الكومبليشن لكن الايرورز هتظهر في ال run time وبرضو مفيش اوتو كومبليت لو عامله اوبجيكت من حاجه مثل

- علشان احدد نوع الحاجه اللي جايالي بكتب @model وجنبها نوع الحاجه اللي هتجيلي وال @model دي razor

- لكن ال Model هي اللي بروبيرني property engin مش



```

@model List<Student>






```

-> هنرجع للكنترولر نعمل اكشن ترجع الطالب حسب ال id



```

public IActionResult DEtails(int id)
{
    StudentBL studentBL = new StudentBL();

    Student studentModel = studentBL.GetById(id);
    return View("ShowDEtails", studentModel); //View ShowDet
}

```

- نعمل view اسمه showDetails وجواه نحدد التايب اللي جايالي عن طريق @model ونوعه student مش student من ليست من

**@model Student**

```

<h2>@Model.Name</h2>
<h3>@Model.Id</h3>


```

*LEC2*

: loosely type ↵<---

- دا اللي كنا بنقول الحاجه اللي بنبعتها للview من الموديل بتجيده في بروبيرتي اسمها Model ونوع Model دا كان dynamic وعلشان نحدد النوع استخدمنا @model ودى razor مش property

كل ال views بثورث من base class `RazorPage<T>` اسمه `Model` property وجواه ال ودي علشان كذا الكلاس كله جيندر ك

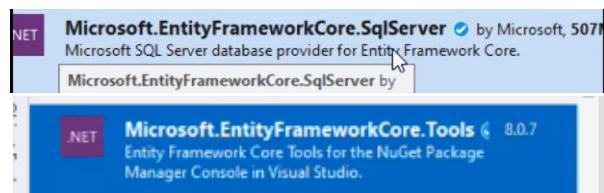
كدا ال view لما هيورث من ال RazorPage هيكون هو كمان generic بس مش هينفع تتساب ال view غير محددة نوع علشان مايكروسوفت لما تجي تعمل اوبحيكت من ال view لازم يكون في نوع محددينه علشان كدا هي هتشيل ال T بتاعت RazorPage وتعتبرها dynamic وبكدا ال view مش هيكون جينيريك وكدا عندي ال Model نوعها dynamic وبكدا هو loosely type

- علشان نخلية strong type view @model بنسخدم عن طريق ال razor هياخذ التایپ اللي جاي بعد ال dynamic @model ويحطه مكان ال RazorPage بتاعت ال

---> علشان نتعامل مع الداتا بيز في طريقتين الـ Entity framework او الـ ADO.NET ودا في منه طريقتين الـ database first او الـ code first

- احنا هنشتغل code first

-> نعمل رأيت كلk على اسم البروجكت ونختار manage NuGet علشان ننزل الباكتجيز اللي محتاجينها



-> علشان اعمل اكتر من كلاس مره واحده في الموديل هقف علي Models واعمل رايت كلک واختار add item و بعدين show compact view و اكتب اسمائهم بينهم .



- نعمل كلاس الدييار تميّنـت فيه الحاجات دي والـي جمبـه عـلـامـة ؟ معـناـهـا انه nullable يعني اقدر مكتـبـشـ الـقيـمةـ بتـاعـتـهـ

```
public class Department
{
    public int Id { get; set; }
    public string Name { get; set; }
    public string? ManagerName { get; set; }

    public List<Employee>? Emps { get; set; }
}
```

-اي بروبيرتي هسميه Id هي تعتبرها primary key لكن لو عايزه اكتبها اي اسم تاني وعايواها تكون هي ال pk  
خخط فوقها [key]

```
[Key]
public int EmpNatia { get; set; }
```

-هنعمل كلاس properties فيه ال employee اللي هنحتاجها وعلشان نربطه بكلاس ال department هنعمل int property obj department جوا department ولازم نربطه ب foreign key غن طريق اتنا نعمل اسمها مثلا ForeignKey("department") بنكتب جواها اسم الاوبجيكت اللي عملناه من Department ولو مكتبناش الاتريبيوت دا بس كتبنا اسم الكلاس اللي عملنا من اوبجيكت وجمهه ID هيعتبره علي طول بس علشان ممكن نغلط في حرف بنحط الاتريبيوت

```
public int Salary { get; set; }

public string JobTitle { get; set; }

public string ImageURL { get; set; }

public string Address { get; set; }

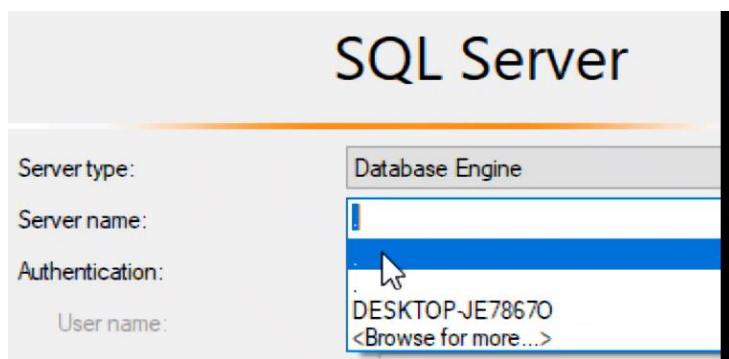
[ForeignKey("Department")]
public int DepartmentID { get; set; }

public Department Department { get; set; }
```

<--- هنعمل بعد كذا كلاس context يورث من DbContext وجواه نكتب اسماء ال tables عن طريق DbSet<className> بنكتب جواها اسم الكلاس اللي عايز اعمله جدول بعدين اكتب الاسم اللي عايزه للجدول وبنعمله نفس اسم الكلاس

```
public class ITIContext:DbContext
{
    public DbSet<Employee> Employee { get; set; }
    public DbSet<Department> Department { get; set; }
}
```

<- بعد كذا هنفتح sql server وبيطلب مني في البدايه 3 حاجات وهي السيرفير اللي مرفوع عليه نوعه واسمها وال authentication



---> جوا ال vs من ال view فوق اقدر اختار server explorer و هي فتحلي علي جمب كل الداتا بيز اللي هستخدمها  
 - علشان اضيف connection جديد هندوس رايت كليك على Data connections و نختار windows data source  
 - الـ windows مختار منها sql server وبعد كدا نختار server name اللي هو (. ) و نختار trust server certificate  
 - والـ encrypt authentication و اختيارها encryption و بعدين اختيار الداتا بيز اللي  
 عايز اضيفها عندي  
 - بعد كدا الداتا بيز اللي هستخدمها ادوس عليها رايت كليك واختار properties وهنلاقي فيهم ال  
 string  
 - هنرجع تاني للكلاس ال context و هنحتاج نعمل override للميثود onConfigure و دي بتأخذ  
 عن طريق ال optionsBuilder دا هقدر استخدم ال useSqlServer و دي هنحتاج اسم السيرفير و اس الداتا بيز  
 والـ authentication وكل دول موجودين في ال connection string اللي اخدهناه كوبى و هنغير عادي اسم الداتا  
 بيز اللي هي initial catalog

```

public class ITIContext : DbContext
{
    public DbSet<Employee> Employee { get; set; }
    public DbSet<Department> Department { get; set; }

    protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)
    {
        optionsBuilder.UseSqlServer
            ("Data Source=.;Initial Catalog=Assiut_DotNet_Q3;Integrated Security=true");
        base.OnConfiguring(optionsBuilder);
    }
}
  
```

الخطوه اللي بعد كدا هنحتاج نعمل ال default constructor علشان بعدين هنعمل parameter less constructor  
 و الـ diyfolt هنخليه يعمل على parent constructor اللي هو الـ base اللي هو بيت  
 DbContext

```

public class ITIContext : DbContext
{
    public DbSet<Employee> Employee { get; set; }
    public DbSet<Department> Department { get; set; }
}

public ITIContext() : base()
{
}

protected override void OnConfiguring(DbContextOptionsBuilder optionsBuilder)
{
    optionsBuilder.UseSqlServer
        ("Data Source=.;Initial Catalog=Assiut_DotNet_Q3;Integrated Security=true");
    base.OnConfiguring(optionsBuilder);
}
  
```

< بعد كدا من تحت هنعمل init add-migration و هي عمل الكلاس دا ---

```
namespace Q3DotNetAssiut.Migrations
{
    public partial class init : Migration
    {
        protected override void Up(MigrationBuilder migrationBuilder)...
        protected override void Down(MigrationBuilder migrationBuilder)...
    }
}
```

-هناقي في اتنين ميثود ال up و down

-لما اعمل update database بيفند ال up لو عايز ارجع في migration اللي عملتها هعمل remove update database و بيفند ال down بس لازم اكون معملتش migration

-في ال up هناقي عامل ال Employee وال المفروض عملنا ال FK من ال dept عند ال علىشان كدا هناقي تحت كل جدول ال constraints بقى حاجة في ال onDelete هي ال cascade لازم غيرها من noAction و دا علىشان ميعملش hard delete وال اللي يتمسح من جدول يمسح حاجات تبعه من الجدول الثاني وممكن كدا الجداول يحصل فيها خطأ بس لما نعملها soft delete هيبيقي noAction و اعمل انا عند الجداول تكون bool مثلاً توضح هو اتمسح او لا property

-> بعد كدا نعمل update-database

<--- بعد كدا في ال sql هندخل بيانات في الجداول ولازم ندخل ال first لان ال emp معتمد عليه

<--- بعد كدا هنعمل controller جديد لل Department

-هنعرف جواه او بجيكت من ال context

-بعد كدا في ميثود ال index هنعمل ليست من ال Department و هنجيب الليست دي باستخدام ال context اللي هتجبنا اللي دخلناهم في جدول ال dept و تعمليهم ToList والمفروض قبلها نعمل (dept.emps).include().Bas في طريقة احسن هنعملها بعدين

```
public class DepartmentController : Controller
{
    ITIContext context = new ITIContext();
    public IActionResult Index()
    {
        List<Department> departmentList =
            context.Department.ToList();
        return View("Index", departmentList); //Model List<department>
    }
}
```

-بعدين ننادي على ال View ونبعتله الليست دي علىشان يلف عليها ويعرضها

<--- علىشان كدا هنقف في الميثود بناء على index و نعمل رايت كلิก و نختار add view و بعدين نختار views المش ال empty وكذا هيعملنا فolder لل dept view وجوا ال views اسمه

-هي عمل فيه التاييل بالشكل دا و ممكن نغيره عادي و لازم نعمل @model و نحدد النوع اللي جابلي

```
@model List<Department>
{@
    ViewData["Title"] = "Department Index";
}

<h1>Index</h1>    I
```

---> لاحظ في كل مره استخدم ال url واكتب متلا `Department/Index`/controller factor / في حاجه اسمها هتعمل او بجيكت من dept ومن خالله تتمادي على index يعني عملت او بجيكت منها ومنه هتشوف كود السي شارب وتنفذه وتعمل ريندر لل view وترجعه html علشان تعرض الصفحة لليوزر

-وبعد كدا الكنترولر فاكتور هيعمل ديستروي لكل الاوجبيكت اللي عملها دي علشان يعمل serve علي حاجه جديده

---> لما كنا بنبع حاجه للview كنا بنعتها مع اسم ال view وهو بيستقبلها في البروبيوري Model بس كدا بنبع حاجه واحده بس ممكن نحتاج نبع اكتر من حاجه لل view

-الکنترولر لاما بنعمله بیورث من کلاس کنترولر و بیورث حاجه تانی dictionary و اسمها ViewData علشان کدا  
الدیکشنری دا لازم یکون لیه key و value

لما نيجي ننادي على اي اند بوينت من الكنترولر دا لو احنا باعтин حاجه لل view هياخدتها ويحطها في ال Model وبرضو هو من نفسه هيبعد لل view الديكتري اللي اسمه `viewData`

-**الـ `viewData`** دا جينيرك علشان اقدر ابعت فيه الحاجه اللي عايزة ها علشان كدا الـ `key` التایپ بتاعه هيكون **string** والـ `value` التایپ بتاعها **object** مش **dynamic** علشان بعدين اعمل انا **كاستينج**

### ViewData<string,object>

---> هنعمل كنترولير لل Employee واول حاجه نعمل اوبجيكت من الكونتيكست

بعدين ميثود اسمها details بتأخذ id وجوهاها بنعمل اويجيك من emp ونلف باستخدام الكونتيكت على الـ emps اللي عندي واجيب اللي الـ id بنطاعه ميعوت للميثود وهبعت الـ emp دالـ view وهستقبله في الـ Model

بعد كدا عندي كام معلومه تاني تحتاج بعثتهم مع الـ `emp` علشان كدا هستخد البروبيرتي اللي اسمها `ViewData` واكتب اسم للـ `key` واساويها بالحاجه اللي عايزة ابعتها وهو هبيعتها للـ `view`

```
ITIContext context = new ITIContext();
public IActionResult Details(int id)
{
    string msg = "Hello From Action";
    int temp = 50;
    List<string> bracnches= new List<string>();
    bracnches.Add("Assiut");
    bracnches.Add("Alex");
    bracnches.Add("Cario");
    //Additional info send to View from Action
    ViewData["Msg"] = msg;
    ViewData["Temp"] = temp;
    ViewData["brch"] = bracnches;
}
Employee EmpM0del= context.Employee.FirstOrDefault(e => e.Id == id);
return View("DEtails", EmpM0del);
```

-> بعدين هنعمل view لـ details

```
@model Employee
{
    ViewData["Title"] = "Details";
}



# Details



## @Model.Name



## @ViewData["Msg"]



## @(int.Parse(ViewData["Temp"].ToString())-10)



@foreach (var item in (List<string>)ViewData["brch"])
    {
        <option>@item</option>
    }


```

-هحدد النوع اللي جاي للـ Model ونستخدمه علشان نعرض الاسم عادي

-لما نحتاج نستخدم الحاجات اللي اتبعت في الديكشنري اللي اسمه ViewData هنستخدم برضو بروبيرتي في الـ ViewData ومنت عيوبها ان لازم نكتبها اسم الـ Key زاي اللي كتبناه بالظبط واحنا بنبعث وهو مش بيعمل اوتو كومبليت

-وتناني مشكله انه واخد القيمه علي انها object و بيرجع الحاجات علي انه string علشان كدا لو حاجه int لازم اعمل عليها ToString

-ولو باعتاله ليست لازم نعملها كاستينج برضو علي انها list<string> وبعدين نعمل foreach عليها ---> في حاجه تاني اسمها ViewBag

-دي هي هي ViewData يعني ديكتري بيأخذ اللي فـ ViewData ويرجعهولي علي انه dynamic بدل object علشان ماحتتجش اعمل كاستينج

-والـ razor في الـ view عدتها برضو زاي viewBag

-علشان استخدمها بنعمل ViewBag.key بنكتب اسم الـ key بعدها

-لو عملنا ViewBag ليه key معين واديناله قيمه وبعدها عملنا ViewData بنفس الـ key بنتابع viewBag هو كدا مش هي عمل key جديد لكن هي عمل override على القيمه اللي عند viewBag والعكس برضو

```
ViewBag.Color = "REd";
ViewData["Color"] = "Blue";
```

-من مشاكل الـ ViewBag برضو ان مفيش اوتو كومبليت ولازم نكتب اسم الـ key صح

---> في حالات بحتاج نعمل custom class او view model

-اول حاله لما نحتاج نبعث اكتر من موديل يعني مثلا هنبعث للview emp ومعاه حاجه من ال dept

-ثاني حاجه لو هبعث شوية بروبيرتيز من الموديل

-ثالث حاله لو عايشه ازود معلومات علي الموديل

---> علشان كدا هنعمل فolder جديد اسمه ViewModel

-ونعمل جواه كلاس باسم الحاجات اللي هنجمعها مع بعض واخره viewModel ونكتب فيه البروبرتيز اللي عايشهها

```
public class EmpDEptColorTempMSgBrchViewModel
{
    public string EmpName { get; set; }
    public string DeptName { get; set; }
    public List<string> Branches { get; set; }
    public int Temp { get; set; }
    public string Msg { get; set; }
    public string Color { get; set; }
}
```

-بعد كدا في كلاس ال emp هنعمل ايتد بوبينت جديد اسمها DetailsVM وبتاخد id وهنعمل جواها اوبيجكت من علشان نجيب الموظف بال id اللي مبعرفت للايد بوبينت ولازم نعمل include للdept بتاعه

-بعدين هنعمل اوبيجكت من ال viewModel اللي عملناه

-بعدين هنعمل mapping يعني نحط قيم للبروبرتيز بتاعت ال emp من ال viewModel بتاعنا

```
public IActionResult DetailsVM(int id)
{
    Employee empModel = context.Employee
        .Include(e => e.Department)
        .FirstOrDefault(e => e.Id == id);
    List<string> branches = new List<string>();

    branches.Add("Assiut");
    branches.Add("Alex");
    branches.Add("Cario");
    //declare viewmode
    EmpDEptColorTempMSgBrchViewModel EmpVM =
        new EmpDEptColorTempMSgBrchViewModel();
    //Mapping
    EmpVM.EmpName = empModel.Name;
    EmpVM.DeptName = empModel.Department.Name;
    EmpVM.Color = "REd";
}
```

-بعدين نبعث للview الاوبيجكت بتاع ال viewModel

```
new EmpDEptColorTempMSgBrchViewModel();
//Mapping
EmpVM.EmpName = empModel.Name;
EmpVM.DeptName = empModel.Department.Name;
EmpVM.Color = "REd";
EmpVM.Temp = 12;
EmpVM.Msg = "Hello FRom VM";
EmpVM.Banches = branches;
return View("DetailsVM", EmpVM); //EmpDEptColorTempMSgBrchViewModel
}
```

--- < بعد كدا هنعمل ال view بتابعه ---

- هنحتاج نحدد النوع من خلال viewModel @model ونكتب اسم ال viewModel اللي بعثنا منه او بجيكت

- بس لو كتبناه على طول مش هيس بيه لازم نعمل الاول @using لاسم المجلد بتابع viewModel

```
@using Q3DotNetAssiut.ViewModel  
@model EmpDEptColorTempMSgBrchViewModel
```

- بس بدل م نعمل السطر دا في كل view هنستخدم فيه ال viewModel الاحسن في فolder ال views هنلقي فайл اسمه viewImports هنحط فيه كل ال using اللي هنستخدمها علشان كل ال views ت Shawوفها

```
@using Q3DotNetAssiut  
@using Q3DotNetAssiut.Models  
@using Q3DotNetAssiut.ViewModel  
@addTagHelper *, Microsoft.AspNetCore.Mvc.TagHelpers
```

- وكذا نقدر نعمل ال view عادي ونستخدم الحاجات اللي بعثناها

```
@model EmpDEptColorTempMSgBrchViewModel  
  
{@  
    ViewData["Title"] = "DetailsVM";  
}  
  
<h1 style="color:@Model.Color">DetailsVM</h1>  
  
@Model.EmpName  
@Model.DeptName  
@<select>  
@foreach (var item in Model.Branches)  
{  
    <option>@item</option>  
}  
</select>
```

*LEC3*

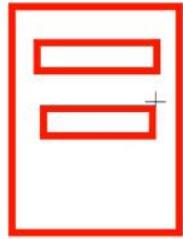
---> اللي فات كنا بنبعث داتا من الباك للفرونت وهن SHOW ازاي العكس بنبعث من فرونت لباك  
من الحالات اللي بحتاج ابعت فيها من الفرونت لباك زي شاشات الـ login او الـ registration او لو عامله CRUD  
وفي حاجه فيها اضافة او تعديل

-في الـ html لما نستخدم تاج الـ form دي بتحدد حاجتين للبلاك ايند وهي الداتا رايحة فين عن طريق attribute action وتنبيه حاجه بتحددتها ازاي الداتا هتتبعد عن طريق اtribut method اللي لو عملناه get بيقى الداتا هتتبعد في الـ query string

لما نعمل التایب بتاع المیثود يكون get بعد م اعمل submit للفورم هبيعت المعلومات بتاعت الاینیوتس كلهم في الurl بالشكل دا يعني في الـ query string

## **HTML Tags :**

```
<form method="Get" action="/Employee/Add">  
    <input id="Salary" name="Salary">  
    <input id="Age" name="Age">
```



**/Employee/Add?Salary=10000 & Age=12**

-هیشوف الاول نوع الميثود لو get هیبص على ال query string ولو post هیبص على body

-بعد کدا هیدور علی controller اسمه Employee

بعدين يدورجوا الكنترولير دا علي ميثود اسمها Add وبنأخذ اتنين بارميتيير هم salary , age ، و لو لقي الميثود دي هبيعت القيم من الريكيوبست يحطها في الابارميتيير بتوع الميثود دي

---> اللي اتعمل دا اسمه Model Binding

-> اول حاجه هنجربيها لما يكون التايب بتاع البارميترز اللي هيجي لهم الداتا من الفرونت يكونو primitive يعني او string

لو مثلاً عملنا كنترولير وفيه ميثرد عنده بتاخد name و age لو كتبنا url وبعثنا فيه name والage و حاجه تالتنه مع ان الميثرد بتاخد حاجتين بس..... اللي هيحصل ان الميثرد هتدور الحاجات اللي تحتاجها بس والباقي كأنها

جفانی  $\rightarrow$  جفری

لكن لو بعثنا حاجه واحده مثل ال name ومبغناش ال age ... كدا البارميترز اللي مش هتلقي قيمه ليها في(formData او ال default) هتحطها int بتعها لو null string queryString تخليه زورو ولو سيجمنت كدا مثل

```
//Bind/TestPrmitive?name=ahmed&age=12&id=10  
//Bind/TestPrmitive/10?name=ahmed&age=12
```

في الحاله دي قيمة id اخزنـت في ال RouteValues ولو بعثها بقيمتين مره تالت سيجمنت يعني في routeValues ومره في ال query string تكون الاولوية لل query string

لو نفس البارميـتر بعـتـالـه قـيـمـتـيـنـ فـيـ نـفـسـ الـ query stringـ هـيـشـوـفـ اـولـ وـاحـدـهـ بـسـ

لكن لو عايزـهـ حاجـهـ تـبـقـيـ لـيـهـ كـذـاـ قـيـمـهـ هـعـلـهـ arrayـ وـاـقـدـرـ اـحـدـهـ فـيـ الـ query stringـ بـنـاعـ كـلـ وـاحـدـهـ كـداـ

```
/TestPrmitive?color[1]=red&color[0]=blue
```

ولو في اينديكس مدخلـتوـشـ قـيـمـهـ هـيـعـتـبـرهـ null

كـداـ دـيـ اـولـ طـرـيـقـةـ لـلـ model bindingـ انهـ بـيـعـتـ الدـاـتـاـ فـيـ الـ formDataـ اوـ الـ query stringـ اوـ الـ routeValuesـ

---> تـانـيـ حاجـهـ مـمـكـنـ اـعـمـلـ bindـ عـلـيـ collectionـ زـايـ الـ listـ والـ dictionaryـ واـيـ حاجـهـ بتـimplementـ الـ Icollectionـ

الـ listـ تـعـالـمـهاـ زـايـ الـ arrayـ عـادـيـ

لكـنـ لوـ الـ الـ بـارـمـيـترـ بـتـاعـيـ عـبـارـهـ عنـ dictionaryـ لماـ بـعـتـهـ فـيـ الـ query stringـ بنـكتـبـ اسمـ الـ بـارـمـيـترـ وـبـدـلـ رقمـ indexـ بنـكتـبـ الـ keyـ الليـ هوـ اسمـ شـخـصـ مـثـلـ وـبـعـدـينـ الـ valueـ كـداـ

```
//Bind Collection  
//Bind/TestDic?Phones[Ahmed]=123&phones[chris]=456  
public IActionResult TestDic(Dictionary<string, string> Phones) {  
    return Content("Ok");  
}
```

