**Validation Rules**

Required

Range

MinLength

StringLength

RegularExpression

DataType

Compare

**Custom Validator**

Public int Age {get;set;}

Steps for making Custom Validator :

1. Make a class, inherit it from ValidationAttribute class

public class CheckDateRangeAttribute :

ValidationAttribute

1. Add code in IsValid method to apply your rule
2. return new ValidationResult

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.Linq;

using System.Web;

namespace WebApplication15.Models

{

public class ReleaseYear : ValidationAttribute

{

protected override ValidationResult IsValid(object value, ValidationContext validationContext)

{

if (value != null)

{

int? ReleaseYear = (int)value;

if (ReleaseYear != null)

{

if (ReleaseYear < 30)

{

return ValidationResult.Success;

}

}

}

return new ValidationResult(ErrorMessage ?? "Release Year should be less than 30");

}

}

public class ReleaseDate :

ValidationAttribute

{

protected override ValidationResult

IsValid(object value, ValidationContext validationContext)

{

if (value != null)

{

DateTime dt = (DateTime)value;

if (dt != null)

{

if (dt >= DateTime.UtcNow)

{

return ValidationResult.Success;

}

}

}

return new ValidationResult(ErrorMessage ?? "Make sure your Release date is >= than today");

}

}

}

Model Class

using System;

using System.Collections.Generic;

using System.ComponentModel.DataAnnotations;

using System.ComponentModel.DataAnnotations.Schema;

using System.Data.Entity;

using System.Linq;

using System.Web;

namespace WebApplication15.Models

{

[Table(name:"tblMovie")]

public class Movie

{

public int Id { get; set; }

[Required(ErrorMessage ="Genre is required")]

[StringLength(10, ErrorMessage ="Maximum 10 characters can be added")]

[MinLength(6, ErrorMessage ="Min 6 characaters needed")]

public string Genre { get; set; }

[Required(ErrorMessage ="Name is must")]

[RegularExpression("[A-Z a-z]+", ErrorMessage ="Only alphabets allowed")]

public string Name { get; set; }

[Required(ErrorMessage ="Director Name is must")]

[RegularExpression("[A-Z a-z]+", ErrorMessage = "Only alphabets allowed")]

public string Director { get; set; }

[Required(ErrorMessage ="Password is must")]

[DataType(DataType.Password)]

public string Password { get; set; }

[Required(ErrorMessage ="Passsord is must")]

[Compare("Password", ErrorMessage ="Passwords do not match")]

public string ReTypePassword { get; set; }

[Required(ErrorMessage = "Release Year is Required")]

[ReleaseYear]

public int? ReleaseYear { get; set; }

[Required(ErrorMessage ="Release Date is Required")]

[ReleaseDate]

public DateTime? ReleaseDate { get; set; }

}

public class Movie3DbContext : DbContext

{

public DbSet<Movie> Movies {get; set; }

}

}