

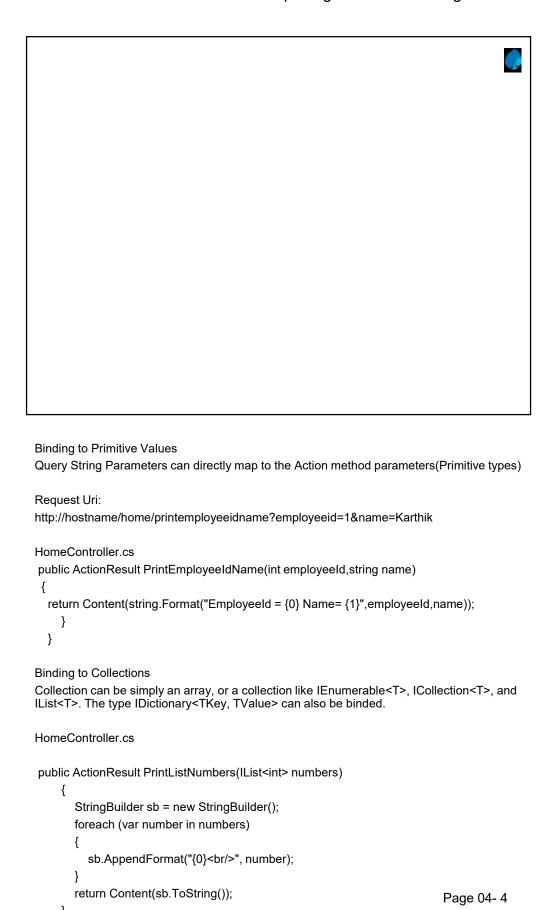
# Lesson Objectives Model Binding Data Annotations

4.1

# **Model Binding**



- Model Binding is mapping the HTTP request data directly to Action method parameters and .NET objects (a Model).
- >ASP.NET MVC framework provides a very powerful model binder that can bind most of the data types like
  - Primitive types
  - · Array & Collection
  - Objects



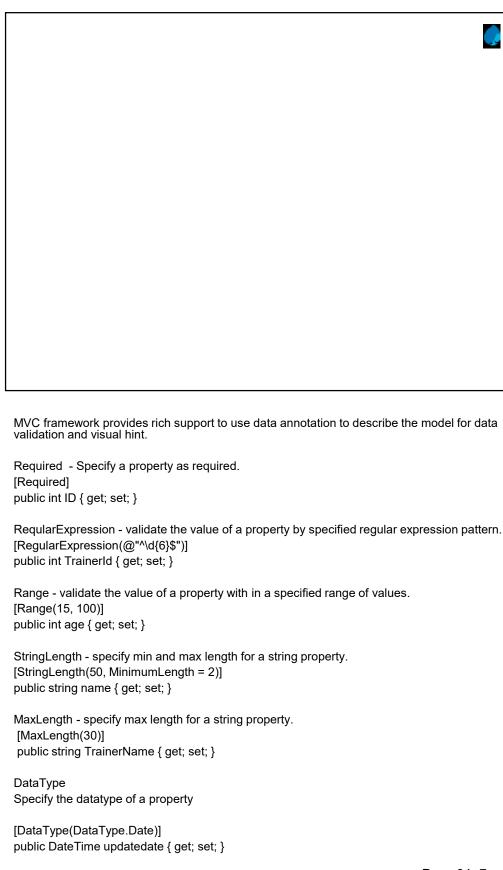
```
Index.cshtml
<form method="get" action="@Url.Action("PrintListNumbers")">
  <input type="text" name="numbers[0]" value="1" />
  <input type="text" name="numbers[1]" value="2" />
  <input type="text" name="numbers[2]" value="3" />
  <input type="submit" value="Submit" />
</form>
Binding to Object
Trainer.cs(Model)
public class Trainer
    public int TrainerId { get; set; }
    public string TrainerName { get; set; }
    public Technology Subject { get; set; }
                                                }
Technology.cs(Model)
  public class Technology {
     public int Technologyld { get; set; }
     public string TechnologyName { get; set; }
                                                   }
HomeController.cs
public ActionResult PrintTrainer(Trainer trainer)
       return Content(string.Format("TrainerId = {0} Name= {1} Technology = {2}",
trainer.TrainerId, trainer.TrainerName, trainer.Subject.TechnologyName));
We can add binding attribute to instruct model binder when to bind property and
when to exclude property
public ActionResult Create([Bind(Exclude="StudentId",
Include="StudentName")]Student student)
{ ... }
```