كـداـ اـخـدـنـاـ اـزـايـ يـbindـ عـلـيـ دـاـتـاـ نـوعـهـ collectionsـ اوـ primitiveـ فـاضـلـ لوـ حاجـهـ نوعـهـ منـ objectـ

```
//Bind/TestObj?id=1&name=Sd&ManagerName=AbdelRahman  
public IActionResult testObj(Department deptObj)  
{  
    return Content("Obj");  
}
```

فيـ الحالـهـ دـيـ مشـ بـكـتبـ اسمـ الـ اوـبـجـيـكـتـ بـنـاعـ الـ deptـ فـيـ الـ urlـ وـلـكـنـ الدـاـتـاـ بـتـكـونـ عـبـارـهـ عنـ البرـوـبـرـتـزـ الليـ موجودـهـ جـواـ كـلاـسـ الـ deptـ

الـ dept عندها ليست من الـ emps لو عايزه ابعتهم في الـ query string هنعمل &emps[0].name=value

عادي نبعت بارميترز تاني مع الـ dept بس لو بعثنا باريتر مثلا name وكان الـ dept عنده بروبرتي اسمها name هبيع في الـ dept الجديد مش بناءت الـ dept علشان كدا المفروض نغير الاسماء متكتش زاي حاجات الأوبجيكت

---> دلوقتي احنا عندنا الـ view اللي بيعرض الـ department

```
@model List<Department>
{@
    ViewData["Title"] = "Department Index";
}
<h1>Index</h1>
<table class="table table-hover table-bordered">
    <tr>
        <th>No.</th>
        <th>Name</th>
    </tr>
    @foreach (var item in Model)
    {
        <tr>
            <td>@item.Id</td>
            <td>@item.Name</td>
        </tr>
    }
</table>
```

عايزين نعمل زرار لما ندوس عليها نقدر نضيف dept جديد .... يعني هيوصلنا لـ view تاني ندخل فيه المعلومات بناءت dept ودا هيكون عباره عن form واخرها submit علشان يأخذ البيانات ويعملها save في الداتا بيز .. يعني هحتاج نعمل view تانيي وانتنين اكشن واحد new وواحده ت عمل save

اول خطوه نعمل لينك في view بتاع الـ dept اسمه new والـ href بتاعه انه هيودينا علي كنترولير الـ Add ومنه علي اكشن اسمه department

```
<h1>Index</h1>
<a href="/Department/Add" class="btn btn-success"> New</a>
<table class="table table-hover table-bordered">
    <tr>
```

تاني خطوه نعمل الاكشن Add اللي يتوصلا لـ view جديد فيه الـ form

```
public IActionResult Add()
{
    return View("Add");
}
```

ثالث خطوه نعمل view جديد اسمه Add فيه ال form فيها اتنين input والاكشن بناء الفورم بيودينا لاكشن في الكنترولير اسمها SaveAdd والميثود ه تكون get

```

        ViewData["Title"] = "Add";
    }

    <h1>Add</h1>

    <form method="get" action="/Department/SaveAdd">
        <div>
            <label>Name</label>
            <input name="Name" id="Name" type="text" class="form-control" />
        </div>
        <div>
            <label>ManagerName</label>
            <input name="ManagerName" id="ManagerName" type="text" class="form-control" />
        </div>
        <input type="submit" value="Create" class="btn btn-primary" />
    </form>

    <a href="/Department/Index">Back To List</a>

```

must be like in model class

الخطوه اللي بعد كدا هنعمل في الكنترولير الاكشن اللي اسمها SaveAdd ... ودي هتاخد او بجيكت من dept ولازم تتأكد الاول ان الحاجات اللي جايه من الفورم مش null وبعدين نستخدم الكونتيكست علشان نعمل add ... بس كدا الحاجات اتعملها save في الميموري مش الداتا بيز... علشان كدا لازن نعمل context.SaveChanges() علشان ي save ف الداتا بيز

وبعد م يخلص المفروض يرجع لصفحة index اللي هتعرض كل dept لكن لو في غلط يفضل في صفحة add

```

public IActionResult SaveAdd(Department newDept)
{
    if (newDept.Name != null)
    {
        context.Department.Add(newDept);
        context.SaveChanges();
        return View("Index");
    }
    return View("Add");
}

```

اول غلطة عملناها اتنا عملنا "index" ودا لان ال index اصلا مستويه يجيela ليست من ال dept هتتخزن عندها في ال Model علشان تلف عليهم وتعرضهم واحنا مش باعтин حاجه لل view

ودي حلها اني اندى على ال view من ال view اللي انا فيه عن طريق RedirectToAction ولو ال view في كنترولير مختلف بكتب اسم ال view وبعده براميتر تاني لاسم الكنترولير بتاعه

```

public IActionResult SaveAdd(Department newDept)
{
    if (newDept.Name != null)
    {
        context.Department.Add(newDept);
        context.SaveChanges();
        //return View("Index", context.Department.ToList());
        //Call Action from another action
        return RedirectToAction("Index");
    }
    return View("Add");
}

```

-تاني غلطه احنا عملنا ان لو في حاجه null مش هيروح لصفحة الـ index ويفضل في الـ Add بس لما ندوس submit هناقي الفورم كلها فضيit واحنا عايزين تفضل الحاجات موجوده علشان يكمل عليها تاني

-احنا لما نعمل submit الدتا بتروح للميثود وتتخزن في الاوبجيكت بتاع newDept علشان كدا لما ارجع انادي Model بتاع view المفروض ابعتله الاوبجيكت اللي جالي ويستقبله هو فـ الـ Model

```
public IActionResult SaveAdd(Department newDeptFromRequest)
{
    if (newDeptFromRequest.Name != null)
    {
        context.Department.Add(newDeptFromRequest);
        context.SaveChanges();
        //return View("Index", context.Department.ToList());
        //Call Action from another action
        return RedirectToAction("Index");
    }
    return View("Add", newDeptFromRequest); //Model DDepartment
}
```

-وبعدين في الـ view في كل input value بتعاته هي القيمة اللي جايه من الـ Model

The screenshot shows the Visual Studio IDE. On the left is the code editor with the Add.cshtml view. It contains an @model Department directive and a form with two input fields: Name and ManagerName, both with their values set to @Model.Name and @Model.ManagerName respectively. On the right is the Solution Explorer window showing a project named "WebApplication1".

```
@model Department
@{
    ViewData["Title"] = "Add";
}



# Add


<!--/Department/SAveAdd?Name=Sd&ManagerName=Ahmed-->
<form method="get" action="/Department/SAveAdd">
    <div>
        <label>NAme</label>
        <input name="Name" id="Name" value="@Model.Name" type="text" class="form form-control" />
    </div>
    <div>
        <label>ManagerName</label>
        <input name="ManagerName" id="ManagerName" value="@Model.ManagerName" type="text" class="form form-control" />
    </div>
    <input type="submit" value="Create" class="btn btn-primary" />
</form>
```

-بس عندي مشكله تاني احنا خلينا ال View يجيده حاجه نوعها dept بس في اول مره هفتح الـ view مش ه يكون معايا اي اوبيجكت ارجعيه

```
public IActionResult Add()
{
    return View("Add");
}
```

-في حل اني ابعتله new Department() بس دا مش كوييس

-لكن الحل الاحسن اني في الفورم استخدم nullable operator

```

<form method="get" action="/Department/SAveAdd">
    <div>
        <label>NAme</label>
        <input name="Name" id="Name" value="@Model?.Name" type="text" class="form form-control" />
    </div>
    <div>
        <label>ManagerName</label>
        <input name="ManagerName" id="ManagerName" value="@Model?.ManagerName" type="text" class="form form-control" />
    </div>
    <input type="submit" value="Create" class="btn btn-primary" />
</form>

```

-هتغير نوع الميثود بتاعت الفورم نخليها post احسن علشان البيانات متظهرش في ال url وتتبع في ال formData

---> بعد م عملنا نوع الميثود تكون post ممكن حد يجي يكتب ال url باديه ويبعدت فيه البيانات ويدوس انتر من غير م يعدي على الفورم وكذا البيانات هتنبع للاشن بتاعي في ال query string مع اني عايزاه يتبع في الفورم داتا

-علشان كدا لازم فوق الاشن نحط اtribut يحدلي نوع الميثود هيكون post ولا اسمه get او filter ودا اسمه HttpGet

```

[HttpGet] ←
public IActionResult Add()
{
    return View("Add");//Model Null
}

//POST HANDLER
[HttpPost]
public IActionResult SAveAdd(Department newDeptFromRequest)
{
    if (newDeptFromRequest.Name != null)
    {
        context.Department.Add(newDeptFromRequest);
        context.SaveChanges();
        //return View("Index", context.Department.ToList());
        //Call Action from another action
        return RedirectToAction("Index");
    }
    return View("Add", newDeptFromRequest);//Model DDepartment
}

```

كدا لو حاولي بكتب ال url ب ايده ويدوس انتر هيدخل علي صفحة بتقول page not work right

---> دلوقتي هنعمل حاجه تاني.... هندخل علي كنترولير Employee وعايزين نعمل اشن اسمه index علشان يعرضلي بيانات الموظفين علشان كدا هبعت مع ال view لист الموظفين

```

public IActionResult Index()
{
    return View("Index", context.Employee.ToList());
}

```

بعد كدا نعمل view بيعرض كل الموظفين وجمب كل موظف هنعمل لينك اسمه Edit علشان لما ندوس عليه نقدر نعدل بيانات الموظف دا.... علشان كدا ال href بتاعت اللينك دا هتاخذ اسم الكنترولير والاشن اللي هيروح له اللي هنعمله اسمه Edit وهيأخذ برضو ال id بتاع الموظف اللي هنعدله

```


| ID       | Name       | Salary       |                                            |
|----------|------------|--------------|--------------------------------------------|
| @item.Id | @item.Name | @item.Salary | <a href="/Employee/Edit/@item.Id">Edit</a> |


```

found int Employee.Id { get; set; } Lm: 22

-هروح لكتنولير ونعمل الاكشن اللي اسمها edit وبيجيلاها id وهي هتجيب الموظف اللي ليه الid دا عن طريق firstOrDefault وبعدين تروح للview اللي اسمه Edit وتبعته الموظف دا

```

public IActionResult Edit(int id)
{
    Employee EmpModel = context.Employee.FirstOrDefault(e=>e.Id== id);
    return View("Edit", EmpModel);
}

```

-هنعمل الview ونعمل فيه فورم الميثود post والاكشن بتاعتها تودينا علي كتنولير الEmployee اكشن اسمها saveEdit ولازم نبعت معها id في الroute segment ... ولو مكتبنهوش مع الاكشن ممكن نعمل input والتابعه يكون hidden

```

<form method="post" action="/Employee/SAveEdit/@Model.Id">
    <input type="hidden" name="id" value="@Model.Id" />

```

-هنكمي الفورم ونعمل input لكل فيلد ونهعمل الvalue بتاعت كل input هتجي من الModel علشان تظهر له البيانات القديمه وهو يعدل عليها

```

<form method="post" action="/Employee/SAveEdit/@Model.Id">
    <input type="hidden" name="id" value="@Model.Id" />
    <div>
        <label>Name</label>
        <input type="text" value="@Model.Name" id="Name" name="Name" class="form-control" />
    </div>
    <div>
        <label>Address</label>
        <input type="text" value="@Model.Address" id="Address" name="Address" class="form-control" />
    </div>
    <div>
        <label>ImageURL</label>
        <input type="text" value="@Model.ImageURL" id="ImageURL" name="ImageURL" class="form-control" />
    </div>

```

must be like in the model class

```
<div>
    <label>Salary</label>
    <input type="number" value="@Model.Salary" id="Salary" name="Salary" class="form-control" />
</div>
<div>
    <label>JobTitle</label>
    <input type="text" value="@Model.JobTitle" id="JobTitle" name="JobTitle" class="form-control" />
</div>
<div>
    <label>DepartmentID</label>
    <input type="number" value="@Model.DepartmentID" id="DepartmentID" name="DepartmentID" class="form-control" />
</div>
<input type="submit" value="Save" class="btn btn-success" />
</form>
```

بعد كدا هنعمل الاكشن اللي اسمها saveEdit وهنعمل عليها اتربيوت ال `HttpPost` وهيجيلها اوبيجكت من `Employee` وممكن نبعت معاه id برضو علشان نستخدمه علي طول... وهنتأكد ان كل حاجه مش بـ `null` ... لو حاجه بـ `null` يرجعني علي صفحة ال `Edit` هي ويأخذ معاه الاوبجيكت زاي م بعنه علشان يكمل عليه

لو الحاجات مش بـ `null` هيروح للداتا بيز يجيب الموظف عن طريق ال `id` بعدين نعدل عليه ونساوي كل بروبرتي للموظف بالحاجه اللي جايالنا من الفورم... و في الآخر اعمل (`SaveChanges()`) و نعمل `RedirectToAction("Index")` اسمه

```
[HttpPost]
public IActionResult SaveEdit(Employee EmpFromRequest, int id)
{
    if (EmpFromRequest.Name != null)
    {
        Employee EmpFromDB= context.Employee.FirstOrDefault(e => e.Id == id);
        EmpFromDB.Address=EmpFromRequest.Address;
        EmpFromDB.Name=EmpFromRequest.Name;
        EmpFromDB.Salary= EmpFromRequest.Salary;
        EmpFromDB.JobTitle= EmpFromRequest.JobTitle;
        EmpFromDB.ImageURL= EmpFromRequest.ImageURL;
        EmpFromDB.DepartmentID= EmpFromRequest.DepartmentID;
        context.SaveChanges();
        return RedirectToAction("Index");
    }
    return View("Edit", EmpFromRequest);
}
```

عندنا مشكله ان اخر حاجه في الفورم عاليينها هي `DeptId` وكذا لما هندي نعدل بيانات الموظف هنختار ال `Dept` على اساس الرقم ودا مش حلو علشان مش شرط تقى عارفين ارقامهم احنا عايزين يظهرلنا درب داون ليست فيها اسماء ال `Dept` كلهم واحدنا نختار منهم

يبقى كذا `view` بتاعت `Edit` محتاجين نبعتها الموظف ومعاه ليست من ال `Dept` يعني كذا هنبعث اتنين لـ `view`

يبقى الحل ممكن نبعت الموظف في ال `Model` ونبعد ال `ViewBag` او `viewData`

او الحل الثاني نعمل ال `emp` مع ال `Dept` من `list` مع بعض في `viewModel`

هنروح لفولدر ال `ViewModel` اللي عملناه ونعمل ادد لكلاس جديد نسميه `EmpWithDeptListViewModel`

في الكلاس اللس عملناه مش هنعمل اوبيجكت من `emp` ولكن هنكتب كل البروبرتيز اللي موجوده في كلاس `emp` ونزود عليها ليست من `Dept`

```
{  
    public int Id { get; set; }  
  
    public string Name { get; set; }  
  
    public int Salary { get; set; }  
  
    public string JobTitle { get; set; }  
  
    public string ImageURL { get; set; }  
  
    public string? Address { get; set; }  
  
    public int DepartmentID { get; set; }  
  
    public List<Department> DeptList { get; set; }  
}
```

---> كدا هنحتاج نعدل ال view بتاع edit بدل م كان بيجيله في ال Model الموظف عايزين نبعته viewModel  
بس في اتنين اكشن ببنادو ال view اول اكشن اللي اسمه Edit اللي بيفتح الصفحه اول مره و الثاني اكشن SaveEdit  
اللي بيرحني لنفس الصفحه لو في حاجه داخله غلط  
-هنمسك اول اكشن اللي اسمها edit ... هعمل جواها او بجيكت من ليست من ال dept اجييه عن طريق الكونتيكت  
وبعدين نكريت view model ونعمل mapping يعني نساوي البروبيترز ببعض نجيب من الموديل الاصلی اللي  
ف الداتا بيز ونحط في ال viewModel و في الآخر نبعته لل view الـ viewModel

```
public IActionResult Edit(int id)
{
    Employee EmpModel =
        context.Employee.FirstOrDefault(e=>e.Id== id);
    List<Department> DepartmentList = context.Department.ToList();
    //-----Create View Mode Mapping
    EmpWithDEptListViewModel EmpViewModel=new EmpWithDEptListViewModel();
    EmpViewModel.Id = EmpModel.Id;
    EmpViewModel.Name = EmpModel.Name;
    EmpViewModel.Address = EmpModel.Address;
    EmpViewModel.ImageURL = EmpModel.ImageURL;
    EmpViewModel.JobTitle = EmpModel.JobTitle;
    EmpViewModel.Salary = EmpModel.Salary;
    EmpViewModel.DepartmentID = EmpModel.DepartmentID;
    EmpViewModel.DeptList = DepartmentList;
    return View("Edit", EmpViewModel);
}
```

-بعد كدا نعده view نخليه يستقبل viewModel ولازم نعمل @using لاسم الفولدر بتاع ال view model بس احنا عملناها قبل كدا ف فايل ال viewImports علشان كدا كل ال views هتشوفها

-تاني حاجه هنعدل في الفورم هنشيل ال input بتاع ال deptId ونعمل دروب داون ليست ب select و لازم نعمل deptId option بال كل value

-في حاجه تاني كدا دائمًا select هنقف على اول option بس احنا عايزين لما ندوس ايديت يقفلنا على الاوشن اللي بيمثل dept الموظف دا... علشان كدا هنعمل كونديشن لو ال item بتاع ال deptId اللي انا بلف عليه يساوي اللي جايلى من الموديل هنزوود اtribut selected على الاوشن دي لو مش بيساويه نعمل الاوشن من غير سيليكيد

```
<div>
    <label>DepartmentID</label>
    @* <input type="number" value="@Model.DepartmentID" id="DepartmentID" name="DeptID" />
    <select id="DepartmentID" name="DepartmentID" class="form-control">
        @foreach (var deptItem in Model.DeptList)
        {
            if (Model.DepartmentID == deptItem.Id)
            {
                <option value="@deptItem.Id" selected>@deptItem.Name</option>
            }
            else
            {
                <option value="@deptItem.Id" >@deptItem.Name</option>
            }
        }
    </select>
```

-بعد كدا نعدل الاكشن اللي اسمه SaveEdit علشان يستقبل viewModel بدل emp.. ولازم كل مره اساوي ليست بال ليست اللي جايها من الداتا بيز لأن احنا في الفورم عاملينها select input مش علشان كدا مش هنتبع لما ادوس submit مع باقي ال inputs

```
[HttpPost]
public IActionResult SaveEdit
{
    int id, EmpWithDEptListViewModel EmpFromREquest //Employee EmpFromREquest
    {
        if (EmpFromREquest.Name != null) ...
        //All Emp Data List<Department>
        EmpFromREquest.DeptList = context.Department.ToList();
        return View("Edit", EmpFromREquest);
    }
}
```

## LEC4

: Helpers <---

---> المحاضره اللي فاتت كنا بنعمل فورم علشان نبعث الداتا من ال view للكونترولير او ال بالك وكان لازم نعمل input لكل حاجه وتكون الاسماء مظبوطه

-اللي عملناه دا اسمه pure html

-لكن هنعمل بطريقة ثاني وهي pure c# ودي هي ال html helpers

-الـ view زاي م ورث اوبجيكت اسمه Model و viewData , viewBag ورث اوبجيكت برضو اسمه @Html

-اوبجيكت دا جواه ميؤود لكل تایپ من بتوع input زاي text بنعملها @Html.TextBox وال checkBox وال DropDownList لكن ال button بس مش موجود

-بعد كدا باقي الاتربيوتس اللي هم ال id,name,value هنأخذهم كبارميتر

### 1) Pure HTML

```
<input type="text" name="Address" id="Address" value="@Model.Address">
```

### 2) Pure C# 'HTML Helper'

→  
@Html.TextBox("Address")  
@Html.RadioButton  
@Html.CheckBox  
@Html.DropDown  
@Html.Password

-كل دول عباره عن فانكشن سي شارب وال razor بعدين يشيلها ويعلم ريندر بکود html

-الفانكشن دي علشان تدور علي القيمة بتاعت Address هتدور الاول في @viewData لو مفيش هتشوف @viewBag ولو مفيش تشويف Model.Address ولو مفيش خالص تسيبها فاضيه حسب النوع لو نامبر يحط صفر ولو تيكست يسيبها فاضي

-بس في مشكله برضو ان مفيش اتو كومبليت ولو غلطت ف حرف في كلمة Address كل حاجه هتبظ

---> النوع اللي عملناه دا اسمه loosely type helper

---> النوع الثاني اسمه strong type helper

-لازم ال view نفسه يكون سترونج تایپ عن طريق اننا نحدد النوع باستخدام @model

-بعد كدا بدل @Html.TextBoxFor(e=>e.Address) @Html.TextBox هنسخدم @Html.TextBoxFor(e=>e.Address) ودي بتاخذ لاما اكسبریشن مش سترينج

-لو عايزين نعمل style او نحط بوتسراب کلاس هنعمل بارميتر ثاني بعد اللما اكسبرشن وهنعمل فيه new کدا{} ونكتب ال anonymous object

Name
 @\* <input type="text" value="@Model.Name" id="Name" name="Name" class="form-control" \*@
 @\* @Html.TextBox("Name") \*@
 @Html.TextBoxFor(e => e.Name, new { style = "color:red" })

بس لو جربنا نديله `class` بنفس طريقة الـ `style` هيطبع ايرورز كتير علشان احنا دلوقي بنكتب سي شارب وكلمة `class` دي كلمة محجوزه عند السي شارب علشان كدا لازم نحط `@class`

```
@Html.TextBoxFor(e => e.Name,  
    new { style = "color:red", @class="form form-control"})
```

---> لو عايزة اعمل table برضو في helper ليها اسمه @Html.LabelFor(e=>e.Name)

```
@Html.LabelFor(e=>e.Name)
@Html.TextBoxFor(e => e.Name,
    new { style = "color:red", @class="form form-control" })
```

-باتاخد لاما اكسيرشن واللي هتعمله هتفتح الكلاس بتاع الحاجه اللي بيسخدمها وتزوح عند البروبيرتي اللي تبع الحاجه اللي عايزة اسمها name ويشوف هي مكتوبه ازاي بالضبط ويعرضها

-بس في حالات مثلا ممكن يروح لبروبيوريتي اسمها DeptID انا كتبها كدا في الكلاس بس مش عايزها تتعرض  
كدا في الفورم

-الحل هنحط اتريبيوت فوق كل بروبيرتى فى الكلاس عايزن نغير اسمها والاتريبيوت دا اسمه

```
[display("new name")]
```

```
 [Display(Name = "Full Name")]
public string Name { get; set; }
```

---> بس لسه الموضوع فيه صعوبه ولازم افker ايه انسب تایپ للبروبراٽي واختار TextBox او اي نوع

-علشان کدا عملو حاجه تاني هتاخد مني البروبيرتي بس وهي علي حسب نوع البروبيرتي هتقرر ايه انسب حاجه وتحططها

وـ `@Html.EditorFor(e=>e.Name)` هذا اسمه ويأخذ لاما فيه البروبرتي وهو بعدين هيحدد ايه نوع الاینبوت

بس، برضو الدوت نت متعر فش، كل التاليس، بناعت الـ html يعني، تاب ياسو، د مثلا الدوت نت متعر فوش

