In this video, we will discuss using the following attributes, with examples.   
**1.** Display  
**2.** DisplayName  
**3.** DisplayFormat  
**4.** ScaffoldColumn

**We will be using table tblEmployee for this demo.**  
Create table tblEmployee  
(  
 Id int primary key identity,  
 FullName nvarchar(100),  
 Gender nvarchar(10),  
 Age int,  
 HireDate DateTime,  
 EmailAddress nvarchar(100),  
 Salary int,  
 PersonalWebSite nvarchar(100)  
)

Insert into tblEmployee values  
('John Smith', 'Male', 35, '2007-01-02 17:53:46.833', 'JohnSmith@pragimtech.com', 45000, 'http://www.pragimtech.com')  
Insert into tblEmployee values  
('Mary Jane', NULL, 30, '2009-05-02 19:43:25.965', 'MaryJane@pragimtech.com', 35000, 'http://www.pragimtech.com')  
  
Generate ADO.NET entity data model for table **tblEmployee**. Change the entity name from **tblEmployee** to **Employee**. Save and build the project.  
  
Right click on the **"Controllers"** folder and add **"HomeController"**. Include the following **"USING"** statement.  
using MVCDemo.Models;  
  
**Copy and paste the following code.**  
public class HomeController : Controller  
{  
    public ActionResult Details(int id)  
    {  
        SampleDBContext db = new SampleDBContext();  
        Employee employee = db.Employees.Single(x => x.Id == id);  
        return View(employee);  
    }  
}  
  
Right click on the **"Details"** action method, and add **"Details"** view. Make sure you are creating a strongly typed view against **"Employee"** class. Select **"Details"** as the **"Scaffold Template"**. Run the application and notice that, the output is not that pretty.  
  
We can control the display of data in a view using display attributes that are found in **System.ComponentModel.DataAnnotations** namespace. It is not a good idea, to add display attributes to the properties of auto-generated "Employee" class, as our changes will be lost, if the class is auto-generated again.

So, let's create another **partial "Employee" class**, and decorate that class with the display attributes.  Right click on the "Models" folder and add Employee.cs class file. Copy and paste the following code.  
namespace MVCDemo.Models  
{  
    [MetadataType(typeof(EmployeeMetaData))]  
    public partial class Employee  
    {  
    }  
  
    public class EmployeeMetaData  
    {  
        //If you want "FullName" to be displayed as "Full Name",   
        //use DisplayAttribute or DisplayName attribute.  
        //DisplayName attribute is in System.ComponentModel namespace.  
        //[DisplayAttribute(Name="Full Name")]  
        //[Display(Name = "Full Name")]  
        [DisplayName("Full Name")]  
        public string FullName { get; set; }  
  
        //To get only the date part in a datetime data type  
        //[DisplayFormat(DataFormatString = "{0:d}")]  
        //[DisplayFormatAttribute(DataFormatString="{0:d}")]  
  
        //To get time in 24 hour notation  
        //[DisplayFormat(DataFormatString = "{0:dd/MM/yyyy HH:mm:ss}")]  
  
        //To get time in 12 hour notation with AM PM  
        [DisplayFormat(DataFormatString = "{0:dd/MM/yyyy hh:mm:ss tt}")]  
        public DateTime? HireDate { get; set; }  
  
        // If gender is NULL, "Gender not specified" text will be displayed.  
        [DisplayFormat(NullDisplayText = "Gender not specified")]  
        public string Gender { get; set; }  
  
        //If you don't want to display a column use ScaffoldColumn attribute.  
        //This only works when you use @Html.DisplayForModel() helper  
        [ScaffoldColumn(false)]  
        public int? Salary { get; set; }  
    }  
}  
  
**Make sure to include the following using statements:**  
using System.ComponentModel.DataAnnotations;  
using System.ComponentModel;  
  
**We will discuss the following attributes in our next video session.**  
DataTypeAttribute,   
DisplayColumnAttribute