Micah Hoffmann

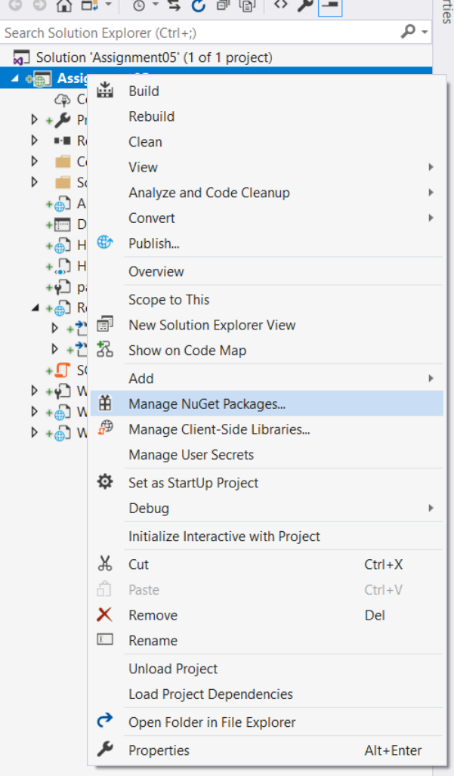
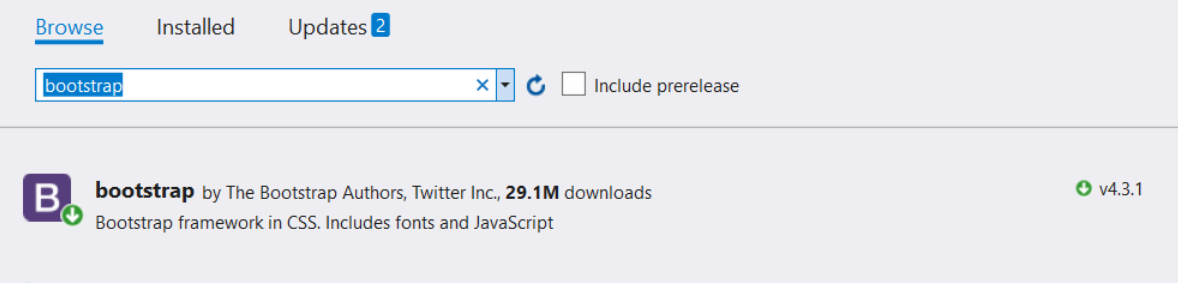
Assignment 05

7/23/2019

This mini project will cover the topic of using client libraries to enhance our project. We are going to use the library Bootstrap to improve the user experience of our web application.