- علشان كدا هنستخدم اتربيوت تاني فوق البروبرتي اسمه [DataType()] ودا من خلاله هنقدر نحدد التایب اللي عايز بنه كدا

`[DataType(DataType.Password)]`

---> عندنا هيلبر برضو علشان الانكور تاج اسمه `@Html.ActionLink` بياخد اول بارميتر الكلمه اللي هتظهر مكان اللينك و الثاني بارميتر اسم الاكتشن اللي هيروح عليها ولو موجوده ف كنترولر مختلف نكتب بعدها اسم الكنترولر

```
<a href="/Employee/Index">Back To List</a>
@Html.ActionLink("Back", "Index", "Employee")
<br />
```

-بس كان في حالات بتحتاج نبتع ال id مثلا مع الاكشن اللي رايحله...في الحاله دي هيأخذ اوفر لوورد رابع لل new{id=Model.Id} و هنعمله بانتا نعمل routeValues ولو في حاجات تاني هيعتها احطها مع ال id

```
:f="/Employee/Details/@Model.Id?name=Ahmed">DEtails</a>
ActionLink("--Details--", "Details", "Employee", new{id=Model.Id, name=Model.Name})
```

---> كدا ال html helpers بتضمن ان ال binding هيحصل صح..بس مشكلتها هيعد ديزاين الـ html كله

-الحل نروح لثالث نوع وهو Mix ويعتمد على server side attribute

-يعني انا هستخدم تاج input عادي ويأخذ class براحته بس ال name,id, value attributes هنوعرض عنها server side attr

لکل تاج specific attributes خاصه بيه لوحده

-كل السيرفر سايد اتر بيوتس بتبدأ بـ asp وبعدين هيكل معايا

-لو مثلا تاج label هعمله ومش هكتبه كلمه يحطها لكن اعمل "asp-for="Name" وهو هيروح لل name ويشوف مكتوبه ازاي ويحطها ولو لقي فوقيه اتربيوت [Display] يحط اللي مكتوب فيه ولو انا كتبت كلمه ف تاج ال label هكتبها بدلهم... وتاج ال input برضو عنده asp-for علشان ال

```
<label asp-for="Name"></label>  
<input class="form-control" asp-for="Name"/>
```

لما نعمل انكور تاج عنده asp-action , asp-controller ولو في حاجات تاني هيعتها للاكشن احطها في

مثلا asp-route-id

```
<a href="/Employee/Details/@Model.Id?name=Ahmed">Details</a>
@Html.ActionLink("--Details--","Details","Employee",new{id=Model.Id,name=Model.Name})
<a asp-action="Details" asp-controller="Employee"
asp-route-id="@Model.Id"
asp-route-name="@Model.Name"
>DEtails</a>
```

---< نعمل صفحة لاضافة موظف باستخدام helpers

-نعمل ف view الـ index اللي بيعرض الموظفين انكور تاج اسمه New علشان يودينا علي صفحة اضافة موظف...الانكور هيكون عنده asp-action , asp-controller ,

```
@model List<Employee>
{
    ViewData["Title"] = "Index";
}

<h1>Index</h1>
<a asp-action="New" asp-controller="Employee">New</a>
```

-بعد كدا نروح علي الكنترولير نعمل اكشن اسمها New نخلي نوعها HttpGet

```
[HttpGet]
public IActionResult New()
{
    return View("New");
}
```

-بعد كدا نعمل view اللي اسمه New

-نعمل form و هيكون الـ default method post بقاعدتها get لكن في الـ html العادي من غير هيلبرز بيكون الـ default

-الفورم دي لما اعمل submit هتروج علي اكشن اسمها saveNew في كنترولير الـ emp و هي عمل label و input لكل بروبرتي...و يعمل زرار للـ submit والـ value بقاعدته create

```
@model Employee
{
    ViewData["Title"] = "New";
}

<h1>New</h1>
<form method="post" asp-action="SaveNew" asp-controller="Employee">
    <div>
        <label asp-for="Name"></label>
        <input class="form form-control" asp-for="Name"/>
    </div>
    <div>
        <label asp-for="Salary"></label>
        <input class="form form-control" asp-for="Salary"/>
    </div>
    <div>
        <label asp-for="Address"></label>
        <input class="form form-control" asp-for="Address"/>
    </div>
    <div>
        <label asp-for="ImageURL"></label>
        <input class="form form-control" asp-for="ImageURL"/>
    </div>
```

```

<div>
    <label asp-for="JobTitle"></label>
    <input class="form form-control" asp-for="JobTitle" />
</div>
<div>
    <label asp-for="DepartmentID"></label>
    <input class="form form-control" asp-for="DepartmentID" />
</div>
</div>

<input type="submit" value="Create" class="btn btn-success"/>

</form>
<a asp-action="Index" >Back To List</a>

```

بعد كدا نروح للكنترولير نعمل الاشن `saveNew` وعليه اتريبيوت `HttpPost` وهيجيله اوبيجكت من `emp`

```

[HttpPost]
public IActionResult SaveNew(Employee EmpFromREquest)
{
    if(EmpFromREquest.Name!=null && EmpFromREquest.Salary >= 6000)
    {
        //save
        context.Employee.Add(EmpFromREquest);
        context.SaveChanges();
        return RedirectToAction("Index");
    }
    return View("New", EmpFromREquest);
}

```

-هحتاج نعدل جزء اننا الفورم بنسخدم فيها `deptID` ودا مش كويس والصح نعمل دروب داون ليست باسماء الـ...`depts` كذا الـ `view` بباعت `new` هنكون مستنيه اوبيجكت من `emp` ومعاه ليست من `dept`...والمره اللي فاتت عملناها بطريقة اننا نعمل `viewModel` للاتنين ونستخدم اوبيجكت منه

-نهعمل المرادي طريقة اننا نبعث الليست في `viewData` في الاتنين اكشن الـ `new` و `SaveNew`

```

[HttpGet]
public IActionResult New()
{
    ViewData["DEptList"] = context.Department.ToList();
    return View("New");
}

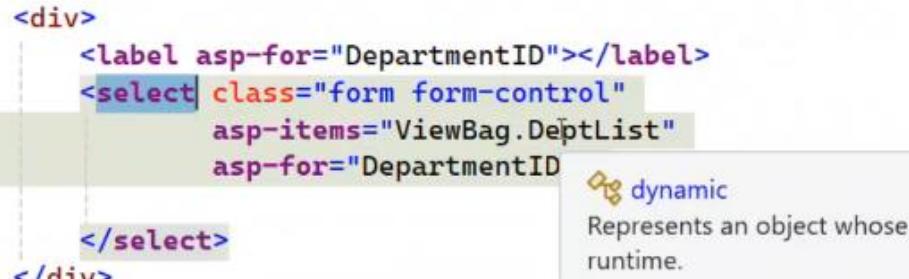
[HttpPost]
public IActionResult SaveNew(Employee EmpFromREquest)
{
    if(EmpFromREquest.Name!=null && EmpFromREquest.Salary >= 6000)
    {
        //save
        context.Employee.Add(EmpFromREquest);
        context.SaveChanges();
        return RedirectToAction("Index");
    }
    ViewData["DEptList"] = context.Department.ToList();
    return View("New", EmpFromREquest);
}

```

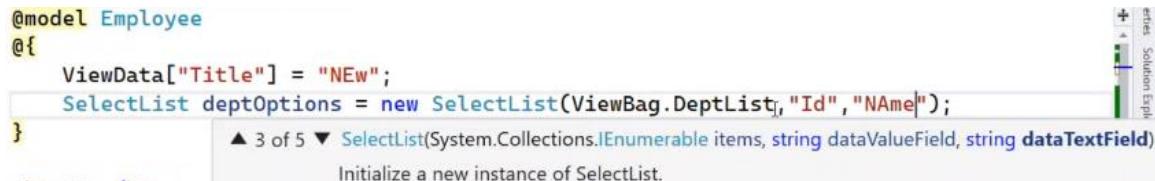
- هنرجع للforum وعايزين نشيل ال input اللي بيأخذ deptID ونعمل select تعرضنا ال depts

- هنجي لما نعمل ال select عندها هيلبر خاص ببها اسمه asp-items ودا المفروض هيأخذ الليست اللي هيلف عليها بس هو مستني IEnumarable من SelectListItem من dept

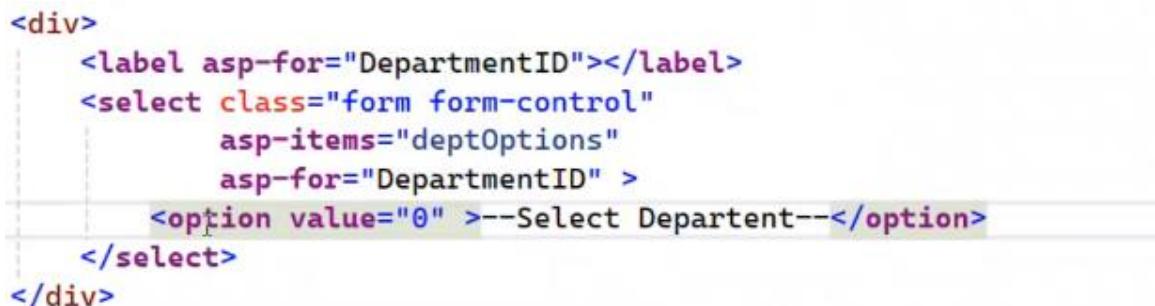
- هجيب ال deptList من deptViewBag علشان معملش casting مع اني بعاتها من الاكتشنز في ViewData بس قولنا عادي الاتنين واحد لكن ال viewBag بتعتبر تایپ الحاجه dynamic بدل object



- بس هو مش عارف يعمل ال casting دا وعايز انا اللي اعمله ... علشان كدا هنضيف من فوق تایپ عملته مايكروسوفت اسمه SelectList ونهنحوه الليست اللي عندنا للتایپ دا ... ببقي كدا هنبعث اول برامتر هو الليست .. وهتاخذ تاني بارميتر هو ال dataValueField وهذا هيأخذ ال id دا الحاجه اللي كانا هنحطها ف اtributot value في ال option ... وهتاخذ تالت بارميتر ال dataTextField ونهعمل Name دا الحاجه اللي كانت هتنعرض في كل option اللي هو dept.name



- وبعد كدا نجي في ال select ونحط ال deptOptions مكان الليست كدا

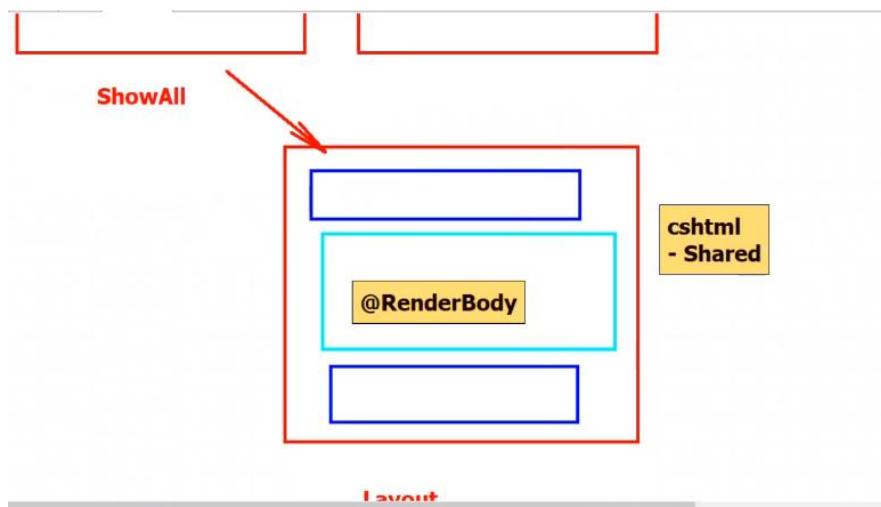


:Layout <----

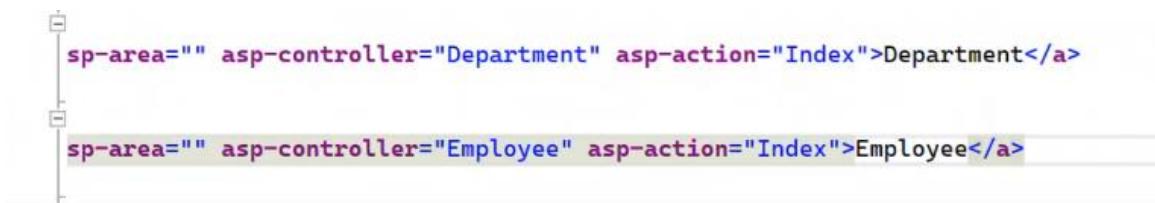
-هنالاحظ كل الـ views اللي بنعملها في header , footer ثابتين بيظهو مع اي صفحة والاختلاف بيكون في الكومنت بتاع ال body

-مايكروسوفت عملت مكان اسمه layout بتحط فيه الاجزاء اللي ه تكون مشتركة وتكرر مع كل view  
الـ layout دا فليل زاي اي view الاكتشن بتاعه cshtml . ومحظوظ في ال shared علشان يبقى منشاف على كل الـ views

-الجزء اللي بيتغير بيكون محظوظ ميثود سي شارب اسمه @RenderBody بتتشال ويتحط مكانها محتويات الـ view اللي عملته



-لو فتحت الـ Layout هنالاقى الـ nav bar فيه لينكات واقدر اضيف انا لينكات تاني مثلًا واحد يروح على صفحة الـ emp واحد للـ dept



ما بين الـ header والـ footer هنالاقى main tag فيها @RenderBody() عاملنا

```
<div class="container">
  <main role="main" class="pb-3">
    @RenderBody()
  </main>
</div>
```

-الـ renderBody يكون مره واحده بس

-بس ممكن يكون في صفحات عايزة ازود فيها جزء على الـ header او الـ footer .. ودا علشان نعمله نستخدم  
@renderSection ودا ممكن يتكرر عادي مش زاي الـ

-علشان نعمل احنا layout .. هندوس علي فولدر الـ shared رايت كليك ونختار add item بعدين all  
layout ... بعد كدا من الجمب نعمل سيرش علي razor layout ونختار tempates ونكتب اسم الـ  
under score(\_name.cshtml) والاحسن نبدأه بـ (\_name.cshtml)

-هي عملني صفحة html وفي البادي عامل div فيها () ... هنضيف احنا الاجزاء اللي عايزينها  
تكون ثابتة

```
<!DOCTYPE html>
<html>
  <head>
    <meta name="viewport" content="width=device-width" />
    <title>@ViewBag.Title</title>
  </head>
  <body>
    <div style="height:100px;background-color:#burlywood;font-size:40px">Header</div>
    <div style="height:300px;background-color:#darkcyan;font-size:40px">
      @RenderBody()
    </div>
    <div style="height:50px;background-color:#burlywood;font-size:40px">Footer</div>
  </body>
```

-بعد كدا نروح لـ view عايزين نضفه الـ layout باتاعنا و هناقي ان الـ view وارث من الـ razor البروري  
الـ layout اللي اسمها Layout ونكتب فيها اسم الـ layout

```
@{
  ViewData["Title"] = "Privacy Policy";
  Layout = "_MyLayoutB";
}
```

-كدا هي شيل الـ default layout و هيحط الـ layout اللي عملناها

-لو عملنا "layout=null" مش هيحط اي حاجه

-قولنا نقدر نعمل كذا renderSection بس كل واحده لازم تاخذ اسم مختلف

```
<body>
  <div style="height:100px;background-color:#burlywood;font-size:40px">Header
    <div style="background-color:#darkcyan;">
      @RenderSection("HeadSection")
    </div>
  </div>
  <div style="height:300px;background-color:#darkcyan;">
    @RenderBody()
  </div>
  <div style="height:50px;background-color:#burlywood;font-size:40px">Footer
    <div style="background-color:#darkcyan;">
      @RenderSection("FooterSection")
    </div>
  </div>
```

-بعد كدا في صفحة ال view اي حاجه بكتبها ف الفايل بيعتبرها تبع ال renderBody...لكن لو في جزء عايزه يكون تبع سيكشن معين هعمل {@section sectionName{}}

-طالما عملت في ال layout اي سيكشن لازم في ال view اكتب حاجه للسيكشن دا مانيفعش نسيب واحد منهم ودان ال required انه default

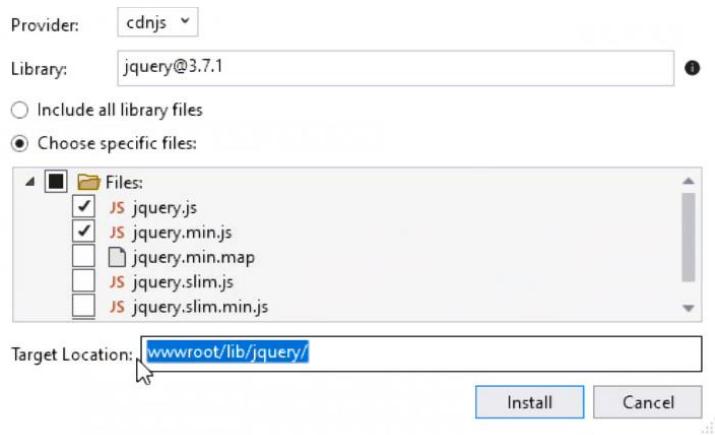
-لكن اقدر ادي السيكشن قيمة bool تعرفني انه required او لا...ال انه default يعني required لكن لو كتبت optional هيقي false

```
<div style="height:50px;background-color:#burlywood;font-size:40px">Footer
    <div style="background-color:#darkcyan;">
        @RenderSection("FooterSection",false)
    </div>
<@{
    ViewData["Title"] = "Privacy Policy";
    Layout = "_MyLayout"; //null
}
<!--RenderBody-->
<h1>@ViewData["Title"]</h1>
<p>Use this page to detail your site's privacy policy.</p>
@section HeadSection{
    <h3>Msg From View To Head Section</h3>
}
@*
@section FooterSection {
    <h3>Msg From View To Footer Section</h3>
} *@
```

---> دلوقتي عايزين ننزل client side library ... لما كنا بننزل حاجه من ال NuGet كانت بتتحط في ال dependencies...بس احنا لو هننزل حاجه زاي jQuery مثلاً محتاجينها تكون في ال wwroot زاي البوتراب

-علشان كدا هنعمل رايت كلّك علي اسم البروجيكت...ونعمل add ونختار client-side lib

-يفتح الصفحة دي وه يكون فيها ال provider هو ال cdnjs .. وهنكتب احنا اسم ال lib اللي عايزينها . ولو مش عايزين ننزل كل حاجه ندوس ثاني اختيار ونختار الحاجات اللي محتاجينها .. ومن تحت نختار المكان اللي تنزل فيه اللي هو عامله wwroot



-لو عندي views كتيره وعايزه ال layout بنتاعتي تتحط عليهم كلهم يعني عايزها تكون ال default ..هناقي في views في فайл اسمه viewStart كل اللي موجود فيه سطر واحد يحدد ال layout

```
@{  
    Layout = "_Layout";  
}
```

-لو عايزين ننزل template جاهزه في سايت اسمه start bootstrap ننزل منه

-هينزل في فولدر فيه صفحة Html ومعاها css,js,assets

هناخد فولدر ال assets ونحطها في wwwroot ...هتفتح فولدر Css ونأخذ اللي جواه ونحطه في فайл css اللي عند ال wwwroot وبرضو ال js نفس النظام نحطهم مع بتوع

-فاضل صفحة ال html هفتحها واحد الكود كوبى... وبعدين في فولدر ال shared نعمل layout جديد ونحط فيه الكود اللي عملناه كوبى...ونعدل علي الكود دا زاي م احنا عايزين ولازم نضيف ال @RenderBody ف المكان اللي محتاجينه ولو عايزين نضيف renderSections براحتنا

-برضو هنروح لكل اللي href بناعها بببدأ بكلمه علي طول زاي حاجات css وال js والصور ونببدأها بـ

```
<link href="/css/styles.css" rel="stylesheet" />
```

-بس لو عندي حاجات كتير بتبدأ ب assets زاي الصور هنعمل quick replace ونكتب كلمة assets ونبدلها بـ /assets

--- اعمل سيرش على ال pagination

## LEC5

: Validation <----

-اول نوع validation هو client side validation

-دي اسرع انواع ال validation وتحصل عند client وبتعمل في ال html

-ثاني نوع هو server side validation بيعمل في ال c#

-في الاكواود اللي فاتت كنا بنعمل ال check في الاختصن نفسها نقول مثلا if(name!=null&&age>20) بس دا مش كويش لأن الكلينت مش هيكون شايف الكونديشنز دي

---> في ال MVC فاليديشن بتعمل في 3 فايلات في ال controller view وال model

-اول حاجه في فايل ال Model: بنحط ال criteria او الكونديشن بتاعتنا زي اني عايز ال name ميكنش null وال age يكون اكبر من قيمة معينه

-ثاني حاجه في ال view: بنعرض فيه رسالة الایورر اللي تظهر لليوزر لو دخل حاجه غلط

-ثالث حاجه ال controller: هيجيده البيانات اللي جايها من الموديل وهو يcheck ان الموديل معدى فاليديشن والكونديشنز اللي عملناها

---> عندنا مثلا موديل لкласс student

لما نعمل فاليديشن على الاسم مثلا: لازم يكون ليه فورمات معين باستخدام ال regular expression ويكون ريكوايرد يعني not null وليه min,max length لازم يكون ريكوايرد ويكون ليه ثانية custom validations ودي هي ال common validations range

in model class

**1- Format "regular express"**

**2-Required "Not Null"**

**3- maxlength**

**4- minLength**

**5- Range "int"**

-ال #c# عامله built in validation attributes للحاجات ال common دي والAttributess دي بحطها فوق البروبيري في الكلاس بتاعي اللي في الموديل... ولازم نعمل dataAnnotation

```
using System.ComponentModel.DataAnnotations;
```

-استخدمنا قبل كدا اتربيوت스 ال [Display()] وال [DataType()]

-هنسخدم على ال Name اتربيوت [Required] ولو فتحنا requireddefinition هنلاقى انه كلاس جوا validationAttribute وبيورث من dataAnnotations

-هنسخدم برضو اتربيوت스 ال [MinLength] و [maxLength] وكل الاتربيوتس مع بعض بيعملها &&

```

// [DataType(DataType.EmailAddress)]
[Display(Name="Student Name")]
[Required]
[MaxLength(30)]
[MinLength(3)]
public string Name { get; set; }

```

-هنجي لل Address هنعمل انه required وعايزين يكون ليه صيغة معينه .. بيقى هنستخدم اتربيوت  
 لما نستخدم ال regularExp بنحط @ لو في حاجات هنستخدمها زاي / او | او  
 الحاجات اللي بتستخدمها السى شارب ولو هنعمل دوت () نعمل قبلها سلاش

```

[Required] ←
[RegularExpression(@"(Alex|Assiut)")]
public string Address { get; set; }

```

وهنعمل كدا على ال age وال

```

[Display(Name="Student Age")]
[Required]
[Range(maximum:50, minimum:20)]
public int Age { get; set; }

```

```

[Required] ← ←
[RegularExpression(@"\w+\.(jpg|png)")]
public string Image { get; set; }

```

كدا خلصنا شغل اول فايل بناع الموديل

-هنعمل ال controller: الكونديشنز اللي عملناها في الموديل لما نروح للكنترولير مش هتشتعل غير علي الاكتشنز  
 اللي بتاخذ ف البارميترز بتاعها او بجيكت من الكلاس اللي عملنا فيه ال validations لكن لو الاكتشن مثلًا بيأخذ  
 validations دا مش هيشتغل عليها ال name و age وال حاجات كلها ك primitive parameters

