In this video, we will discuss **ValidateInput**attribute. This attribute is used to enable or disable request validation. By default, request validation is enabled in asp.net mvc. Let's understand this with an example.   
  
   
  
**Step 1:** Create an asp.net mvc4 application using Empty template.  
  
**Step 2:** Add a HomeController. Copy and paste the following code.  
public class HomeController : Controller  
{  
    public ActionResult Index()  
    {  
        return View();  
    }  
  
    [HttpPost]  
    public string Index(string comments)  
    {  
        return "Your Comments: " + comments;  
    }  
}  
  
   
  
**Step 3:** Add Index.cshtml view. Copy and paste the following code.  
<div style="font-family:Arial">  
@using (Html.BeginForm())  
{  
    <b>Comments:</b>   
    <br />  
    @Html.TextArea("comments")  
    <br />  
    <br />  
    <input type="submit" value="Submit" />  
}  
</div>  
  
**Step 4:**Navigate to **/Home/Index**. Type the following text in the **"Comments"** textbox and click **"Submit"**.  
<h1>Hello</h1>  
  
Notice that, you get an error - A potentially dangerous Request.Form value was detected from the client (comments="<h1>Hello</h1>"). This is because, by default, request validation is turned on in asp.net mvc and does not allow you to submit any HTML, to prevent XSS (Cross site scripting attacks). We discussed XSS in [Part 55](http://csharp-video-tutorials.blogspot.com/2013/07/part-55-what-is-cross-site-scripting.html) & [Part 56](http://csharp-video-tutorials.blogspot.com/2013/07/part-56-how-to-prevent-cross-site.html) of [asp.net mvc tutorial](http://www.youtube.com/user/kudvenkat/videos?view=1&flow=grid).  
  
However, in some cases, you want the user to be able to submit HTML tags like <b>,<u> etc. For this to happen, we need to turn off request validation, by decorating the action method with ValidateInput attribute as shown below.  
[HttpPost]  
[ValidateInput(false)]  
public string Index(string comments)  
{  
    return "Your Comments: " + comments;  
}  
  
At this point, you should be able to submit comments, with HTML tags in it.