



UPPSALA
UNIVERSITET

. NET-programmering

Model forts

Flera modeller i en vy

- Fyll på i Interface och implementering
- Många dataklasser (poco)
 - Skapa en mapp i Models som håller reda på dem
 - OBS! Namespace ändras för poco-klasserna
 - Ta bort ordet poco i namespace



UPPSALA
UNIVERSITET

Exempel

Lärarklass (i POCO-mapp)

```
public class Teacher
{
    public int TeacherID { get; set; }
    public string Code { get; set; }
    public string Name { get; set; }
}
```

ISchoolRepository

```
IQueryable<Teacher> Teachers { get; }
```

FakeSchoolRepository

```
public IQueryable<Teacher> teachers => new List<Teacher>
{
    new Teacher { TeacherID = 1, Code = "TT", Name = "Tomas Trivial" },
    new Teacher { TeacherID = 2, Code = "JC", Name = "Jonas Cluedo" },
    new Teacher { TeacherID = 3, Code = "RM", Name = "Rakel Monopol" }
}.AsQueryable<Teacher>();
```



UPPSALA
UNIVERSITET

Exempel forts

@model IEnumerable<Student>



@model ISchoolRepository

Index.cshtml

```
@foreach (Teacher teacher in Model.Teachers)
{
    <div>
        <p>@teacher.TeacherID</p>
        <p>@teacher.Code</p>
        <p>@teacher.Name</p>
    </div>
}
```

Skicka med id till nästa vy

```
app.UseMvcWithDefaultRoute();
```

```
routes.MapRoute(  
    name: "default",  
    template:  
    "{controller=Home}/{action=Index}/{id?}");
```

ex.: /Products/Details/17



Tag Helpers

- Tag Helper
 - asp-controller
 - asp-action
 - asp-route-id

```
<a asp-controller="Product" asp-action="Details"  
    asp-route-id="17">  
    Visa produktdetaljer  
</a>
```



UPPSALA
UNIVERSITET

Controller/Action/Id

```
public class ProductController : Controller
{
    public ActionResult Details(int id)
    {
    }
}
```

OBS!
Samma namn som i route



UPPSALA
UNIVERSITET

Exempel - View

Index.cshtml

```
@foreach (Teacher teacher in Model.teachers)
{
    <div>
        <p><a asp-controller="Home" asp-action="ShowTeacher"
            asp-route-id="@teacher.TeacherID">@teacher.TeacherID</a>
        </p>
        <p>@teacher.Code</p>
        <p>@teacher.Name</p>
    </div>
}
```




UPPSALA
UNIVERSITET

Exempel - Action

```
public ActionResult ShowTeacher(int id)
{
    //LINQ uttryck för att hämta ut en lärare
    var teacherDetail = from td in repository.Teachers
                        where td.TeacherID == id
                        select td;

    return View(teacherDetail);
}
```



IEnumerable<T>

- Flesta klasser och interfaces inom Collections ärver av denna
- Enkel hämtning av data och skickar vidare
- Hämtar hela tabellen och går sedan igenom den
- Fungerar bäst med interna källor

IQueryable<T>

- Ärver av IEnumerable<T>
- Skapar ett SQL anrop
- Fungerar bäst med externa källor



UPPSALA
UNIVERSITET

View - ShowTeacher

```
@model IEnumerable<Teacher>

@foreach (Teacher teacher in Model)
{
    <div>
        <p>@teacher.TeacherId</p>
        <p>@teacher.Name</p>
        <p>@teacher.Code</p>
    </div>
}
```



UPPSALA
UNIVERSITET

Exempel - Action

```
public ActionResult ShowTeacher(int id)
{
    var teacherDetail = repository.Teachers.Where
        (td => td.TeacherID == id).First();

    return View(teacherDetail);
}
```



UPPSALA
UNIVERSITET

View - ShowTeacher

```
@model Teacher
```

```
<div>  
    <p>@Model.TeacherID</p>  
    <p>@Model.Name</p>  
    <p>@Model.Code</p>  
</div>
```