-برضو اغلب الميثود اللي هتعمل تشيك علي فاليديشنر بيكون نوعها post

-الكنترولير قولنا بيورث من كلاس كنترولير وكان وارث منه حاجات زاي ViewData, ViewBag, context, ...  
 بيورث برضو بروبرتي اسمها ModelState ودي بتشتعل لما يكون البارميتر بناع الميثود نوعه او بجيكت

-الModelState دا dictionary<T> هي نوع الاوبجيكت اللي اتبعت للميثود... يكون  
 keys بتو عه باسماء البروپریتیس اللي عند الاوبجيكت اللي جه للميثود... يعني مثلًا كلاس student اللي اتبعت  
 منه او بجيكت لميثود معينه في الكنترولير دا عنده 5 بروپرتي حتى لو في واحد منه مش عليها اي فاليديشنر  
 بيأخذهم كلهم يعملهم keys عنده

-بعدين هيمسك كل key ويشوف ايه الحاجات اللي تخليه valid

-بعدين المفروض ان الداتا اللي دخلها الكلاينت هتخزن مع بعض في او بجيكت جوا الميموري

-هتجي ال ModelState تبص على الوبجيكت دا وتشوف كل key قيمته في الميموري ايه وتشوف الفاليديشنز count لعدد الایروز اللي معدتش الفاليديشنز لل key دا يعني مثلا الاسم عليه 3 criteria اللي هم count of errors=0 و max و min تشفوقي القيمه اللي دخلت لو معديه ال3 شروط دول بيقى required

-تشوف كل ال keys بنفس الطريقه

-لو لقيت ان كا ال keys بزирه يعني كلهم معدين الشروط..ال ModelState عندها بروبرتي count اى ال keys بعدها ال count كلام زيره تكون true ولو واحده بس مش زيره هيقي false اسمها IsValid

```
[HttpPost]
public IActionResult SaveEdit(int id, Student newStd)
{
    // Validation logic
    if (ModelState.IsValid)
    {
        // Logic to save newStd
    }
}
```

Key	Count
Name	0
Age	0
Image	0
Address	0
Dept_id	0

-يبي بدل ما كنا في الاكشن بنعمل تشيك ب (if(name!=null&&.....) بدل كل دا هنعمل تشيك ان ModelState.IsValid

-ماينفعش يكون الميثود او الاكشن جايela اتنين اوبجيكت من اتنين موديل مختلفين مش هيعرف يعمل علي مين تشيك

-هنعدل الاتنين اكشنز ال SaveEdit اللي بيجلها الداتا لما اعمل submit من الفورم ..وال New اللي بدوس عليها علشان تفتحلي ال view اللي هعدل فيه او اضيف طالب جديد والاتنين نوعهم HttpPost

```
[HttpPost]
public IActionResult SaveEdit(int id, Student newStd)
{
    Student oldStudent = context.Students.FirstOrDefault(s => s.Id == id);
    // if (newStd.Name!=null&& newStd.Age > 10 )
    if(ModelState.IsValid==true)
    {
        //get old object
        oldStudent.Name = newStd.Name;
        oldStudent.Address = newStd.Address;
        oldStudent.Age = newStd.Age;
        oldStudent.Image = newStd.Image;
        oldStudent.Dept_Id = newStd.Dept_Id;
        context.SaveChanges();
        return RedirectToAction("Index");
    }
}
```

```

[HttpPost]
[ValidateAntiForgeryToken]//intenal calling only "FRom the same domain
public IActionResult New(Student newStd)
{
    // if (newStd.Name != null)
    if(ModelState.IsValid==true)
    {
        //save
        context.Students.Add(newStd);
        context.SaveChanges();
        return RedirectToAction("Index");
    }
}

```

-بعد ما ندخل كل البيانات في الفورم .. احنا كنا عملنا دروب داون ليست تعرض فيها اسماء ال Depts اللي هختار منها .. هيكون عنده مشكله ان حتى لو دخلت كله صح هيجبلي انه مش valid

-ممكنا حل اني اعمل viewModel فيها الحاجات ال primitive بس من غير الريفرنس بتاع ال dept او اني ادخل الكلاس بتاع student واعمل اوجبيكت من ال dept();

```

public virtual Department Department { get; set; } =new Department();
}

```

---> كدا عملنا فاليديشنر في فايل الموديل و عملنا تشيك في الكنترولير... فاضل نعرض الايرورز عن طريق view

-فيه اtribيوت تبع الهيلبرز خاص بتاج ال span اسمه asp-validation-for هو اللي هنستخدمه ياخذ اسم البروبرتي ... و او انا مش محدد مسج ايرور معينه هو هيعرض ال msg error من عنده

```

<form method="post" asp-action="New" asp-controller="Student" > @*asp-route-id="@Model*
<div>
    <label asp-for="Name"></label>
    <input asp-for="Name" class="form-control" />
    <span asp-validation-for="Name" class="text-danger"></span>
</div>
<div>
    <label asp-for="Address"></label>
    <input asp-for="Address" class="form-control" />
    <span asp-validation-for="Address" class="text-danger"></span>
</div>

```

-علشان اغير المسج بكتبها مع الاربيوتي بتاع فاليديشن اللي حطيناه علي البروبرتي في الموديل كدا

```

[Required]
[RegularExpression(@"\w+\.(jpg|png)",ErrorMessage ="Image must be jpg or png")]
public string Image { get; set; }

```

-ال div عندها asp-validation-summary لو عايز اعرض الايرورز مع بعض وبتاخد 3 قيم.. non مش هتعرض حاجه.. ModelOnly.. all بتعرض المشاكل بتاعت الموديل نفسه ولو عامله custom errors يعرضها برضو. بس مشكلتهم بيتعرض كل المشاكل مع بعض مش كل ايرور تحت input بتاعه

---> لما نعمل select علشان نخلي اول اختيار كلمه معينه مثلا select item وعايزينها يعتبرها مش من الاختيارات الاساسية بندتها Value=0

```
<div>
    <label asp-for="Dept_Id"></label>
    <select asp-items="deptItems" asp-for="Dept_Id" class="form-control">
        <option value="0" >-- Select Department --</option>
    </select>
    <span asp-validation-for="Dept_Id" class="text-danger"></span>
</div>
```

-بس كدا محتاج اعمل check ان الحاجه اللي هختارها مش اللي ليها value زورو دي او ممكن اختيارها -1 او اي حاجه مش primary key

-بس مفيش عندي اtribut احطه فوق البروبerti يخليني اقدر اعمل تشيك ان ال option قيمته مش زورو او -1 مثلا

---> علشان كدا محتاج اعمل custom validation وفي 3 طرق

-اول طريقة اني بعد ما اعمل تشيك ان ModelState.IsValid = true عمل تشيك ان Dept\_Id != 0 ولو مش بزورو يعمل الكود بتاعه عادي ..لكن لو بزورو عايزه ابعتله ايرور مسج علشان تظهر في ال view

-ال ModelState عندها بروبرتي تانية اسمها ModelState.AddModelError key هي الحاجه اللي هيعرض لها ال error اللي هو Dept\_ID ..لو key عنده هيعرضه عادي في مكانه لكن لو مش موجود عنده هيعرضه مع كل الايرورز في ال div ...هنعمل الكلام دا علي اكشن saveEdit

```
//Custom Validation dept_Id!=0
if (newStd.Dept_Id != 0)
{
    //save
    context.Students.Add(newStd);
    context.SaveChanges();
    return RedirectToAction("Index");
}
else
{
    //error message send view
    ModelState.AddModelError("Dept_Id", "Select Department");//div
}
}

ViewData["DeptList"] = context.Departments.ToList();
//return View("New", newStd);
return View(newStd);
```

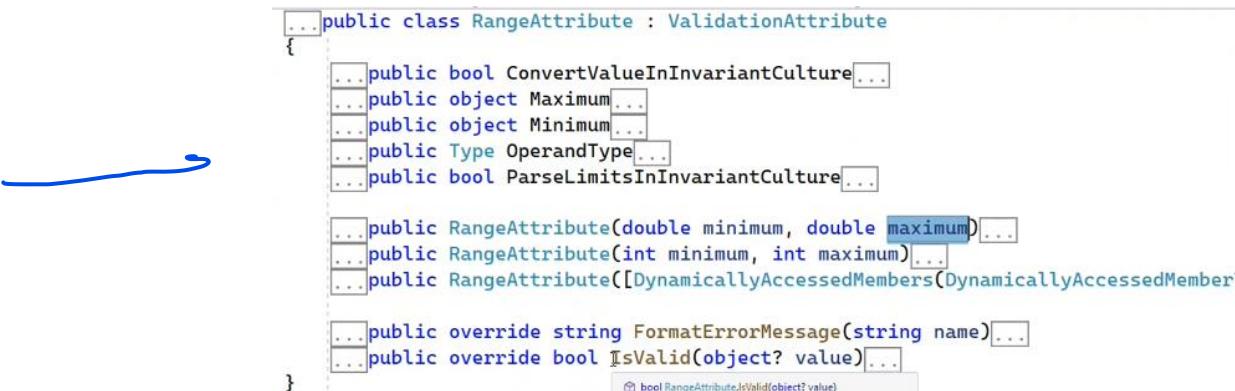
كدا الطريقة دي بستخدمها لو عايزه اعمل ال custom validation الخاص بيا ... اول لو ممكن يحصل اكسپشن واعمل try , catch وابعثهم الايرور اللي جالي وبعمل ال key فاضي ف الحاله دي

```

if (newStd.Dept_Id != 0)
{
    try
    {
        //save
        context.Students.Add(newStd);
        context.SaveChanges();
        return RedirectToAction("Index");
    }catch (Exception ex)
    {
        ModelState.AddModelError("", ex.Message);
    }
}

```

--- عايزين نعمل ال custom validation بتاعنا ينفع بيقي اtribbuit يتحط فوق البروبرتي  
 -زاي مثلا عايزه اعمل اtribbuit يتحط علي الاسم يتاكد ان الاسم دا مش موجود قبل كدا في الداتا بيز بتاعتي  
 -لو مسكت اي اtribbuit زاي Required هنلاقي انه كلاس بيوث من ValidationAttribute وبيعمل  
 -Lmited() ...باقي الatribbuits هنلاقي بتعمل override لحاجه ثاني وهي FormatErrorMessage



```

public class RangeAttribute : ValidationAttribute
{
    public bool ConvertValueInInvariantCulture...
    public object Maximum...
    public object Minimum...
    public Type OperandType...
    public bool ParseLimitsInInvariantCulture...

    public RangeAttribute(double minimum, double maximum)...
    public RangeAttribute(int minimum, int maximum)...
    public RangeAttribute([DynamicallyAccessedMembers(DynamicallyAccessedMember...
    
    public override string FormatErrorMessage(string name)...
    public override bool IsValid(object? value)...
}

```

علشان نعمل الatribbuit بتاعنا...في فولدر الموديل هنعمل كلاس ونمسيه UniqueNameAttribute ونخليه  
 يورث من DataAnnotations using ValidationAttribute

بعد كدا هنعمل override على ValidationResult ودي بترجع حاجه من نوع ValidationResult  
 .. واحده بتأخذ object والثانويه بتأخذ object

في الحالتين الـ object اسمه value بيجي فيه قيمة البروبرتي اللي بنحط عليها الatribbuit بتاعنا يعني لو حطناه  
 على Name هنجي قيمة الاسم دا في الـ object اللي اسمه value...والـ context بيجلبي فيه اوبيجيكت كله بتاع  
 الموديل اللي حطيت فيه الatribbuit بتاعي

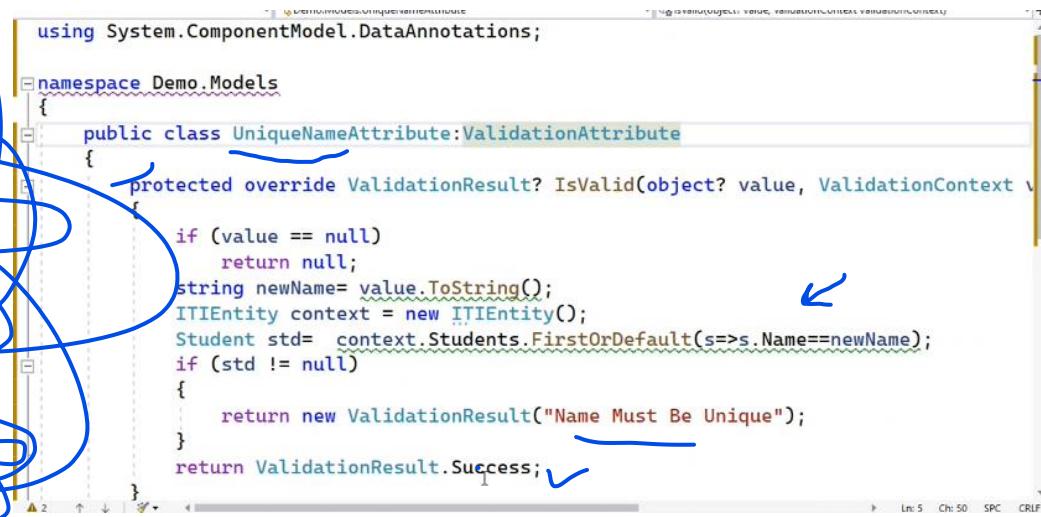
هنسخدم اللي بيأخذ اوبيجيكت وكونتيكست

اول حاجه هنتأكد ان الـ value null...بعدين هنعمل اوبيجيكت من الـ context بتاعي علشان اقدر اجيب من  
 الـ students من الداتا بيز

- هعمل سيرش في ال student اللي ف الداتا بيز بالاسم اللي جالي في ال value لو رجعلي ان في اي طالب بالاسم دا بيقي كدا محتاجه ارجع ايرور لأن انا بعمل الاتربيوت دا علشان يتأكد ان الاسم اللي دخلته unique مش موجود قبل كدا

- علشان كدا هرجع new ValidationResult وفيها المسمح ايرور بتاعتي

- لكن لو مفيش الاسم دا في الداتا بيز هرجع ValidationResult.success



```

using System.ComponentModel.DataAnnotations;

namespace Demo.Models
{
    public class UniqueNameAttribute : ValidationAttribute
    {
        protected override ValidationResult? IsValid(object? value, ValidationContext context)
        {
            if (value == null)
                return null;
            string newName = value.ToString();
            ITCIEntity context = new ITCIEntity();
            Student std = context.Students.FirstOrDefault(s => s.Name == newName);
            if (std != null)
            {
                return new ValidationResult("Name Must Be Unique");
            }
            return ValidationResult.Success;
        }
    }
}

```

-وكذا اقدر احط الاتربيوت بتاعي علي اي بروبرتي في اي كلاس وہيشوفها لأن انا عملت كلاس الاتربيوت في نفس فولدر ال models

```

[Display(Name="Student Name")]
[Required]
[MaxLength(30,ErrorMessage ="Name must be less than 30 letter")]
[MinLength(3,ErrorMessage ="Name must be greater than 2 letter")]
[UniqueName]
public string Name { get; set; }

```

-بس ال custom attribute اللي انا عملته دا بيشتعل client side مش server side علشان كدا هنلاحظ في الفورم لما ندخل اسم ويكون غلط ونعمل submit بيعمل ريفريش من فوق لأنه راح للسيرفير علشان يشوف الداتا بيز وبعدين يعرض الايرورز

-كذا لو حبيت اجيب ال student من الداتا بيز يستخد ال context بتاعي ... لكن لو عايوه اجيب اللي جالي في ال ValidatorContext يستخدم بروبرتي عندي اسمها objectInstance بس دي بترجعي الاوبجيكت اللي جاله على هيئة object وانا لازم اعمله casting

```

Student stdDb= context.Students.FirstOrDefault(s => s.Name == newName);
Student stdForm=(Student)validationContext.ObjectInstance

```

- اقدر اعمل في كلاس الاتربيوت اللي عملته اي بروبرتي براجحي مثلًا بروبتي لل msg وہياخدتها الاتربيوت كدا

```
[UniqueName(msg ="hiiiii")]
public string Name { get; set; }
```

-وممكن المسج دا هي اللي نحطها علشان تتعرض في الايررو

---> في هيلبرز اسمها @Html.validationFor

---> كدا اخدنا طريقتين لل custom validation

-اول طريقة اللي بتكون limited على مستوى الاكتشن بتاعي وبتشتغل سيرفير سايد اللي هي بنعمل تشيك جوا الاكتشن بعد ما نتأكد ان ModelState.IsValid وقولنا بنسخدمها لو في كونديشن معينه عايز اعملها اول لو هيحصل اكسشن وهمل tray, catch

-ثاني طريقة اني عملت custom attribute واستخدمه مع اي موديل عندي عادي وبرضو شغال سيرفر سايد

---> عايزين دلوقتي نشغل الفاليديتورز اللي عملناه علي الكلاينت سايد مش سيرفر سايد

-لما بنعمل input كان عندنا اtribeotid ال name مهم علشان السيرفر والاداتا تتبعته واتtriboetid ال id مهم علشان الكلاينت سايد

-عندنا باكيدج اسمها JQuery-validation وعندنا باكيدج ثاني jquery-validations دي كانت بتعمل call لميثود معينه وكذا بنكتب كود js علشان نعمل الفاليديشن

-في باكيدج ثاني jquery-unobtrusive-validation دي يستخدم اtribeotid زاي -asp بس اسمه- زاي jquery او بس علشان الصفحة تفهمهم لازم اعمل include للباكيدجز بتوع ال data-range

-في الاول كان دائمًا الفورم مش بيظهر فيها الايرورز غير لو دوست submit وبيروح للسيرفر بعدين يعرض الايرور تحت كل input لكن احنا عايزين الايرورز تكون كلاينت سايد يعني لما الكلاينت يقف على اي input ويطلع منه يظهر له الايرور تحته لحد ما يدخل القيمه الصح من غير ما يروح للسيرفر علشان يعمل كدا لازم الصفحة تكون عارفة ال unobtrusive-validations

-مايكروسوفت منزله اصلا ال 3 libraries بتوع ال jquery, jquery-validation, jquery-unobtrusive و موجودين ف فايل ال lib في ال wwwroot

-علشان نستخدمهم في ال view من تحت خالص هنعمل تاج لل script وال src بتاعها هو ال pass بتاع ال library

-لازم نحط ال 3 تحت بعض بالترتيب كدا

```
<script src="~/lib/jquery/dist/jquery.min.js"></script>
<script src="~/lib/jquery-validation/dist/jquery.validate.min.js"></script>
<script src="~/lib/jquery-validation-unobtrusive/jquery.validate.unobtrusive.min.js"></script>
```

-كدا الفورم هتشتغل كلاينت سايد واوال م ادوس علس input لو كتبت حاجه مش valid فيه وطلعت منه هيجيلي المسج تحته علي عطول

-بس لما اكتب كل حاجه مش هيعمل تشيك علي ال unique علي الاسم هيعمل تشيك لما ندوس submit لان دي مش بتتعمل غير سيرفر سايد

-يرضو ال summary اللي بتتعرض في div مش بتظهر في الاول لأنها سيرفر سايد  
 ---لو عملنا inspect على الصفحة لما تتعرض هنلاقي انا ال scripts اللي عملناهم متكررين لأنهم موجودين  
 اصلاً في ال default layout بس احنا عايزينهم يكونوا من تحت خالص  
 -في ال layout عاملين سيشن خاص بال scripts كدا وانا في ال view لو محتاجه اعمل scripts استخدم  
 السيشن دا احطهم فيه

```

        </body>
</html>

```

-نهعمل في ال view ال { Scripts } ونحط فيه ال script اللي عايزينه يتحط في السيشن دا ف  
 ال layout بس منكررش اللي موجود في ال layout ... لأن هو مستخدم ال jquery ف ال jquery

```

@section Scripts{
    @*<script src "~/lib/jquery/dist/jquery.min.js"></script>*@
    <script src "~/lib/jquery-validation/dist/jquery.validate.min.js"></script>
    <script src "~/lib/jquery-validation-unobtrusive/jquery.validate.unobtrusive.min.js"></script>
}

```

-سطرين ال script دول بدل م نكتبهم هنلاقيهم في فайл ال shared في فайл ال validationScripts ناخدهم كوبى  
 --- تالت نوع من ال validators هو انتا هنعمل ال custom validators بتعانينا بس هنخليه يستغل كلاينت  
 سايد

-نهعمل عن طريق ال ajax علشان كدا ال calling validators هيحصل لما نقف على ال input ونطلع منه  
 مش لما نعمل submit وكذا هبيان انه شغال كلاينت سايد بس هو اصلاً سيرفير سايد

-مايكروسوفت عامله اتربيوت خاص بال mvc مش موجود في ال dataAnnotation اسمه [Remote] وبنعمل  
 لل mvc علشان هو موجود فيها using

-عايزين مثلاً نعمل custom validator يتتشيك ان الحاجه اللي مدخلها بتحتوي على كلمة ITI وممكن نعملها  
 عادي بـ reg exp بس هنعملها بـ validator

-ال Remote بيأخذ اول حاجه اسم اكشن واسم الكنترولر بتعانها والمسج بتعان الايرور

```

[UniqueName(msg ="hiiiii")]
[Remote(action:"CheckName", controller:"Student", ErrorMessage ="Name Must Contai
public string Name { get; set; }

```

-يعني هنعمل في الكنترولر اللي اسمه student اللي اكشن اللي اسمها checkName ولازم تكون تأخذ بارميتر  
 نوعه string وفيه اسم البروبerti اللي حاطين عليه الاتريبوت دا... والاكتشن دا لازم يرجع json بـ true او  
 والاكتشن دا نوعها get

```

//Remote Attribute using Ajax Call
public IActionResult CheckName(string Name)
{
    if (Name.Contains("ITI"))
        return Json(true);
    return Json(false);
}

```

كدا لازم الصفحة بناعت ال view تكون بت support validation اللي نوعه كلينت سايج  
لما نعمل inspect ونشوف شكل ال input بناع ال name هيكون كدا

```


<label for="Name">Student Name</label>
<input class="form-control" type="text" data-val="true" data-val-maxlength="Name must be less than 30 letter" data-val-maxlength-max="30" data-val-minlength="Name must be greater than 2 letter" data-val-minlength-min="3" data-val-remote="Name Must Contain ITI" data-val-remote-additionalfields="*.Name" data-val-remote-url="/Student/CheckName" data-val-required="The Student Name field is required." id="Name" maxlength="30" name="Name" value> == $0
<span class="text-danger field-validation-valid" data-valmsg-for="Name" data-valmsg-replace="true"></span>


```

ال حاجات اللي هي unobtrusive وكل دا تبع ال data-val-maxlength اللي استخدمناه وهو  
بيضيفهم... وبرضو بيضيف حاجات ال Remote زاي url يحط فيه اسم الاكتشن والكونترولير  
لو عايزين نعمل تشيك ان الاسم فيه iti بس الناس اللس ال assiut بس address بناعهم

هنتع لل Remote حاجه في ال additionalField ودا بيأخذ اسم بروبرتي تانيه غير اللي مخطوط عليها  
الاتربيوت ولازم تكون بروبرتي موجوده في نفس الموديل.. هنعمل فيها address