**Use NuGet to get libraries**

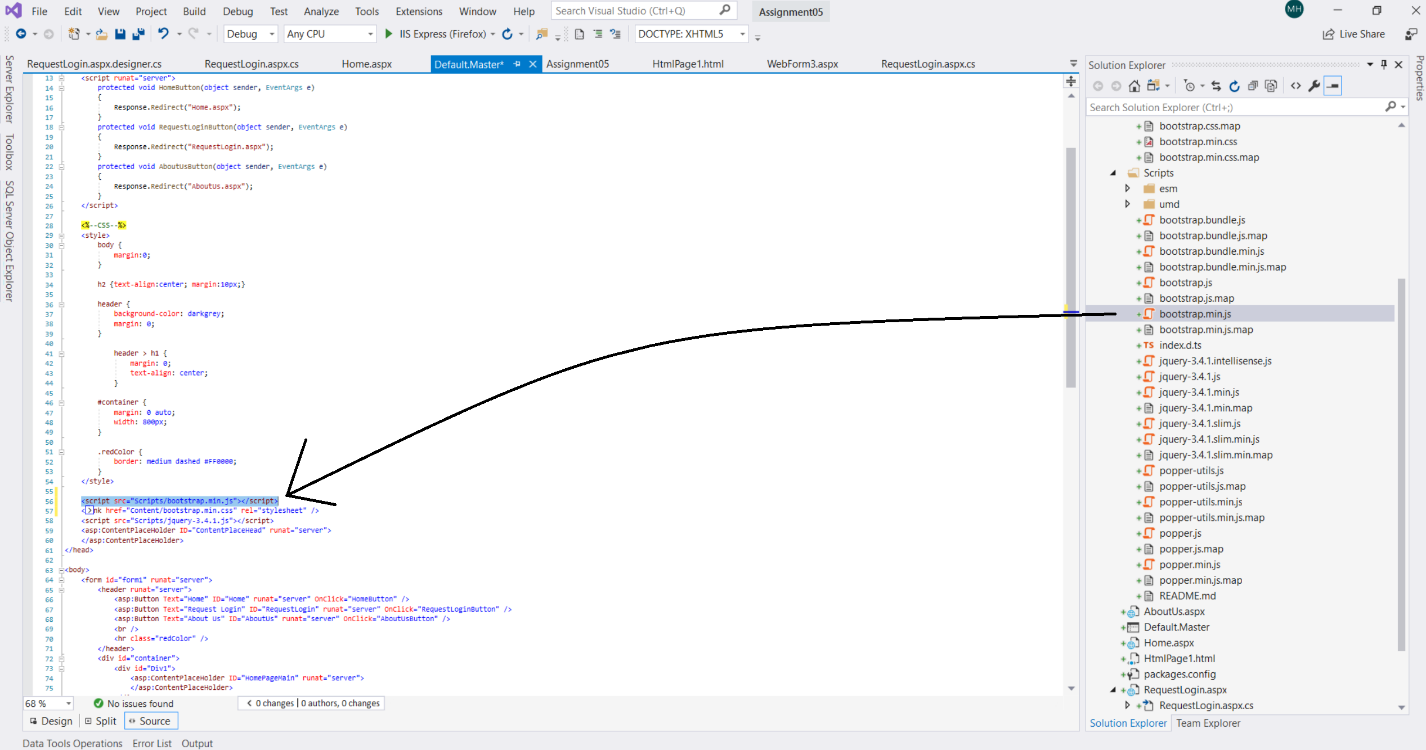
Bootstrap is a very well supported libraries that can be managed via the NuGet package manager in Visual Studio. To access NuGet, right click on the project and click NuGet.

Once in NuGet, Browse for your desired package, and click install. In our case, search for Bootstrap. 

**One the package is installed, add them to our project**

We can easily do this by finding the fill in the project tree, and dragging it into the proper location within our code.

Because we are using the .Master file, if we add it to the <head></head> tag there, that will serve the entire web application.



**Bootstrap**

One of the most power aspects of the Bootstrap framework, is its implementation of columns and building the Bootstrap grid. The bootstrap grid is an easy to implement html tag that can add “responsiveness” to a webpage. Responsiveness is the ability for a webpage to resize and shift its content based on the size of the screen it is being displayed on.

[How it works](https://getbootstrap.com/docs/4.3/layout/grid/)

*Bootstrap’s grid system uses a series of containers, rows, and columns to layout and align content. It’s built with flexbox and is fully responsive. Below is an example and an in-depth look at how the grid comes together.*

<div class="container">

<div class="row">

<div class="col-sm">

One of three columns

</div>

<div class="col-sm">

One of three columns

</div>

<div class="col-sm">

One of three columns

</div>

</div>

</div>

*The above example creates three equal-width columns on small, medium, large, and extra large devices using our predefined grid classes. Those columns are centered in the page with the parent .container.*

So in our case we break the form into rows, label – input field, and we wrap each row in a <div class=”row”>. Within each row we treat the labels as one column, and the input fields as another, so we then apply CssClass values to each column value. Column-sm-3 for the label, and col-sm-9 for the input field. This means the label will take up ¼ of the container width and the input field will take up ¾.