4.2

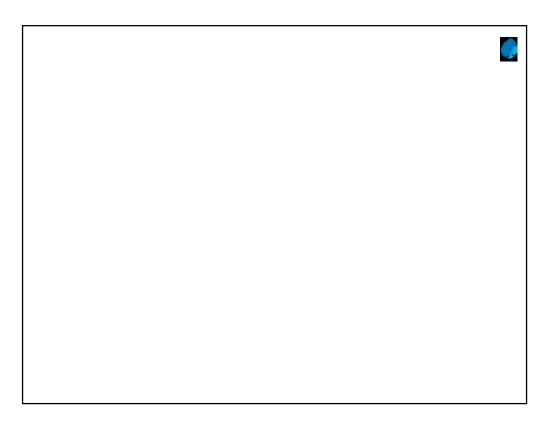
## **Data Annotations**



- Data Validation can be easily applied to models by using Data Annotation attribute.
- ➤ Data Annotation attributes classes are present in System. Component Model. Data Annotations
- Data Annotations help to define the rules to the model classes or properties for data validation and displaying suitable messages to end users.
- Client side validation can be enabled by referring jquery.validate.js and jquery.unobtrusive-ajax.js



[DataType(DataType.PhoneNumber)] public string phone { get; set; }



### Other Datatypes

DateTime -Represents a date and time of day.

Date -Represents a date value.
Time -Represents a time value.

PhoneNumber -Represents a phone number value.

Currency -Represents a currency value.

-Represents an e-mail address.

Password -Represent a password value.

Url -Represents a URL value.

CreditCard -Represents a credit card number.

PostalCode -Represents a postal code.

### DisplayName

Specify the display name for a property. [DisplayName("Employee Name")] public String EmpName { get; set; }

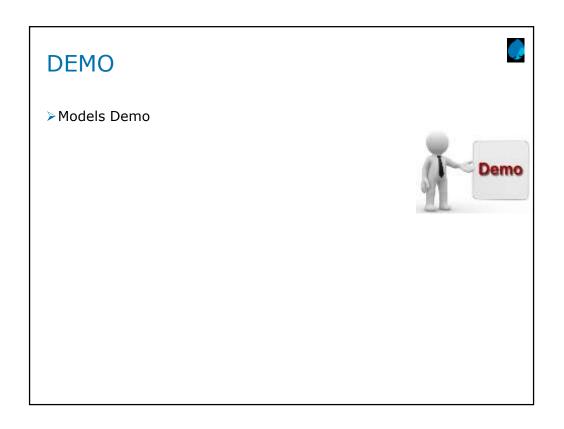
### DisplayFormat

Specify the display format for a property like different format for Date property. [DisplayFormat(DataFormatString = "{0:d}")] public DateTime JoiningDate { get; set; }

### ScaffoldColumn

Specify fields for hiding from editor forms.

[ScaffoldColumn(false)]
public int Empld { get; set; }



# Summary



- We can bind most of the data types like primitive types, Array & Collection and Objects
- Data Annotations allows us to describe the rules which we need to apply to the model properties
- ➤ Using System. Component Model. Data Annotations attributes we can perform client and server validation without any additional coding
- >We need to set Client Validation Enabled and Unobtrusive Java Script Enabled to true in the web. config file to perform client validation.