هنعدل الاكتشن نخليها بناخدين حاجتين ال Name string وال Address

```

public IActionResult CheckName(string Name, string Address)
{
    if (Name.Contains("ITI"))
        return Json(true);
    return Json(false);
}

```

كدا لما نكتب حاجه في ال name , address هنلاقيها جت في حاجه اسمها Query مش ف  
 formData او queryString

## LEC6

:Middle Wares <-----

-في البداية خالص قولنا لازم لما نخلص الابليكشن بتاعنا بنرفعه على جهاز او سيرفير عنده IIS ودا السوفت وير اللي هيخلية يقدر يhandle requests

- السوفت وير دا كدا عباره عن كود يعني مجموعة methods وكل ميثود بتهاندل حاجه معينه وانا الويب سايت بتاعي مثلا هو هيعرض صفحة html ونوعه mvc لما ارفعه كذا هيعدى علي stack of methods علشان يشوف ايه الحاجه اللي هيستخدمها...يعني ممكن السايت بتاعي يكون تحتاج اول ميثود واخر ميثود في ال stack وكذا هيعدى علي كل الميثود اللي في النص وهو مش تحتاجها

- علشان كذا البروسيس كانت بطئه عن ال core

-لكن بعد كدا في ال net core. اللي هي عمل هاندل للريكيوبيست هو ال run time... ودور ال IIS بس انه يعتبر واجهه خارجيه هيأخذ ال request .. وهي Shawf الابليكشن دا شغال ب ايه لو بال net core. يبعثه لل run time

-يعنى كذا اللي هيحصل انا هرفع السايت بتاعي علي سيرفير مثلا شغال بال IIS... بعد كذا الكلاينت هيطلب السايت بتاعي ... ال IIS هتاخذ الدومين بتاع السايت اللي طلبه اليوزر بس مش هتعمله هاندل. ولكن احنا لما ننزل ال run time بينزل معاه proxy server اللي هو kestrel هو اللي هيساعد السيرفير ويأخذ الريكيوبيست ويهندل

-يعنى كذا ال IIS اخذ الريكيوبيست وبعده لل kestrel ... والكيسترال عنده الريكيوبيست كله وكل ما يهندل جزء من الريكيوبيست بيحطه في ال response

-لما يخلص الكيسترال هينادي ال main function .. وهي هتنطبق شوية configuration وتعمل build وبعدين تعمل run وتنتهي على الاكتشن والكونتريولير بتوعي وتبعه ال view

-الـ main بتتغير من ابليكشن للثاني علي حسب الحاجات اللي تحتاجها يعني ممكن احتاج اعمل authentication بنضيف الحاجات بتاعته

-بعد كذا ال main علشان تهندل الريكيوبيست بيكون عن طريق HttpPipeline ودا هنحط فيه مجموعة الفانشن اللي هتنفذ ... وفي البداية ال pipe line دا كان موجود في ال IIS

-كذا ال pipe line دا بحط فيه مجموعة الميثود المهمه بس اللس هيحتاجها الريكيوبيست دا وكل واحده منهم هيكون اسمها middle ware

-يعنى كذا ال pipe line عباره عن set of middleWares

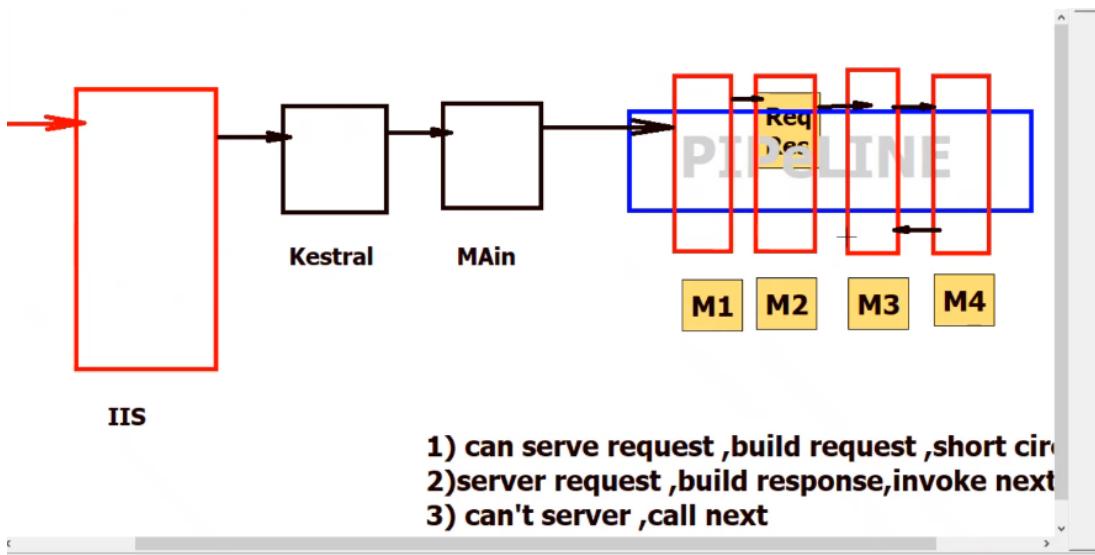
-والـ middle ware عباره عن فانشن او order middle wares لازم ليهم request delegate ... والـ middle wares بيكون ترتيب معين يعني مثلا ال authentication لازم يكون في الاول خالص

-كذا ال main معاهها ال request ,response واحنا حطينا ال middle wares بالترتيب في ال pipe line ... بعد كذا ال main هتاخذ الريكيوبيست وتعديه علي ال middle wares بالترتيب

-اول طريقة انها هتعدي على اول middle ware تشو夫 لو لقيت ال response بتاعها هيتنفذ من خلال middle main دا خالص هتنفذه وتعمل short circuit مش تحتاج تعدي علي الباقي وتعمل رينتين ترجع لل main وال main ترجع لل kestrel وهو يبعث لل IIS ومنها لليوزر

-تاني طريقة انه بعد ما يعدي علي اول middle يلاقي لسه الريسبونس تحتاج حاجات تاني علشان كدا هيعدي علي middle اللي بعد كدا ويعمل الريسبونس... لحد ما الريسبونس بيقي جاهز

-ثالث طريقة ان middle مش عنده حاجه تفید الریکویست وكل اللي هيعلملي invoke middle اللي بعده وكل middle لما يخلصو هترجع للmiddle اللي قبلها لحد ما يصلو للmain



: 2 categories Middle wares بتنقسم لـ ---

-اول كاتيجوري بتتفذ جسب ال attitude .. اللي هو هتفذ وتعمل شورت سيريك.. او هتفذ وتنه اللي بعدها او مش هتنبه اللي بعدها...ونوع بيعمل mapping عن طريق انه يشوف ال url وعلي اساسه يعمل لحاجه معينه

-الMiddle wares عباره عن شويه ميثود بتكون موجوده في الويب ابليكيشن او بجيكت... والويب ابليكيشن او بجيكت دا موجود في ال main function

-الاو بجيكت دا عنده ديليجيتس عن طريقها بنادي علي ميثود معينه مثلًا لو ناديت علي Use() دي هتعمل invoke middle وتحت execute للـ Run هتستخدم execute وبعدين تعمل Mapping url Map و terminate

### Method == > webapplication

- 1) Execute ,Invoke Next → **Use**
- 2) execute ,terminate → **Run**
- 3) Mapping (url ,execute ) → **Map**

في الـ Main بنعمل الـ middle بتوعا بعد سطر الـ builder.Build

```
public static void Main(string[] args)
{
    var builder = WebApplication.CreateBuilder(args);

    // Add services to the container.//Day8
    builder.Services.AddControllersWithViews();

    var app = builder.Build();

    // Configure the HTTP request pipeline.//Day4 (Middleware)
    if (!app.Environment.IsDevelopment())
    {
        app.UseExceptionHandler("/Home/Error");
    }
    app.UseStaticFiles();
```

-تاني custom or middle wares category لـ middle ... وهو انه يشوف انا عايز اعمل الـ middle ببني myself يعني built in middle ... ولا استخدم حاجات جاهزه عملاها مايكروسوفت اللي هي use, run , map ويرضو الحالتين دول بيستخدموا

-دا شكل الـ pipe line الجاهز اللي موجود في الـ main

```
#region Built in Middleware
// Configure the HTTP request pipeline.//Day4 (Middleware)
//if (!app.Environment.IsDevelopment())
//{
//    app.UseExceptionHandler("/Home/Error");
//}
//app.UseStaticFiles();

//app.UseRouting();

//app.UseAuthorization();

//app.MapControllerRoute(
//    name: "default",
//    pattern: "{controller=Home}/{action=Index}/{id?}");
#endregion
app.Run();
```

-لو هنعمل custom middle بتاعنا... اول حاجه نستخدم الـ use و هناقي انها بتطلب الـ HttpContext و الفانشن اللي هتعملها invoke لما تخلص وهترجع async لأنها شغال task



من الـ `HttpContext` بجنب الـ `request` ومنه استخدم اللي محتاجاه ... مثلا لو عايز الـ `url` بعمل `HttpContext.Request.path`

لو عايزه اكتب حاجه في الريسبونس يستخدم `HttpContext.Response` وبعمله `async` بعدين انادي على الـ `Next` والاتنين نعمل عليهم `await` وقبل الـ `next` `HttpContext` ونعمل `async` ... ونعمل `middle` تاني بنفس الطريقة ونكتب اي حاجه في الريسبونس

```
var app = builder.Build();

//Custom Middleware "inline Middleware"
app.Use(async (HttpContext, Next) => {
    await HttpContext.Response.WriteAsync("1)Middleware 1\n");
    await Next.Invoke();
    //
});
app.Use(async (HttpContext, Next) => {
    await HttpContext.Response.WriteAsync("2)Middleware 2\n");
    await Next.Invoke();
    //
    I
});
```

بعد كدا ننادي على الـ `Run` ودي بتاخذ الـ `HttpContext` بس ومش هتاخذ `next` ... ولو عملنا اي `middle` بعد الـ `run` مش هيتفذ لأنها هتعمل `terminate`

```
app.Run(async(HttpContext) => {
    await HttpContext.Response.WriteAsync("3)Terminate\n");
});

app.Use(async (HttpContext, Next) => {
    await HttpContext.Response.WriteAsync("4)Middleware 4\n");
    await Next.Invoke();
    //
});
```

يعني كدا لما اعمل اي ريكويست لو كتبت اي `url` عندي هيطبع 1 middleware1 وبعدين 2 middleware2 ومش هيطبع terminate4 .. ولكن الـ `middleware4` هترجع لـ 2 `middle2` بس مش هتطبع تاني هترجع من اخر سطر ومنها ترجع لـ 1 `middle1` وكذا تبقى خلصت

لو بعد سطر الـ `next.Invoke` كتبت حاجه كدا وهي الـ `terminate` راجعه هترجع عند سطر الـ `next.Invoke` وتشوف لو حاجه تحته هتفذها عادي لكن مش بتتفذ من الاول

كدا

```

    app.Use(async (httpContext, Next) => {
        await httpContext.Response.WriteAsync("1)Middleware 1\n");
        await Next.Invoke();
        await httpContext.Response.WriteAsync("1)Middleware 1--\n");
    });

    app.Use(async (httpContext, Next) => {
        await httpContext.Response.WriteAsync("2)Middleware 2\n");
        await Next.Invoke();
        await httpContext.Response.WriteAsync("2)Middleware 2----\n");
    });

    app.Run(async(httpContext) => {
        await httpContext.Response.WriteAsync("3)Terminate\n");
    });

    app.Use(async (httpContext, Next) => {
        await httpContext.Response.WriteAsync("4)Middleware 4\n");
    });

```

1)Middleware 1  
 2)Middleware 2  
 3)Terminate  
 2)Middleware 2----  
 1)Middleware 1--

---< بنعمل سيرش على built in middlewares in asp.net core علشان نعرف كل الـ built in middle-wares in asp.net core  
 -هترجع للـ main اللي موجوده ف الـ built in middle ware من الاول

```

#region Built in Middleware
// Configure the HTTP request pipeline //Day4 (Middleware)
if (!app.Environment.IsDevelopment())
{
    app.UseExceptionHandler("/Home/Error");
}
app.UseStaticFiles();

app.UseRouting();

app.UseAuthorization();

app.MapControllerRoute(
    name: "default",
    pattern: "{controller=Home}/{action=Index}/{id?}");
#endregion

```

-اول middle فيه useException هو view ودا علشان يقوله لما يحصل اكسشن يفتحنا الـ Error...  
 -والاحظ انهم حطوا middle بناع الاكسشن دا اول واحد فيه ودا علشان بعد م نخلص اخر خطوه زاي الـ run  
 -بترجع تاني من تحت لفوق علشان كدا لازم يكون من فوق علشان ممكن ف النص يحصل اكسشن وكدا هيعدى  
 عليه واحد راجعين عادي لكن لو اتحط تحت ممكن بعد م نعدى عليه وننجي نرجع يحصل حاجه ف النص وكدا  
 -مش هنقدر نوصل للایرور

-والmiddle دا بيشتغل لو احنا في مرحلة ال development تكون مش view لأن في المدخلات view لا يعرض لنا احنا الایرور على طول في المدخلات vs مش لازم بيعرض لنا view للایرور

-تاني middle هو use StaticFiles ودا علشان اقدر استخدم ملفات ال css و js والبوسترaps

-ثالث واحد ال useRouting دا اللي بيخلو البروجيك يعرف يعمل ال routing ويعرف ايه اسم الكلاس و ايه اسم الميثود علشان يشتغل صح.. وهي دلوقتي in built يعني ممكن منكتيش سطر ال routing دا بس الاحسن نكتبه علشان البروجيكتس القديمه

-رابع واحد ال useAuthorization ... في البدايه ايه الفرق بين authentication و authorization

-ال authentication يعني معايا valid account ... لكن ال authorization يعني ايه صلاحياتي في الاقوانت دا... ولما اجي اعمل authentication لل middle اعمل authentication قبل ال authorization اللي موجود

-آخر middle هو MapControllerRoute ودا بيأخذ ال url وبتعمله execute

---> وكذا خلصنا ال middlewares وعرفنا ازاي ال asp.net بتنهنل الريكيويست بدل ما تعتمد على ال IIS

: state management <-----

-هندما موضوع جديد وهو ال state management

-مع كل ريكويست السيرفير هي عمل كنترولير.. وبعد كدا يشوف الميثود اللي هي عملها call وي عمل او بجيكت من ال view اللي هيرجعه وي عمله render علشان يحوله كله ل html ويرد عليا بال html دا وبعد كدا يمسح كل الاوجبيكتس اللي عملها عنده

-طيب ايه فكرة ال state management

-انا لما اجي اطلب حاجه من السيرفير في البدايه اول مره اطلب منه هيفتح channel بيني وبين السيرفير.. وكذا كل يوزر هيجي للسيرفير هيفتحله channel خاصه بيه

-علشان السيرفير يميز كل سشن او تشنال اتعلمت بيدي لكل واحده فيهم رقم اسمه session ID(sid) مثل الرقم دا بيفضل مع اليوزر.. بيكون موجود عندي في الбраوزر وبلاقيه في ال network

-هيجي يوزر ثاني عايز يكلم السيرفير.. السيرفير هيشف لـ هو اول مره يكلمه عن طريق انه يدور هل اليوزر دا عنده cookie اسمه sid او لا ... لو مش عنده يفتحله channel وي عمله sid وهو راجع بال response يدي ال sid لليوزر دا

-ال cookies بجيها عن طريق inspect application وافتتح من ال application ومنه هلاقى ال cookies .. اول من فوق عند السيرفر في علامه ! جمب ال url بدوس عليها واحبب الكوكيز

-لما اعمل ريكويست هيكون في الكوكيز كتير هياخدتها الويب سايت من عندي ويعتها للسيرفير... وكل ما تكون الكوكيز كتيره دا هيخلو الريكيويست يكون ابطأ

-كلas الكنترولير اللي بنورث منه عنده او بجيكت من حاجه اسمها httpRequest ودي عندها بروبرتي اسمها cookie علشان يتحمل فيها الكوكيز بتاع اليوزر لما يعمل ريكويست

---> بعد كدا قولنا لما بيحصل ريكويست للسيرفير هيروح يكريت اوبيكت من الكنترولير ويشوف الميثود اللي هينادي عليها وال view اللي هي عمله ربender ولما يخلص هيمسح كل الاوبيكت اللي عملهم

-بس احيانا بتكون في معلومه مهمه جوا الاوبيكت اللي عمله دا زاي لما حد يعمل in log لازم يفضل فاكر مين الشخص دا

-يعني كدا في معلومه مهمه هحتاج احتفظ بيها من داخل الاوبيكت دا.. ودا اسمه state management يعني في حاجه محتاج احتفظ بال state بتاعتتها

---> طيب ايه الاماكن اللي اقدر اخزن فيها المعلومه مهمه دي

-اول مكان في server per channel وتناني مكان هو مع الكوكي بتاعت اليوزر نفسه يعني كدا لما اليوزر يعمل ريكويست هيتبعد معاه اتنين كوكى هم ال sid والمعلومه مهمه

-في طرق تانية كتير زاي ال query string او ال hidden field

---> هنشوف اول حاجه لما نخزن المعلومات مهمه عند السيرفير مش اليوزر

-في معلومات لازم تتخزن عند السيرفير مش اليوزر زاي البيانات مهمه في البنك بتكون متخزنده في سистем البنك مش مع كل يوزر علشان ميقدرش يعدل فيها... وبرضو دلوقتي في الويب سايتس اللي بنشتري من عليها في البداية كانت الحاجات اللي هختارها وهرجح في ال cart كانت مش بتتحط على السيرفير غير لو اكترت اني هشتريها .. لكن دلوقتي بتخزن في السيرفير علشان لو قفلت وفتحت تاني ارجع الاقوي الحاجات اللي كنت عايز اشتريها موجوده علشان كدا دلوقتي بنخزنها في سيشن او في الداتا بيز

-هنعمل مثلا كنترولير اسمه state ونعمل عنده ميثود اسمها SetSession ... ومثلا الميثود دي كان في لوجيك معين اتنفذ فيها وعايزين حاجه معينه جوا الميثود دي بعد ما تخلص تقضي القيمه دي موجوده عندي واحفظ بيها في السيشن

-عندنا ال HttpContext كان عنده حاجات زاي request و response استخدمناهم قبل كدا... وعنده برضو حاجه اسمه session ... ونستخدم من جوا ال session دي ميثود اسمها set وعندى منها 3 ميثود وكلهم بيأخذو value key ... والتلاته بيأخذو ال key لكن الفرق في ال value key

-اول واحد هى set ودى بتاخد value بال bytes ... والثانى setString بتاخد ال key ... والثالث int setInt بتاخدتها ال key

-لو كان جايلى اوبيكت بحوله ل json وبعدين اعمل setString

```
public class StateController : Controller
{
    public IActionResult SetSession(string name)
    {
        //logic
        HttpContext.Session.SetString("Name", name);
        //logic
        HttpContext.Session.SetInt32("Age", 21);
        return Content("Data Session Save Success");
    }
}
```

-كدا انا احتفظت بمعلومتين عند السيرفير هم ال name, age ومربوطين بال channel بتاعت اليوزر اللي هيسخدمهم

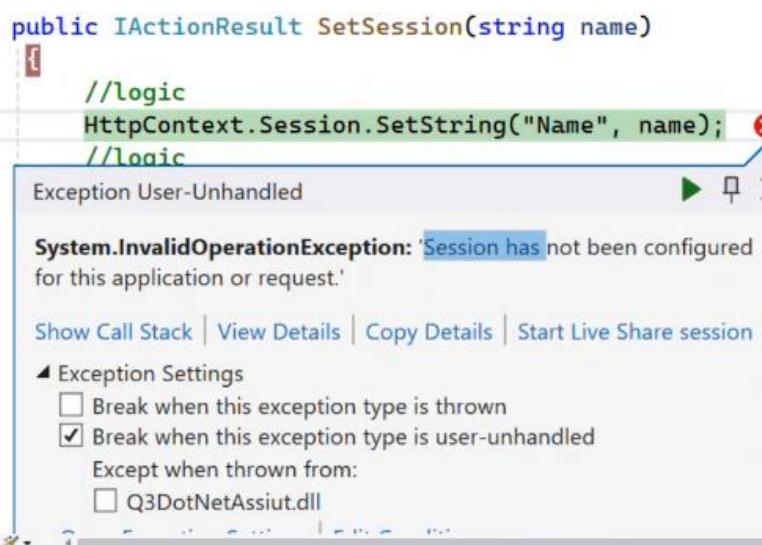
-بعد كدا لما احتاج اقرا المعلومات اللي احتفظت بيها دي مش شرط اقرابا عن طريق ميئود جوا نفس الكنترولير  
اللي اسمه...state ولكن عادي في اي مكان هعمل HttpContext.Session.get

```
public IActionResult GetSession()
{
    //logic
    string n= HttpContext.Session.GetString("Name");
    int? a = HttpContext.Session.GetInt32("Age");

    return Content($"name={n} \t age={a}");
}
```

-لاحظ ان احنا لما جينا نستقبل قيمة الـ age خلينا الـ a تكون nullable علشان ممكن يرجعلي حاجة null انا عامله  
الـ a تكون int

-دلوقي اول م اعمل run هلاقيه جابلي الاكسيشن دا



-ودا معناهاني معملتش middle ware اللي يفهمه ازاي يستخدم السيشن علشان يقدر يكتب فيها  
لما نعمل سيرش على الايرور هلاقي انه بيقول انه middle ware اللي تحتاجه هو () app.UseSession() واعمل  
سيرش علشان اعرف هستخدمه قبل ميدل routing ولا بعده

```
app.UseStaticFiles();
I
app.UseSession();
app.UseRouting();
```

-برضو بعد نحط الميدل بتاعنا هيطلع ايرور تاني كبير معناه انه مش قادر يفهم السيتينج بتاعت الميدل دا  
علشان احل المشكله دي بنعمل قبل الـ builder.Build() السطر اللي هيفهمه السيتينج بتاع السيشن  
في services.AddSession() default method بنسخدمها وهي ()

```
// Add services to the container./Day8
builder.Services.AddControllersWithViews();
builder.Services.AddSession();
var app = builder.Build();
```

- والميثود دي ليها كذا overload غير ال default دا... في session options بياخد ال overload time out للسيشن لو عايو ازود او اقله اعدل فيه عادي. بتاخذ لمدا اكسبرشن فيه ال options وترجعلي او بجيكت فيه الحاجات اللي عايزه اعدلها كدا

```
// Add services to the container./Day8
builder.Services.AddControllersWithViews();
builder.Services.AddSession(options => {
    options.IdleTimeout = TimeSpan.FromMinutes(30);
});
var app = builder.Build();
```

---> تاني طريقة في التخزين اننا نخزن المعلومه في ال cookie

- هعمل ميثود setCookie ومش محتاجين نعملها middle ware زاي السيشن لأن الابلكيشن بي support cookies عادي...بس عادي نغير السيتینج بناعتها عن طريق builder.services.AddCookiePolicy وتأخذ options وندور على الاولشن اللي محتاجين نغيره

- دلوقتي علشان اعمل setCookie اللي انشأ الكوكي دي هو السيرفير وعايز بيعتها عند اليوزر عن طريق response ويعلم append على الكوكيز اللي موجوده

- لاحظ لما هرجع الكوكي بناع ال age بترجع ال key, value الاتنين اك

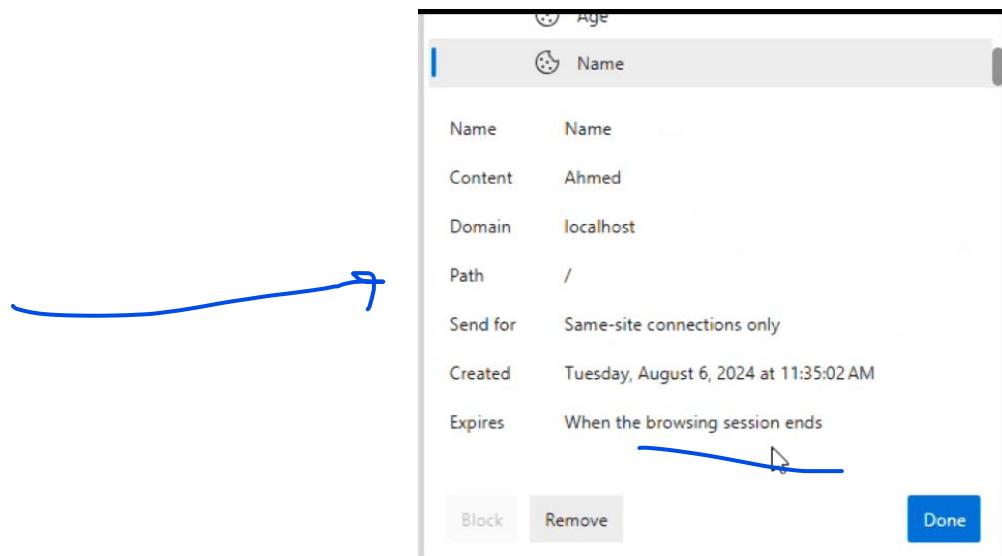
```
public IActionResult SetCookie()
{
    //logic
    HttpContext.Response.Cookies.Append("Name", "Ahmed");
    HttpContext.Response.Cookies.Append("Age", "12");
    return Content("Cookie Save");
}
```

- علشان نعمل getCookie هنسخدم ال request زاي ال getSession لأن خلاص دلوقتي الكوكي جاتلي... ولا حظ يستخدم ال key جوا انديكسر []

```
public IActionResult GetCookie()
{
    string name= HttpContext.Request.Cookies["Name"];
    string age = HttpContext.Request.Cookies["Age"];
    return Content($"name={name} \t age={age}");
}
```

-كدا لاما اكتب فب ال url `/state/setCookie` بتعطي save للاسم والعمر ولما افتح الا Cookies هلاقي name , age فيهن ال

--- كل كوكى ليها اكسبراشن وبتكون مربوطه بال session end علشان كدا النوع دا من الكوكى اسمه default ودا هو ال session cookie



-في نوع تاني اسمه persistent cookie يعني ممكن اخلي الكوكى تعيش بالأيام والسنين زاي لاما بنعمل لاي ابلكيشن ونخلي ال google يفضل فاكر الباسورد بتاعنا remember me

-والنوعين بنحددهم واحنا بنكريت الكوكى في setCookie يعني

-علشان نعملها append..واحنا بنعمل persistent options بندتها برضو زاي مثلا لو عايزه براوزر تاني يقدر يشوف الكوكى بتاعتي او غير ال expiration

-هنعمل اوبجيكت من cookieOptions علشان لما نحتاج نستخدمه في كذا مكان او نحتاج نوصله

```
public IActionResult SetCookie()
{
    CookieOptions options = new CookieOptions();
    options.Expires = DateTime.Now.AddDays(1);
    //logic
    //Session Cookie
    HttpContext.Response.Cookies.Append("Name", "Ahmed");
    //Presisitent cookie
    HttpContext.Response.Cookies.Append("Age", "12", options);
    return Content("Cookie Save");
}
```

-كدا ال name نوعه سيشن كوكى وال age ال persistent علشان اخد الاوبشنز

## LEC7

---> هنبدأ من دلوقتي نضبط الـ **structure** بناء البروجيكت بشكل صح... لأن في البدايه كنا لكل كلاس بنعمله كلاس تاني فيه الـ **business logic** وكنا بنعمل اسمه اخره **BL** وكنا بنحطهم كلهم في الـ **model** والكونترولير هيكل الموديل

-بس برضو كنا بنكتب بيزنس لوجيك جوا الكونترولير ودا بيعمل تكرار للكود...وهيصعب علينا التيسنج برضو لأن في الكونترولير بكتب **queries** في الاشن اللي تحتاجها ولو في اكشن تاني تحتاجه نفس الحاجه بكرر الـ **query** تاني

---بس دلوقتي محتاجين نطبق مبدأ الـ **separate of concerne** من مميزات الـ **mvc**...هنعمل الموديل فيه البيزنس لوجيك والكونترولير فيه الهاندل بناء الريكويست والـ **view** فيها الـ **presentation layer**

---> تاني مبدأ هنطبقه هو (**DRY**)**(Don't Repeat Yourself)** منكرش اي جزء في كذا مكان ولو في حاجة هنسخدمها كتير نعملها في فانكشن ونستخدمها في المكان اللي تحتاجينه علشان لما نحتاج نعدل يبقى في مكان واحد

---> كدا هنبدأ نضبط الكونترولير اللي عندي لأن الكونترولير كدا مش دوره خالص انه يكلم الـ **context** ولكن هو هيكل الـ **mode** يجيب اللي تحتاجه ويعطه للـ **view** يعرضه

-يبيكي كان عندنا مثلاً كلاس الـ **employee** وكلاس الـ **department** وكان الكونترولير بناء كل واحد بنعمل فيه او بجيكت من **context** ونستخدمه وعن طريقه نكلم الكلاسيز بناعتهم

-لكن دلوقت لكل كلاس هنعمله الـ **business layer** الخاصه بيها...والكونترولير هيروح يكلم الـ **BL** دي على طول والـ **BL** هي اللي تكلم الـ **context**

-الـ **BL** هيكون فيها ميثودز بتهدل حاجه معينه

---> بدل من نخلي اسمه **Business layer** هنخلي اسمها **Repository**...ولكل موديل نعمل الـ **repo** بناعته كل **repo** افل عدد ميثودز ممكن نعملها هم الـ **CRUD operations** يعني الـ **create , read, update, delete** ---> كدا هنعمل فولدر جديد للـ **Repository**

-مش بنعمل **repo** لكل الحاجات اللي جوا فولدر الـ **Models** يعني الـ **viewModels** مش بيتعملها **repo**...ولكن هنفتح كلاس الـ **context** بناعنا ونشوف الموديل اللي مستخدمنا فيها وهي اللي نعملها ريبو اللي هم

```
public class ITIContext:DbContext
{
    public DbSet<Employee> Employee { get; set; }
    public DbSet<Department> Department { get; set; }
    //
    public ITIContext():base()
    {
    }
}
```

-كدا هنعمل اتنين كلاس ريبو واحد للـ **dept**..وكدا كل ريبو هيكون جواه الميثودز اللي هتكلم الداتا بيز

-دا شكل الريبو بناع dept

```
public class DepartmentRepository
{
    //CRUD
    public void Add(Department obj)
    {
    }

    public void Update(Department obj)
    {
    }

    public void Delete(Department obj)
    {
    }

    public List<Department> GetAll()
}
```

-هعمل implementation للميثود الاساسيه ونزوود الحاجات الثانيه اللي هحتاجها

-بس في البدايه نعرف اوبيكت من الكونتيكست ونعمله new في الكونسٹراکتور

```
public class DepartmentRepository
{
    ITIContext context;
    public DepartmentRepository()
    {
        context = new ITIContext();
    }
}
```

-علشان احنا بنشتغل 8 core لما نيجي نستخدم الكونتيكست علشان نعمل add مش لازم نعمل context.Department.Add(obj) ولكن هعمل علي طول (obj) و هو من الـ obj هيعرف نوع الحاجه دي ونفس الكلام للـ update...والـ update بتعمل حاجه تاني لو انا اديتها اوبيكت مش موجود ليه id هي هتضيفه عندي

-لكن في الـ delete هعدلها نخليها تأخذ الـ ID و هعمل ميثود تاني GetByID وجوا الـ delete اندى عليها واجب الـ dept بالـ id دا وبعدين اعمله remove

```
//CRUD
public void Add(Department obj)
{
    context.Add(obj);
}

public void Update(Department obj)
{
    context.Update(obj);
}

public void Delete(int id)
{
    Department dept=GetById(id);
    context.Remove(dept);
}
```

-هنعمل getAll, GetById بالشكل دا

```
public List<Department> GetAll()
{
    return context.Department.ToList();
}
public Department GetById(int id)
{
    return context.Department.FirstOrDefault(d => d.Id == id);
}
```

Ln: 22 Ch: 10

-بس كدا في حاجه احنا معملهاش بعد ما عملنا add, update, delete المفروض كنا نعمل saveChange

-احنا هنعمل ال save في فانكشن لوحدها علشان في حالات تحتاج اعمل save بعد كذا خطوه مش اول ما اعمل add اعمل save وكل م اعمل حاجه اروح للداتا بيز واعملها save لوحدها... ولكن هنعمل save للكونتيكت كله ودا هيخلينا نروح للداتا بيز عدد مرات اقل ودا احسن لـ performance

```
public void Save()
{
    context.SaveChanges();
}
```

-وكذا كل م تحتاج اعمل save هنادي علي الميثود دي

-هنعمل نفس الريبو لل Employee

---> يبقى كدا اول م نعمل بروجيكت نعمل موديل بعدين ريبو بعدين كنترولير

-هنروح دلوقتي للكنترولير بناع كل واحد ونعدله وابدأ حاجه هنشيل منهم الكونتيكت وبدها نعمل اوبيجكت من الريبو ونحتاج احياناً نعمل اوبيجكت من كذا ريبو

```
public class EmployeeController : Controller
{
    // ITIContext context = new ITIContext();
    DepartmentRepository DepartmentRepository;
    EmployeeRepository EmployeeRepository;
    public EmployeeController()
    {
        DepartmentRepository = new DepartmentRepository();
        EmployeeRepository = new EmployeeRepository();
    }
}
```

-بعد كذا نعدل كل اند بوينت .. مثلاً اال new كنت ببعتها في ال ViewData كل ال dept علشان كذا هنادي علي getAll

```
[HttpGet]
public IActionResult New()
{
    ViewData["DeptList"] = DepartmentRepository.GetAll();
    return View("New");
}
```

بعد كدا ال saveNew كانت بتضيف emp جديد علشان كدا من ريبو ال emp هننادي على ال Add بعدين ال save

```
[HttpPost]
public IActionResult SaveNEw(Employee EmpFromREquest)
{
    if(ModelState.IsValid==true)
    {
        try
        {
            //save
            EmployeeRepository.Add(EmpFromREquest);
            EmployeeRepository.Save();
            return RedirectToAction("Index");
        }catch (Exception ex)
        {
            ModelState.AddModelError(string.Empty, ex.InnerException.Message);
        }
    }
    ViewData["DeptList"] = IDepartmentRepository.GetAll();
    return View("NEw", EmpFromREquest);
}
```

بعد كدا ال Index كانت بتعرض كل الموظفين

```
public IActionResult Index()
{
    return View("Index", EmployeeRepository.GetAll());
}
```

بعد كدا ال هنعدا ال getAll, GetById لكن ال mapping مش بنجلي فيها لأن اللي بيعمله الكنترولير مش الريبو

```
public IActionResult Edit(int id, string name)
{
    Employee EmpModel =
        EmployeeRepository.GetById(id);
    List<Department> DepartmentList = DepartmentRepository.GetAll();
    //-----Create View Mode Mapping
    EmpWithDEptListViewModel EmpViewModel=new EmpWithDEptListViewModel();
    EmpViewModel.Id = EmpModel.Id;
    EmpViewModel.Name = EmpModel.Name;
    EmpViewModel.Address = EmpModel.Address;
    EmpViewModel.ImageURL = EmpModel.ImageURL;
    EmpViewModel.JobTitle = EmpModel.JobTitle;
    EmpViewModel.Salary = EmpModel.Salary;
    EmpViewModel.DepartmentID = EmpModel.DepartmentID;

    EmpViewModel.DeptList = DepartmentList;
    return View("Edit", EmpViewModel); //Model EmpDeptListViewModel
}
```

بعد كدا نعدل الـ saveEdit نستخدم الـ getById و بعد ما نطبّط كل حاجة نستخدم update, save وفي آخرها  
بنجيب ليست من الـ dept

```
[HttpPost]
public IActionResult SaveEdit
(int id, EmpWithDEptListViewModel EmpFromREquest)//Employee EmpFromREques
{
    if (EmpFromREquest.Name != null)
    {
        Employee EmpFromDB = EmployeeRepository.GetById(id); //context.Employee
        EmpFromDB.Address=EmpFromREquest.Address;
        EmpFromDB.Name=EmpFromREquest.Name;
        EmpFromDB.Salary= EmpFromREquest.Salary;
        EmpFromDB.JobTitle= EmpFromREquest.JobTitle;
        EmpFromDB.ImageURL= EmpFromREquest.ImageURL;
        EmpFromDB.DepartmentID= EmpFromREquest.DepartmentID;
        EmpFromDB.Id= EmpFromREquest.Id;
        EmployeeRepository.Update(EmpFromDB);
        EmployeeRepository.Save();
        //context.SaveChanges();
        return RedirectToAction("Index");
    }
    return RedirectToAction("Index");
}
//All Emp Data List<Department>
EmpFromREquest.DeptList = DepartmentRepository.GetAll(); //context.Departm
return View("Edit", EmpFromREquest);
}
```

و هنعدل الـ details برضو

بعد كدا هنيجي للـ DetailsVM هنلاقي انه يحتاج الموظفين و اعمل include لـ dept معاه بس هنعملها  
و ممكن اعمل في الـ view فانكشن جديده ترجع الموظف و تعمل include برضو

```
public IActionResult DetailsVM(int id)
{
    Employee empModel = EmployeeRepository.GetById(id);
    |
    List<string> branches = new List<string>();

    branches.Add("Assiut");
    branches.Add("Alex");
    branches.Add("Cario");
    //declare viewmode
    EmpDEptColorTempMSgBrchViewModel EmpVM =
        new EmpDEptColorTempMSgBrchViewModel();
    //Mapping
    EmpVM.EmpName = empModel.Name;
    EmpVM.DeptName = empModel.Department.Name;
```

-بعد كدا نعدل كنترولير ال department

```
//-----  
DepartmentRepository DeptREpo; I  
public DepartmentController()  
{  
    DeptREpo = new DepartmentRepository();  
}  
  
public IActionResult Index()  
{  
    List<Department> departmentList = DeptREpo.GetAll();  
    //context.Department.Include(d=>d.Emps)  
    //.ToList();  
    return View("Index", departmentList); //Model List<department>  
}  
  
//POST Handler  
[HttpPost] //Fillter  
public IActionResult SAVeAdd(Department newDeptFromRequest)  
{  
    if (newDeptFromRequest.Name != null)  
    {  
        DeptREpo.Add(newDeptFromRequest);  
        DeptREpo.Save();  
        return RedirectToAction("Index");  
    }  
    return View("Add", newDeptFromRequest); //Model DEpartment  
}  
}
```

:SOLID principles <-----

 **S : Single Responsibility Principle**

**O : Open / Closed Principle**

**L : Liskov Substitution Principle**

**I : Interface Segregation Principle**

**D : Dependency Inversion Principle**

-هنتكلم دلوقتي على شوية مبادئ مهمه بتطبيقي على ال object oriented وكل مبدأ ليه اختصار لوحده

**S(single responsibility principle SRP) -1**

-الـ principle دا بيطبق مبدأ ال separate of concern يعني كل حاجه خاصه بجزء معين .. ودا احنا طبقناه عن طريق اننا عملنا كنترولير لكل حاجه لواحدها يعني الاند بوينتس بتاعت ال emp في كنترولير لواحدها وبتوع الـ dept لوحدهم ... وبرضو ربيو لكل واحد لواحده مش ربيو فيه كل حاجه... وبرضو من البداية على اسم ال mvc عاملين موديلز لواحدها وليها ربيو والكنترولز وال views كل حاجه متقسمه

## O(open/closed principle OCP)-2

-ودا يعني open for extend/closed for modification يعني نخلي الكلاسيز بتاعتي اقدر ازود عليها عادي لكن مش علشان ازود حاجه جديده اغير في حاجه موجوده من الاول ودا نحاول نطبقه قدر الامكان من البداييه خالص واحنا بنعمل الموديلز بتاعتنا

## L(liskov substitution principle LSP)-3

-ودا المبدأ اللي بيخلطي لو في فانكشن بتاخذ parent class لو انا بعتلها child من ال parent دا تشتعل عادي ودا بنائي علي مبدأ من مبادئ ال oop وهو ال polymorphism ... علشان كدا بعدين لما بنعمل شغلنا generic بنعمل ريبو تكون فيها ال crud تكون هي ال parent وبعدين باقي الريبو يورثو منها ويزودو بقى اللي محتاجينه

## I(interface segregation principle ISP)-4

-اولا احنا بنعمل انترفيس علشان نعمل generalization ويكون في contract كل الناس تمشي عليه...بنعمل مثلا انترفيس فيه اتنين ميثود والكلاسيز بتوع emp , dept بيعملو implement للانترفيس دا ..لو في كلاس منهم احتاج ميثود الثاني مش محتاجها هنفصلها في انترفيس لواحدتها والكلاس اللي محتاجها بس هو اللي بعملها implement

## D(dependency inversion principle DIP)-5

-معناه نعكس الاعتماديه ...يعني مثلا في الداتا ستراكتر بيكون في الجوريزمز لكل حاجه .. مثلا في ال sort ليها هذا الجوريزم منها ال bubbleSort ..هنعمل كلاس اسمه BubbleSort وجواه ميثود sort بتاخذ array of int وتعملها سورت بطريقة ال bubble

-بعد كدا لقينا اتنا عندنا كلاس تاني اسمه stack عنده array وميثود اسمها sortStack وجواها عايزين نستخدم ال bubbleSort علشان كدا هنعرف اوجبيكت من ال bubbleSort جوا الكلاس دا ونستخدمه

```
class Stack
{
    int[] arr;
    BubbleSort BubbleSort;
    public Stack()
    {
        BubbleSort = new BubbleSort();
    }
    void SortStack()
    {
        BubbleSort.Sort(arr);
    }
}
```

-طيب بعد كدا ظهر لنا الجوريزم جديد لل sort واحسن من ال bubble اسمه selectionSort وعملته الكلاس بتاعه برضو في نفس البروجيكت مع كلاس ال stack ...bubble...بس كدا ما زال ال stack معتمد على ال bubble ولو احتاجت اخليه يستخدم الالجوريزم الثاني هدخل اعدل فيه وكذا الكود بتاعي بقى مش closed for modification

-وكذا ال stack دا يعتبر bubbleSort low level class معتمد على high level class اللي هو ال

-مبأ ال DIP عاليزین منعملش الكلام دا منخلیش high level یعتمد على low level...والحل اتنا نعمل مثلًا انترفیس للسورت او abstract class

-الاحسن نعمل انترفیس مثلًا ISort عندہ میٹود sort والی ہیعمله یفھمه بقی ال sort ہیتعمل ازاں sort implement bubbleSort لانترفیس وال selection implement یعنی ال bubbleSort تعمل برضو ویعملو

```
interface ISort
{
    void Sort(int[] arr);
}

class BubbleSort:ISort
{
    public void Sort(int[] arr)
    {
        //arr sort using bubble sort
    }
}

class SelectionSort:ISort
{
    public void Sort(int[] arr)
    {
        //arr sort using selection sort
    }
}
```

-هنجی بقی لل high level class بتاعنا اللي هو stack هنعمل عنده اوبجیکت من isort ...وفي مكانین اعرف فيهم الاوبجیکت دا.. اول مكان ان یتبعتني الاوبجیکت في الكستراکتور بتاع الكلاس واساویه بالریفرینس اللي عملته من isort...وتانی طریقه ابعت نوع ال sort للفانکشن اللي اسمها sortStack

```
class Stack//High level class
{
    int[] arr;
    ISort SortObj; //Low Level Class
    public Stack(ISort _sortObj)
    {
        SortObj = _sortObj;// new BubbleSort();
    }
    void SortStack()
    {
        SortObj.Sort(arr);
    }
}
```

-وكذا كل stack اعمله يقدر يستخدم sort اللي هو عاليزه

```
Stack s1 = new Stack(new BubbleSort());
Stack s2= new Stack(new SelectionSort());
Stack s3 |= new Stack(new ChrisSort());
```

-علشان كدا كل ما نعمل repo هنعملها interface علشان نطبق المبدأ..ونبعت الانترفيس للكنترولير وفي  
ال runtime ابعت الربيو اللي انا عايزها

-هنعمل انترفيس للربيو بناهت dept و واحد لل emp ..هنعمل add و نختار class بس نكتب الاسم  
IEmpRepo و هي عملها مع نفسه انترفيس مش كلاس ولو لقينا انها كلاس غيرها لانترفيس  
كدا لما نبدأ اي بروجيكت نعمل الانترفيس بعدين الكلاسيز بعدين الكنترولير

```
public interface IEmployeeRepository
{
    public void Add(Employee obj);
    public void Update(Employee obj);
    public void Delete(int id);
    public List<Employee> GetAll();
    public Employee GetById(int id);

    public void Save();
}
```

-بعد ما عملنا انترفيس ال EmpRepo نفسه نخليه ي implement الانترفيس دا وكل م  
حتاج نزود حاجه نحطها في الانترفيس و نعملها implement في repo

-نعمل نفس الكلام لل IDepartmentRepository

-كدا عملنا الانترفيس والكلاسيز او الربيو بتوعهم ...هموح بعد كدا للكنتروليرز ونغير بدل ما كان بيستخدم  
او بجيكت من الربيو على طول نخلية يستخدم او بجيكت من الانترفيس ونعرف الانترفيس دا في الكونستراكتور يعني  
اي حد هيستخدم الكنترولير بيعتلي هو هيستخدم اي انترفيس

```
public class DepartmentController : Controller
{
    // ITIContext context = new ITIContext();
    IDepartmentRepository DeptREpo;
    public DepartmentController(IDepartmentRepository deptRepo)
    {
        DeptREpo = deptRepo; // new DepartmentRepository();
    }
}
```

-ونفس الكلام عند كونترولير ال emp وعادي هيأخذ اتنين انترفيس في الكونستراكتور

```
ic class EmployeeController : Controller  
  
    // ITIContext context = new ITIContext();  
    IDepartmentRepository DepartmentRepository;  
    IEmployeeRepository EmployeeRepository;  
    public EmployeeController(IDepartmentRepository deptReo, IEmployeeRepository EmpRepo)  
    {  
        DepartmentRepository = deptReo;  
        EmployeeRepository = EmpRepo;  
    }
```

---> لحد كدا طبقنا الخمس مبادئ بس ال controller factor مش هيعرف يكريت الاوبيكشن بناء الانترافيس دي

---> بعد ما طبقنا كل الكلام اللي فات هنلاقي اننا طبقنا مبدأ تاني اسمه IOC(inversion of control)

-برضو المبدأ دا علشان مفيش كلاس يعتمد على كلاس

-كنا وصلنا ان الكونترولير فاكتور مش عارف يمررت الانترفيس اللي بيأخذ الكونستراكتور بناء الكونترولير وكان بيجيلي الايرور دا

## An unhandled exception occurred while processing the request.

```
InvalidOperationException: Unable to resolve service for type  
'Q3DotNetAssiut.Repository.IDepartmentRepository' while attempting to activate  
'Q3DotNetAssiut.Controllers.EmployeeController'.  
Microsoft.Extensions.DependencyInjection.ActivatorUtilities.ThrowHelperUnableToResolveService(Type type, Type requiredBy)
```

---> علشان كدا هنبدأ نتكلم علي حاجه جديد وهي ال Dependency Injection(DI)

-احنا علشان شغالين net core. هي بت support ال DI... ولو حد شغال not core في حاجات المفروض تنزل

-ال MVC عندها حاجه اسمها IOC container ...يكون فيها object types ...يعني مثلا انا هكون طالب حاجه من DeptRepo هيروح الكونتيير دا يكريتلي حاجه نوعها

-ال MVC عندها ال service provider ودي عباره عن كوليكتشن وهي دي ioc container

-اللي هيكلم ال serviceProvider هو اللي بيكربيت اوبيكشن من الكونترولير اللي هو الكونترولير فاكتور

-يعني مثلا لما نعمل ال url /Employee/index هيروح ال controller factor للكونترولير اللي اسمه Employee علشان يعملي اوبيكشن منه هيلاقني انه مش بيستخدم ال parameter less constructor ولكن بيستخدم parameter constructor ومحتج يجيده اوبيكشن من انترفيس ال IEmpRepo , IDepDeptRepo

-ال controller factor مش عنده طريقه يكريت بيهما الاوبيكشن دا علشان كدا هيروح لل serviceProvider علشان تكريت اوبيكشن بي implement الانترفيس اللي محتجها دي

-طيب ايه الحاجات او العمليات اللي بيعملها ال serviceProvider

-اول حاجه Register دي اللي هي لو طلب انترفيس معين يكريت او بجيكت معين ..تاني حاجه لو حد احتاج حاجه من نوع السيرفس اللي عمالها ريجستر دا اسمه resolve ...بعد كدا لما يكون مش محتاج السيرفس او الاو بجيكت dispose يعمله

-كدا الكنترولير لما يحتاج حاجه بيعت للسيرفس بروفيدر يقولها تقدري تعنلي Resolve للحاجه دي...لو عرفت هتبعله الاوبجيكت ولو معرفتش هطلع ايرور زاي اللي كان ظاهرلنا اللي هو unable to resolve

-والايرور دا معناه انا اصلا معملتش Register الاول علشان يقدر يعمل Resolve

---> ازاي بقى هنعمل register

-هنعمله قبل ال builder مایعمل build...يعني هنروح لفайл ال program بعد ما اتعمل builder اوبجيكت من

-في 3 انواع سيرفيس اقدر اضيفها...اول حاجه ال framework service ودي سيرفس موجوده اصلا مش انا اللي هعملها وبرضو معمولها register وانا بس هستخدمنا علي طول ومثال عليها السيرفس اللي بتقرأ من ال appSetting اللي كنا بنحط فيها الكونيكتشن سترينج والسيرفس اللي بتقرأ منه هي ال IConfiguration ودي اقدر في اي كونستراكتور اقوله ياخد حاجه من نوع IConfiguration وهيعرف يستخدمها عادي مش زاي IDepartmentRepo

-تاني نوع هو ال built in service ودي بتكون متعرفه وموجوده بس محتاج اعملها register ومثال عليها addControllerWithView, AddControllerSession في ال Main اللي هم السيرفس اللي استخدمناهم في

```
public static void Main(string[] args)
{
    var builder = WebApplication.CreateBuilder(args);
    //Framework service :already declare ,already register
    //built in service :already declare ,need to register
    // Add services to the container //Day8
    builder.Services.AddControllersWithViews();
    builder.Services.AddSession(options => {
        options.IdleTimeout = TimeSpan.FromMinutes(30);
    });
    //

    var app = builder.Build();
}
```

-ثالث نوع هو ال custom service ودي انا اللي هعرفها واعملها register زاي ال repo بتوعنا

-علشان كدا اول حاجه همسك السيرفس بروفайдر باني اعمل builder.services بعدين اشوف هستخدم ايه من عنده..اول حاجه نستخدمها هي <> AddScoped ودي علشان اعرفه لما يحتاج انترفيس نوعه كذا يعمل او بجيكت نوعه كذا..علشان كدا بتحتاج تاخذ ال source type اللي هو الانترفيس وبتاخذ ال destination اللي هو هتعمل منه او بجيكت اللي هي الريبو ولازم تكون بت implement الانترفيس اللي بعنته في ال source type..وكدا احنا عملنا register

```
var builder = WebApplication.CreateBuilder(args);
//Framework service :already declare ,already register
//built in service :already declare ,need to register
// Add services to the container //Day8
builder.Services.AddControllersWithViews();
builder.Services.AddSession(options => {
    options.IdleTimeout = TimeSpan.FromMinutes(30);
});
//Custom Service "Register"
builder.Services.AddScoped<IDepartmentRepository, DepartmentRepository>();
builder.Services.AddScoped<IEmployeeRepository, EmployeeRepository>();

var app = builder.Build();
```

كدا عملنا register والكونترولير هيطلب الحاجه و هيحصل resolve ويشتغل صح

---> بعد ما حصل resolve والسيرفيس عملتني اوبيكت من الريبو او الحاجه اللي محتاجه واستخدمناها والview رجعي...الاوبيكت اللي اتعمل دا هيحصل فيه ايه هيفضل موجود ولو حد طلب resolve تاني هيتعمل نفس الاوبيكت ولا ايه اللي هيحصل؟

- علشان نجرب ونشوف ايه اللي بيحصل.. في الانترفيس بناتع `IDeptRepo` هنعمل بروبرتي جديدة `ID` ونوعها `Guid`... وبعدين في الـ `DeptRepo` هنيجي في الكونسستراكتور ونعمل `id` بالشكل دا باستخدام كلاس `string`

min 45 p2 lec8

```
public DepartmentRepository()
{
    context = new ITIContext();
    Id = Guid.NewGuid().ToString();
}
```

وـكـا المـفـرـوض كل ما يـتـعـمـل او يـبـحـكـت هـيـتـعـمـلـه id جـديـد

بعد كدا هنعمل كنترولير نجرب فيه اسمه serviceController و هنجي في الكونستراكتور بتاعه ونبعتله او بجيكت من IDeptRepo هنلاقي انه بيسألني لو عايز ي create and assign field بنفس الاسم اللي بعنه للكونستراكتور ... و هنلاقيه عملنا private field وجوا الكونستراكتور هيساوي الـ field دا بالاوجبيكت اللي بعنه للكونستراكتور بالشكل دا

-بعد كدا لو في الكنترولير دا في اي اكشن مثلا اسمها index محتاجه ابعتها او بجيكت من الانترفييس بتاع `IdeptRepo` ب Russo ... علشان اقدر ابعتها هحط قبلها اتريبيوت ال [FromServices] ودا علشان ميحصلهاش بليند ويجبها من السيرفس بروفايدر

```
public IActionResult Index([FromServices]IDepartmentRepository deptrr)
{
    return View();
}
```

-وكذا قدرنا نعمل inject في كونستراكتور او في اكشن معين ولسه هنعمل انجيكت في view  
لو مثلا عايزه ابعت ال id اللي عملته في الريبو لل view... هبعتها في ال ViewBag

```
public IActionResult Index()// [FromServices]IDepartmentRepository deptrr)
{
    ViewBag.Id = deptREpo.Id;
    return View("Index");
}
```

-هنكريت ال view دا وجواه نعرض ال id اللي بيجلها... علشان اعمل انجيكت بقى في ال view هعمل `@inject`  
وبعدها اسم الانترفييس... بس الاول نحط nameSpace في ال viewImport ... او بدل كدا نقدر نكتب جمب اسم  
ال nameSpace نفسه عند الكنترولير بتاعنا نكتب جمبه كلمة global

```
using Microsoft.AspNetCore.Mvc;
global using Q3DotNetAssiut.Repository;

namespace Q3DotNetAssiut.Controllers
{
    public class ServiceController : Controller
    {
        private readonly IDepartmentRepository deptREpo;

        public ServiceController(IDepartmentRepository deptREpo)
        {
            this.deptREpo = deptREpo;
        }
    }
}
```

-بس كدا هي global على مستوى الكنترولير دا بس... والاحسن من كدا نعمل من علي اسم البروجيكت add  
ونعمل class اسمه global using ونحط فيه اي name space عايزه

```
global using Q3DotNetAssiut.Repository;
```

-وكذا اقدر اعمل انجيكت تاني في ال view .. وكذا احنا عملنا مرتين انجيكت مره ف الكونستراكتور ومره في  
view وعايزين نشوف هل كل مره ال id بيختلف ولا بيكربيت اول او بجيكت وبيتعهولي كل م اعمل انجيكت

```
@inject IDepartmentRepository deptREpo  
{@  
    ViewData["Title"] = "Index";  
}
```

```
<h1>Index</h1>
```

```
<h3>Id From Controller @ViewBag.Id</h3>  
<hr />  
<h3>Id From View @deptREpo.Id</h3>
```

---> في حاجه تاني اننا نقدر نحدد ال life time بنطاع السيرفس واحنا بنعملها register في ال main وكدا هنقدر نحدد ال IOC هيكريتها امتى ويموتها امتى

-اول حاجه بنستخدمها اسمها AddSingleton .. دي اللي بتعمل انهها بتعمل ريجستر واول م حد يطلب بتعمله اوبجيكت واي حد بعده هيطلب تاني هتبعتله نفس الاوبجيكت اللي عملته... علشان كدا لما نغيرها لـ addSingleton هنا لباقي ال view بيرجع نفس id

```
//Custom Service "RegisterB  
builder.Services.AddSingleton<IDepartmentRepository, DepartmentRepository>;  
builder.Services.AddScoped<IEmployeeRepository, EmployeeRepository>();
```

ـ الثاني نوع ريجستر هو AddTransient ودا بي عمل اوبجيكت لكل انجيك

-والثالث اللي هو AddScoped دا بقى بي عمل اوبجيكت كل ما اعمل request مش كل ما اعمل انجيك يعني في نفس الريكيويسست بيدى كل الانجيكتات نفس الاوبجيكت

---> علشان كدا ال dispose وهو ان الاوبجيكت ينتهي بيتحدد حسب انا هعمل register باي نوع

---> دلوقتى فاضل نعمل انجيك للكونتيكست علشان ميكوش فى اي كلاس بيعتمد على كلاس

-هنجي عند ال deptRepo و EmpRepo نعمل انجيك للكونتيكست بتعاونا في الكونستراكتور بنطاع كل ريبو

```
public class EmployeeRepository : IEmployeeRepository  
{  
    //Operation Employee  
    ITIContext context;  
    public EmployeeRepository(ITIContext context)  
    {  
        this.context = context;  
    }
```

ـ بس كل دا لسه برضو بنستخدم ال default constructor علشان كدا لما عملنا new context() مكنش ينفع  
نبعتله اي options... بس كدا عايزين نعمل كونستراكتور ياخذ بارميترز او DbContextOptions ... وبعد كدا  
نبعثهم الـ base

```

public class ITIContext:DbContext
{
    public ITIContext():base() { }

    public ITIContext(DbContextOptions<ITIContext> options):base(options)
    {
    }
}

```

بس كدا لسه هنحتاج نعمل register لأن ال context عباره عن built in service علشان كدا هنحتاج اعملها واي register الحاجات اللي بنسخدمها علشان نعمل ببها ريجستر بتبدأ بكلمة Add زاي...واحنا هنسخدم AddDbContext ودا جينيرك لأن ممكن يكون عندي اكتر من كونتيكت connection وبعد كدا نبعتها الاوبشنز ..هتاخد delegate نديها الاوبشنز ونستخدم useSqlServer ونديله ال onConfigure اللي كان عند ال string

```

//Built in service need register
builder.Services.AddControllersWithViews();
builder.Services.AddSession(options =>
{
    options.IdleTimeout = TimeSpan.FromMinutes(30); //session abort
});
builder.Services.AddDbContext<ITIContext>(options => {
    options.UseSqlServer("Data Source=.;Initial Catalog=Inatke44_9M_Monf;Integrated Security=True;Encrypt=False");
});

```

ـ كان الاول ننادي على ال new Context هو ينادي على ال() اللي هو ال DbContext وكمان هو بيستخدم ال connection string اللي بندليها ال default constructor لل parent ودا بيستخدم OnConfigure وبيكون هارد كود بتحطه احنا جواها وكمان لو احتجنا نغيره لازم نعمل build من الاول وبرضو لما غيرنا مكانه وحطيناه واحنا بنعمل ريجستر برضو هو كذا هارد كود

ـ علشان كذا لازم ندخل على الجسون فايل بتاع appSetting وهنضيف key اسمه ConnectionString او بجيكت نكتب جواه ال connection string بتاعتني ...نديله اسم للكونكشن سترينج علشان ممكن يكون عندي كذا واحد مثلاً نخليه cs

```

{
    "Logging": {},

    "AllowedHosts": "*",
    "ConnectionStrings": {
        "cs": "Data Source=.;Initial Catalog=Inatke44_9M_Monf;Integrated Security=True;Encrypt=False"
    }
}

```

ـ دلوقتي هنرجع لل main وعايزن نخلي ال useSqlServer بدل ما كنا نكتب لها ال connectiopn string نخليها تقرأه من ال appSetting ... وعلشان نعمل كذا عن طريق ال builder ستخدم حاجه اسمها configuration ومنها نستخدم GetConnectionString ودي بتدخل جوا فايل ال appsetting وتدور على حاجه ال key بتاعها اسمه connection string وابعتها اسم الكونكشن اللي محتاجه

```

    });
    builder.Services.AddDbContext<ITIContext>(options => {
        options.UseSqlServer(builder.Configuration.GetConnectionString("qs"));
    });
}

```

-زاي ما قدرت اقرا من ال config file في ال main اقدر برضو اقرا منه في اي كونترولير لان دي سيرفس اقدر  
اعملها انجيكت واستخدمها على طول لان هي معمولها رجسٽر وجاهزه

```

public ServicesController(IDepartmentRepository DeptREpo, IConfiguration config) // inject
{
    deptREpo = DeptREpo;
    this.config = config;
}

```

-وقدر في ال view برضو اعمل @inject واستخدمها  
---> كدا اللي عملناه بالترتيب ... كلاس الكونتيكست خليناه بيأخذ ال default constructor وواحد تاني بيأخذ  
الاوبشنز

```

public class ITIContext : DbContext
{
    public ITIContext() : base() { } // call default

    // class inject ITIContext
    public ITIContext(DbContextOptions<ITIContext> options) : base(options)
    {

    }
}

```

بعد كدا جوا كل ربيو عملت انجيكت للكونتيكست بتاعي في الكونسٽراكتور بتاعها

```

public class EmployeeRepository : IEmployeeRepository
{
    // Operation Employee
    ITIContext context;
    public EmployeeRepository(ITIContext context)
    {
        this.context = context;
    }
}

```

وكل ربيو ليها انترفيس بنعمله انجيكت في الكونترولير اللي محتاجينه فيه

```

public class EmployeeController : Controller
{
    // ITIContext context = new ITIContext();
    // DIP (high Class (Controller) base on abstraction( interface)
    IEmployeeRepository EmployeeRepository;
    IDepartmentRepository DepartmentRepository;
    // design pattern : Dependency Injection (don't create but ask)
    // DI
    public EmployeeController(IEmployeeRepository empRepo, IDepartmentRepository depREpo)
    {
        EmployeeRepository = empRepo;
    }
}

```

-بعد كدا لازم نعمل register للـ context built in service ومن الـ custom services هنضيفه  
ونخليه يقرأ الـ connection من الـ appsetting

```
builder.Services.AddDbContext<ITIContext>(options => {
    options.UseSqlServer(builder.Configuration.GetConnectionString("cs"));
});

//register your custom service
builder.Services.AddScoped<IEmployeeRepository, EmployeeRepository>();
builder.Services.AddScoped<IDepartmentRepository, DepartmentRepository>();
```

---> دلوقتي في حاجه لو انا هعمل mapping يعني هاخد من الموديل احط في الـ view model اعمل الكلام دا  
في الـ viewmodel ولا الكـنـتـرـولـير

-الاحسن نسيبها في الكـنـتـرـولـير عادي لأن الـ viewmodel بتاخد موديل وترجع موديل لأنها مش هتعرف انا هنادي عليها في  
انهي كـنـتـرـولـير بالـظـبـط

-الاحسن بقى هعمل layer تاني اسمها service وكـذا الكـنـتـرـولـير يـكلـمـ السـيـرـفـسـ وهي تـكلـمـ الـviewmodelـ ومنـهاـ يـكلـمـ  
الـkontextـ

## LEC8

:Partial view <-----

-دا يعني جزء من ال view اللي عندنا بنأخذ الكود بتاعه ونحطه في جزء لوحده اسمه partial view ... ودا لسببين

-اول سبب ان يكون ال view الكود بتاعه Long Markup يعني طويل علشان كدا بعمله Break up واقسمه لاجزاء

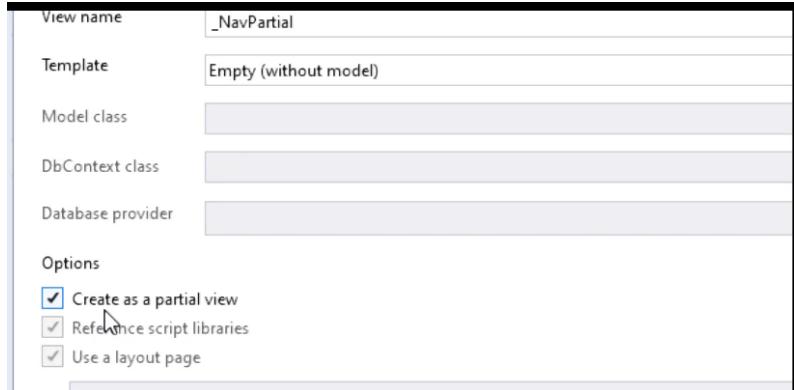
-ثاني سبب ان يكون في Repeating لجزء معين من من كود ال view بنكرره في كذا view تاني علشان كدا ال block of code اللي بيتكبر دا بنحطه في partial view ونستخدمه في الاماكن اللي تحتاجه

---> طيب ال view شبه ال view العادي ف ايه ومختلف ف ايه؟

-بيتعمل عادي في فولدر ال views والاكتشن بتاعه cshtml ولو الجزء دا اللي هيتكبر دا هنسخدمه في views موجوده مع بعض في فولدر واحد بنحطه معاهم لكن لو هنسخدمه في views في فولدرز مختلفه بنحطه في ال shared... لكن الاختلاف ان ال view عباره عن view من غير layout

---> هنمسلك مثلا ال layout اللي في ال shared هناقي ان جزء ال nav لوحده طويل هنأخذه ونحطه في partial view

-علشان كدا هنعمل جوا ال shared add و view ونختار تاني اختيار Razor view بعد كدا لما نكتب الاسم الاحسن نبدأه ب (\_) under score علشان يبقى واضح ليانا انه partial view عادي او ممكن يكون create as partial view...بعدين نختار layout



-هنأخذ جزء ال nav ونحطه في الفايل اللي عملناه دا

-بعد كدا علشان نستخدم الجزء ال partial جوا ال layout مكان ال nav اللي اخذناه في طريقتين

-اول طريقة باستخدام Html Helper ...@Html.Partial("\_navPartial")... هنعمل فانكشن هتاخذ اسم الفايل وتعمله ريندر وتحطه في نفس الفولدر معاه او في ال shared... وال Partial دى فانكشن هتاخذ اسم الفايل وتعمله ريندر وتحطه في المكان دا

-بس لما نستخدم الطريقة دي هيظهر warning وبيقولي الاحسن استخدام PartialAsync علشان تشتلل ف تاسك dead lock او نستخدم تاج ال <partial> لان بعد فتره ال Partial مش هيكون ليها استخدام وممكن تعمل للبرنام

يمكن برضو نستخدم (`@Html.RenderPartial()`) ودي هناخج اسم ال partial view بس هترجمي void...مع ان باقي ال html helpers بترجمي html content ... علشان كدا ملينفعش ننادي عليها inline لاوم تكون في `@{Html.RenderPartial( NavPartial)}` يعني هنعمل كذا block of code

-علشان كدا اللي هيحصل هيتعمل call للفانكشن دي والريسبونس وهو راجع هيكتب عليها علشان كدا هي اسرع من فانكشن ال()...بس برضو هي كمان قديمه والاحسن نستخدم النسخه async ببناعتتها ومننساش نستخدم await

-تاني طريقة والاحسن علشان احنا شغالين core هنستخدم Tag Helper و هو عباره عن `<partial name=" navPartial" >` و السطر دا نقدر نكتبه في اي view تحتاج ال

```
_Layout.cshtml" > X
<body>
  <header>
    <!--Using Html Helper @Html-->
    @* @Html.Partial("_NavPartial") *@
    @* await Html.PartialAsync("_NavPartial") *@

    @* *@
    // Html.RenderPartial("_NavPartial");
    await Html.RenderPartialAsync("_NavPartial");
  } *@

  <!--USing Tag Helper-->

    <partial name="_NavPartial" /> I
</header>
```

-كدا اخدنا طريقة اننا عملنا يار شيل لجزء من كود طويل

-تاني طريقة لو في جزء من الكود بيتركت في كذا view بس مش بيتركت فيه كلهم لأن لو فيه كلهم هحطه في الـ layout ممكن

مثلاً في الكنترولير بتاع ال emp هنعمل فانكشن اسمه Delete بترجملي view فيه موظف معين اللي عايزين نمسحه وبعدين هنعمل واحده تانى، اسمها ConfirmDelete

```
public IActionResult Delete(int id)
{
    return View("Delete", EmployeeRepository.GetById(id));
}
```

-هعمل ال View بناعها و هيعرض ال details بناع الموظف اللي جالها .. ولاحظ ان كدا احنا كرنا اللي كان عاملينه ف view ال Details بس زياده عليه حاجه واحده وهو لينك يوديني علي الاكتشن بناعت ConfirmDelete

## Delete شكل الـ

```
@model Employee
{@
    ViewData["Title"] = "Delete";
}

<h1>Delete</h1>

<h1>@Model.Name</h1>
<h2>@Model.DepartmentID</h2>
<h2>@Model.Salary</h2>

<a href="/Employee/Index">Back To List</a>
<a href="/Employee/ConfirmDelete/@Model.Id">Confirm Delete</a>
```

## Details الـ

```
Details.cshtml * x EmployeeController.cs

{@
    ViewData["Title"] = "ITI Details";
}

<h1>Details</h1>

<h1>@Model.Name</h1>
<h2>@Model.DepartmentID</h2>
<h2>@Model.Salary</h2>

<a href="/Employee/Index">Back To List</a>
<a href="/Employee/Edit/@Model.Id">Edit</a>
```

-هلاقي جزء كبير متكرر في الاتنين ولو في حاجه تاني حبينا نعرضها هنروح نعدل في الفايلين

وكذا احنا كسرنا مبدأ ال DRY .. علشان كدا هنجي في فولدر ال Emp اللي جوا ال Views هنعمل فيه partial view لان ال details , delete view الاتنين موجودين في نفس الفولدر بتاع Emp عملنا البارشيل معاهem... وهنأخذ @model strong type الجزاء اللي متكرر في الاتنين نحطه في البارشيل ... ونخلص فايل البارشيل بانتنا نعمل razor page لانه وراث من ال Employee

```
_EmpCard.cshtml * x Delete.cshtml Details.cshtml EmployeeController.cs
1 @model Emp
2
3 <h1>@Model.Name</h1>
4 <h2>@Model.DepartmentID</h2>
5 <h2>@Model.Salary</h2>
```

وهنجي مكان الكود اللي شلناه في ال partial tag details, delete ونستخدم ال

```
<h1>Details</h1>

<partial name="_EmpCard" />
```

كدا تاج البارشيل بتاخذ اسم الفايل اللي ريحاله وهي رايحة بتبعنته معها الموديل اللي بيستخدمه ال view اللي هو @model Employee

-لكن لو البارشيل مستنيه برضو حاجات عن الـ view والموديل بتاع Emp اللي معاليا في الـ Dept عامل include  
لـ Dept معاه...يبقى هبعتهاله في attribute اسمه model كدا

```
<h1>Details</h1>
<partial name="_EmpCard" model="Model.Department" />
```

---  
كدا استخدمنا البارشيل في الـ layout وفي الـ view

-تالت استخدام ان احنا نخلي الاكشن بتاعتنا ترجع partial view مش view عادي

-مثلا كان في اكشن Index كانت بترجع View الـ index علشان تعرض كل الموظفين..احنا بدل ما نستخدم فانكشن الـ View() هنسخدم PartialView() ودي اللي بتعمله مختلف انه هتعرض كود الـ Html اللي جوا view اللي هبعتها اسمه لكن مش هتعدي على الـ layout اللي في \_ViewStart...يعني احنا كنا حاطين ف الـ header, footer لما نستخدم PartialView هيعرضلي view من غيرهم...وبرضو مشهيكون مستخدم حتس البوستراب اللي كان معتمد انه هيجبيه من الـ layout

```
public IActionResult Index()
{
    return PartialView("Index", EmployeeRepository.GetAll());
}
```

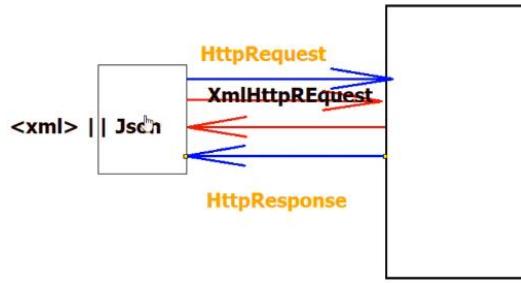
-علشان كدا نقدر نعمل اكشن ترجع لنا partial view اللي كنا حطينا فيه الجزء اللي بيترعر في details, delete ... ولازم نبعت معاه نوع الـ Model اللي هو الـ Emp من الـ ViewBag

```
//Employee/EmpCardPartial/1
public IActionResult EmpCardPartial(int id)
{
    return PartialView("_EmpCard", EmployeeRepository.GetById(id)); //Model=Null
}
```

-وادي مع ان \_EmpCard دا بارشيل بس ممكن استخدم معاه فانكشن الـ view و ساعتها هيعدي على الـ \_viewStart ويعرض معاه

-بس ايه فايدة اني عملت اند بوينت ترجعلي جزء من صفحه...لان انا اكيد ه تكون الصفحة الاساسيه متحملة عندي...وكان في الـ Ajax بنروح نجيب حاجه من السيرفر والصفحة تكون لسه مكمله معاليا عادي لأن اللي بيتعمل كان xmlHttpRequest يعني الريكيويست والريسبونس بيكونو في شكل xml علشان كدا لما بنعمل بالـ Ajax لما نجيب حاجه من السيرفر هنلاقي ان الصفحة محصلش ليها ريفريش

-في البداية بنعمل ريكويست كامل HttpRequest الصفحة بفضل مستنيه لحد ما السيرفير يرد عليا بـ HttpResponse..لكن بعد ما الصفحة تحمل لو في جزء معين منها مثلا لينك او زرار لما ادوس عليه عايزه ينفذ كود js علشان يعمل partial request هيعمله بـ XMLHttpRequest و ساعتها السيرفر هيرد عليا بـ json او XML...وبعد كدا احنا باستخدام الـ DOM, js بشوف الاوبجيكت اللي رجعلي واحط كل حاجه مكانها



- علشان كدا Ajax هي عباره عن مجموعة فيتشرز متجمعه مع بعض منهم ال XMLHttpRequest وال JS(DOM) وال CSS

- يعني كدا اللي هيحصل ان ال view بتاعي اتحمل وفي زرار لما ادوس عليه هينادي على اند بوينت بترجع وباستخدام ال dom هنأخذ الجزء دا من الصفحة ونعرضه

- احنا عملنا الاند بوينت اللي هترجع جزء من صفحة ال details, delete اللي فيها تفاصيل الموظف

- هنجي في ال view بنطاع ال index هنعمل لينك اسمه Details بینادي على الاند بوينت دي ... وهنعمل في نفس ال view دا هنعمل div من فوق كدا عايزه لما ادوس علي اللينك يعرضلي تفاصيل الموظف في المكان دا

```

<h1>Index</h1>
<a asp-action="New" asp-controller="Employee">New</a>
<div id="div1" style="border:2px solid red"></div>
<table class="table table-bordered table-hover">
    <tr>
        <th>ID</th>
        <th>Name</th>
        <th>Salary</th>
        <th></th>
    </tr>
    @foreach (var item in Model)
    {
        <tr>
            <td>@item.Id</td>
            <td>@item.Name</td>
            <td>@item.Salary</td>
            <td>
                <a href="/Employee/Edit/@item.Id">Edit</a>
            </td>
            <td>
                <a href="/Employee/EmpCardPartial/@item.Id">Details</a>
            </td>
        </tr>
    }

```

- لحد كدا اللينك دا عادي لما ادوس عليه هيعملني httpRequest ويفتحلي صفحه جديده فيها التفاصيل

- علشان كدا احنا عايوين نستخدم ال js و هنستخدم event اسمه onclick و هخليه ينادي علي فانكتشن اسمها id و نبعتها ال getEmpData()

```

<td>
    <a href="/Employee/EmpCardPartial/@item.Id" onclick="GetEmpData(@item.Id)">Details</a>
</td>

```

-كدا فاضل نعمل الفانكشن بتاعت script ... لازم من تحت نعمل تاج ال GetEmpData وجوهاها مش هنعمل jquery ولكن هنسخدم ال Jquery ... علشان كدا هنضيف في ال src اللينك بتاع ال pure js -ممكنا نعمل section للـ scripts علشان تتعرض ف الآخر وممكن لا عادي

-هفتح script تانية نعمل فيها الفانكشن...أول حاجه المفروض الليك اللي عملته فوق بعد ما ينفذ ال هيروح ينفذ ال href بس انا خلاص مش هحتاجه ينفذه لاني عمل كل حاجه ف الفانكشن ... علشان كدا أول حاجه عملها ف الفانكشن هي href يعني يمنع الديفولت اللي هو ينادي ال event.preventDefault

-تاني خطوه عايزين نعمل ajax call للاند بوينت باستخدام jquery ... هنعمل عليها سيرش نكتب ajax call مثلاً ونأخذ الكود w3school ونفتح بناء jquery using

-هلاقى انه بيعمل ajax call عن طريق \$ .ajax

-بعد كدا اول حاجه بيأخذها هي ال url نحط فيها ال باس بتاع الاند بوينت وعلشان لازم نبعت معها id هنعمل + ونبعت ال id اللي جه للفانكشن

-تاني حاجه ال success دي بتكون فيها result function(result) وال response دا اللي جايلى فيه معلومات الموظف

-ثالث حاجه هنأخذ ال result وعايزين بقى نحطها في المكان اللي عايزينه اللي هو ال div علشان كدا عن طريق السليكتور #div1 ونحط فيها ال result ... ممكن لو هشتغل جافا سكريبت عادي هعمل document.getElementById(div1).innerHTML=result

```
<script src="~/lib/jquery/dist/jquery.js" ></script>
<script>
    function GetEmpData(EmpID) {
        event.preventDefault();
        //Ajax Call Endpont using jquery

        $.ajax({
            url: "/Employee/EmpCardPartial/" + EmpID,
            success: function (result) {
                console.log(result);
                $("#div1").html(result);
            }
        });
    }
</script>
```

-بس كدا هو بيعرض جزء الصفحة دا زاي ما هو وانا مقدرش اتحكم في اي حاجه يعني لو عايز اغير في حاجه من اللي هتعرض او معارضش كل التفاصيل واعرض الاسم بس مثلاً

-الحل اني اخلي الاند بوينت ترجعلي داتا وانا باستخدام ال js اعرضها براحتي

-مثلاً لو عندي drop down list لما بدوس عليها بتعرضلي الحاجات في شكل pollet list يعني جمبهم نقطه سودا وتحت بعض بس ممكن انا عايزه اغير واخليهم يتعرضوا في شكل جدول مثلاً... علشان كدا المفروض نعمل الاند بوينت لما ندوس على الدروب داون ليست تنتادي علي الاند بوينت دي ويكون فيها بارشيل معين تنتادي عليه

---> احنا عايزين نعمل drop down list اختار منها ال department وتحتها دروب ليست تانيه بتعرض الموظفين علي حسب ال dept اللي اخترتها فوق

-لما نحتاج نعرض الموظفين مش هنجيب كل معلومات ال dept مره واحده ونختار منها الموظفين علشان هيقي كل الريسبرونس بطئ ولكن انا هكون محمله كل ال depts عندي ولما يحصل تغير واختار dept معين هبعث للسيرفر هعمل ajax call علشان بيعتني اسماء الموظفين اللي شغالين في ال dept اللي اخترتها

-بعد كدا اسماء الموظفين هتبعتلي في json file json file هاخد واعرض اللي جاي فيه في الدروب ليست بتاعت الموظفين

-كدا اللي هنعمله اول حاجه في كنترولير ال dept نعمل اند بوينت اسمها DeptEmps بترجم view وال view هياخد موديل فيه ليست من depts

```
public IActionResult DeptEmps()
{
    return View("DeptEmps", DeptREpo.GetAll()); //List<department>
}
```

-بعد كدا نعمل ال view هنحدد ان select @model List<Department> علشان نعرض الدروب داون ليست هنعمل فيها foreach تلف على ال depts وتعرضهم في ال options وال value بتعت كل deptID هي ال

-بعد كدا هنعمل select تانيه علشان نعرض فيها الموظفين بتوع القسم اللي هختاره

-علشان كدا في البدايه هنعمل الاول اند بوينت تانيه في كنترولير ال department هنأخذ ال deptID...وهرجع view json مش json

-علشان اقدر اجيب الموظفين وانا جوا كنترولير ال dept هحتاج اعمل انجيكت للريبو بتاعت ال emp

-بعد كدا جوا الريبو عايوين نعمل ميثود تأخذ ال deptID وترجملي ليست من الموظفين...هنضيفها الاول في الانترفيس بتاع الريبو كدا

```
public interface IEmployeeRepository
{
    public void Add(Employee obj);

    public void Update(Employee obj);

    public void Delete(int id);

    public List<Employee> GetAll();

    public Employee GetById(int id);

    public void Save();

    public List<Employee> GetByDeptID(int deptID);
}
```

-بعد كدا جوا الريبو نفسها implement الميثود دي و هنعمل عن طريق where و بعدين نعملToList

```
public List<Employee> GetByDEptID(int deptID)
{
    return context.Employee.Where(e=>e.DepartmentID== deptID).ToList();
}
```

-بعد كدا ارجع للاند بوينت في الكترونلر واجيب لیست الموظفين وابعنها للجسون ..ولما نكتب ال url لازم نكتب علشان لو كتبنا الرقم على طول هيعتبره id الموظف deptId=1

```
//Department/GetEmpsByDEptId?deptId=1
public IActionResult GetEmpsByDEptId(int deptId)
{
    List<Employee> EmpList= EmployeeREpo.GetByDEptID(deptId);
    return Json(EmpList);
}
```

-بعد كدا نرجع لل view تاني عايزين لما نختار dept معين يحس بالتغيير علشان كدا علي ال select بقى  
هنجعل " onchange="GetEmps()" علشان ترجملي فيها الموظفين ومش هنبعط حاجه  
علشان هتاخذ الحاجه اللي في deptId كل option value اللي هي GetEmps

```
<select id="DeptId" name="DeptID" class="form form-control" onchange="GetEmps()">
    @foreach(var deptItem in Model){
        <option value="@deptItem.Id">@deptItem.Name</option>
    }
</select>
<br />
<select id="Emps" name="Emps" class="form form-control">
</select>
```

-بعد كدا نعمل تاج script ونستخدم ال src بقى

-بعد كدا نعمل سكريبت تانية نعمل فيها الفانكشن بتاعت GetEmps...هنجعل الاول getElementById وبنعنه  
الاسم اللس اكتب في اتربيوت ال id عنده select اللي هو DeptId ...بعدين نعمل ajax call للاند بوينت اللي  
هترجع جسون فيه الموظفين...اول حاجة ياخدها الاجكس هو ال url بقى الاند بوينت هنكتبه ونعمل deptId  
علشان دا متغير...وتاني حاجة result اللي فيجاها ال success دى فيجاها الجسون ونعمل احنا جوا الفانكشن دي  
اللي عايزينه يتتفذ على الجسون اللي جاي اللي هو array

-هعمل for loop تاخذ كل موظف من ال result ...عايزين ناخذ من كل موظف يمسكه ال id وال name بتوعه  
ونحطهم جوا ال options بقى هتعرض الموظفين نحط ال id في ال value وال name في الحاجه  
اللي هتظهر لنا...علشان كدا هنمسمك الاول ال select عن طريق getElementById ...بعدها على طول نعمل  
innerHTML للسيليكت اللي مسكنها ونفضيها علشان كل مره يعرض الموظفين اللي محتاجهم بس

-بعد كدا نرجع لل for هنستخدم innerHTML ونحط ال option ونكتب ال value والقيمة علشان كدا في البداية كنا  
مع كل onChange بعد مامسمك السيليكت بنفضيها

```

<script src="~/lib/jquery/dist/jquery.min.js"></script>
<script>
    function GetEmp() {
        var deptID = document.getElementById("DeptId").value;
        var empElement = document.getElementById("Emps");
        empElement.innerHTML = "";
        console.log(deptID);
        //Ajax call json

        $.ajax({
            url: "/DEpartment/GetEmpsByDeptId?deptId=" + deptID,
            success: function (result) {
                console.log(result)
                for (let emp of result) {
                    empElement.innerHTML += "<option value='" + emp.id + "'>" + emp.name + "</option>";
                }
            }
        });
    }
</script>

```

add this here  
 empElement.innerHTML=" "; //to clear select before after  
 looping with another change  
 so when u chose another dept no contact old chosen dept  
 returned employees here loop and make to dev.

-كدا كل ما نختار dept هيعرض الموظفين بنوعه من غير ما يحصل اي ريفريش للصفحة بسبب ajax

#### :Routing <-----

-من مميزات.mvc انه بتعمل routing map يعني ال url مش بحتاج اكتب فيه الباث او الاكتشن او اسم الملف... ولكن ال mapping اللي بيحصل انه بيأخذ ال url وهو يوصلنا لصفحة ال .html ... ودا لأن مايكروسوفت بتسخدم ال restful وهو اني اكتب الاسم كذا class/method علشان كدا بنكتب الكنترولير والاندبوينت

-عادي نعدله ونكتب حاجه تاني يعني ممكن اقوله لما اكتب m1 يوديني علي اندوينت معينه في كنترولير معين

-نحتاج اتنين MiddleWare اول واحد MapRouteControler ودا بيعرف ال Routes ولما يعرف ايه route المعين اللي عايز يتنفذ هيعمله execute

-ثاني UseRoute MiddleWare دا زاي ال guide لانه بناء على ال map اللي اتعلمت في الميدل الاول  
هيختار الماب اللي هيشتغل

-لما نفتح فайл ال program هنلاقي الاتنين موجودين وكاتب ال default route اني اكتب الكنترولير والاكتشن

```

app.UseRouting(); //Guid

app.UseAuthentication();
app.UseAuthorization();

//DEcalre & execute
app.MapControllerRoute(
    name: "default",
    pattern: "{controller=Home}/{action=Index}/{id?}");
//app.Map("/djhdsjh", (app) => { });
#endregion
app.Run();

```

-هنعمل مثلا كنترولير اسمه Route وجواه ميثير اسمها default route هيكون  
بس انا عايزه اعمل ال custom route بتاعي يعني لو كتبت m1 بوديني على نفس

placeholder anything inside it will remove and substitute by the new input min 33 p2  
inside{} is variable u can make ur own but we have reserved 2 action and controller as variables

-ايه مكونات ال url ... اول حاجه ال / دي اسمها delimiter .. والكلامه اللي بين انتين delimiter اسمها  
لو كتبنا حاجه بين {} ه تكون placeholder / دا pattern علشان كدا  
لما اعمل key يحطها في ال بتاع ال name ahmed /asd/ahmed هيأخذ

-هرجع تاني علشان نعمل ال route بالاسم اللي عايزينه هنعمله في فайл ال program قبل ال middle naming  
لان ال default MapControllerRoute لاوم يكون اخر حاجه...هنجي قبله ونعمل ال pattern ... confination route  
ودا معناه انى هعمل definr لل route وهياخذ name و default هيروح له

```
//DEcalre & execute
//naming confinatoin Route (Defain rout with name ,pattern ,default )
nameUnique
app.MapControllerRoute("Route1", "R1",
    new { controller = "Route", action = "Method1" }
);
app.MapControllerRoute("Route2", "R2",
    new { controller = "Route", action = "Method2" }
);

app.MapControllerRoute(
    name: "default",
    pattern: "{controller=Home}/{action=Index}/{id?}");
#endregion
app.Run();
```

-طيب لو الاشن اللي هيروح له بيأخذ حاجات يعني عايز مش نكتب R1 ونسكت عايزين نبعط معاها حاجه

-علشان كدا الحاجه دي ه تكون متغيره و هنعملها في key placeholder ب معين وكذا لما اكتب قيمة معينه  
هتجلي في ال query string مش في ال route values

```
pattern
app.MapControllerRoute("Route1", "R1/{name}/{age}",
    new { controller = "Route", action = "Method1" }
);
```

-بس كدا ال key بتاع age حتى لو بعتله فيه حروف مش ارقام هياخذها يحطها في ال route في ال  
values لما ال model binder يجي يعمل بайнده مش هيعرف

-يعني كدا الصح ان من الاول ماكنش يحصل serve لما اكتب url فيه حاجه غلط.. علشان كدا هحتاج نعمل  
على ال route parameter constraint يمكن عمل كذا كونسرين كدا  
يعني احدد كمان القيم اللي هتدخل وممكن استخدم regular expression عن طريق regex()

```
app.MapControllerRoute("Route1", "R1/{name}/{age:int:max(40)}",
    new { controller = "Route", action = "Method1" }
);
```

-كدا لو جربت اكتب ف ال age حروف او حاجه اكتر من 40 هيرجعلي notfound  
 -لو عايزين نعمل optional segment يعني ممكن نبعتها مع ال url وممكن لا... اول حاجه لازم تتحط اخر  
 وحط؟ جمب اسم ال segment دا

```
//Optional segment
app.MapControllerRoute("Route1", "R1/{name}/{age:int}/{color?}",
    new { controller = "Route", action = "Method1" })
);
```

-طيب كدا احنا لكل ميثود بنعمل route لوحدها... بس احنا عايزين نعمل لكل كنترولير route واحد نثبت فيه  
 ال name وال controller ونغير في ال action

-في action اسمه received placeholder اللي هيجي فيه هيكون اسم الميثود اللي هرولها وممكن اديله  
 قيمة default علشان لو مكتبتش حاجه يروح على ميثود معينه علي طول

```
app.MapControllerRoute("Route1", "R/{action=Method1}",
    new { controller = "Route" })
);
```

-كدا برضو انا بعمل route لكل كنترولير... لو عايزه اعمل واحد لكله عنده received placeholder تاني اسمه  
 وممكن اديله default controller برضو

```
app.MapControllerRoute("Route1", "{controller=Route}/{action=Method1}");
```

```
app.MapControllerRoute(
    name: "default",
    pattern: "{controller=Home}/{action=Index}/{id?}");
#endregion
ann.Run();
```

-اللي عملناه دا اول طريقة وهي ال naming combination route... في طريقة تانية وهي ال api  
 ودي استخدامها اكتر في

-ودي بنجي فوق الميثود ونعمل اتربيوت كدا [Route] وتأخذ pattern فيه حاجه ليترال ه تكون سيمبليت  
 او placeholder... بس لما نستخدم الطريقة دي مش هنقدر ننادي الاند بوينت باي طريقة تانية من اللي فاتو يعني  
 لو حاولت اكتب اسم الكنترولير والميثود مش هقدر يصلهم

call action by



```
//Default route
//Route/METHOD1?name=ahmed
//R1/ahmed/12
[Route("M1/{age:int}/{name?}"), Name = "R1"] only way to call this method1
public IActionResult Method1() //string name, int age
{
    return Content("M1");
}
```

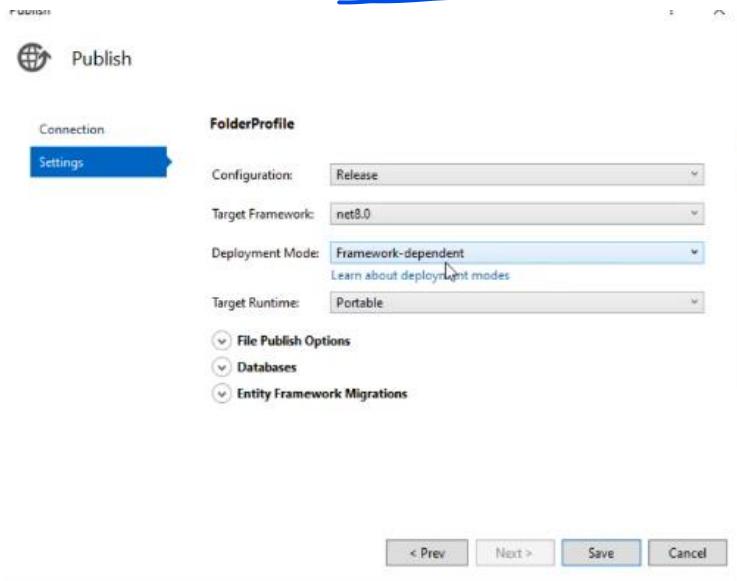
----> اخر حاجه عايز نعرف ازاي نعمل Deployment

- علشان لما نرفع البروجيكت على السيرفير مش هنرفع البروجيكت كله ولكن بنضغطه

- بندوس رأيت كلك على اسم البروجيكت ونختار publish بعدين نختار hosting folder بعدين لما اشتري Hosting برفع الفولدر دا عليه

- بعد ما اخترنا next بندوس folder ونختار الامكان اللي هنحط فيه الفولدر دا اللي هيتضغط فيه البروجيكت وبعدين ندوس finish

- بعد كدا في حاجه اسمها publish profile... هندوس show all setting هيتفتح الصفحة ددي



- اول حاجه ال deployment mode لو عملته framework-dependent دا معناه ان لازم الجهاز اللي هعمل عليه الهوست يكون منزل ال runtime... ثاني اختيار اني اخليه self-contained يعني لما اعمل deployment هيحولي البروجيكت ل DLL و هينزل معاه الباكيدجز وال runtime اللي هحتاجه عن طريق اني اختار target runtime من operating system ||

- بعد كدا ندوس publish وكل حاجه هتطلع في الفولدر اللي اخترتة واللي هرفعه هو فайл ال DLL