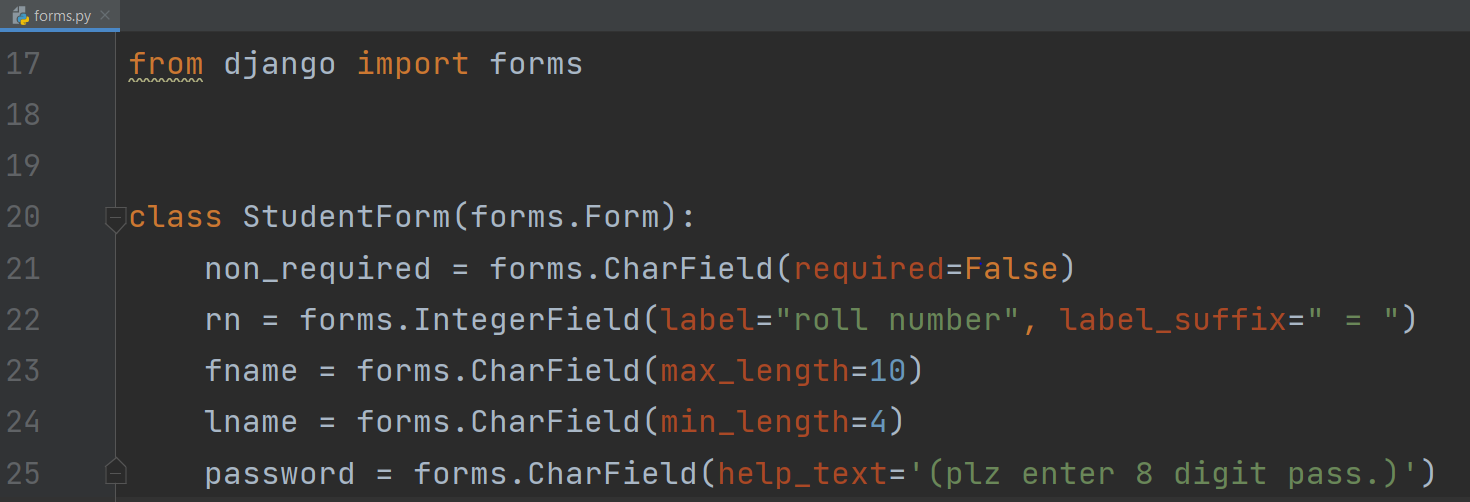
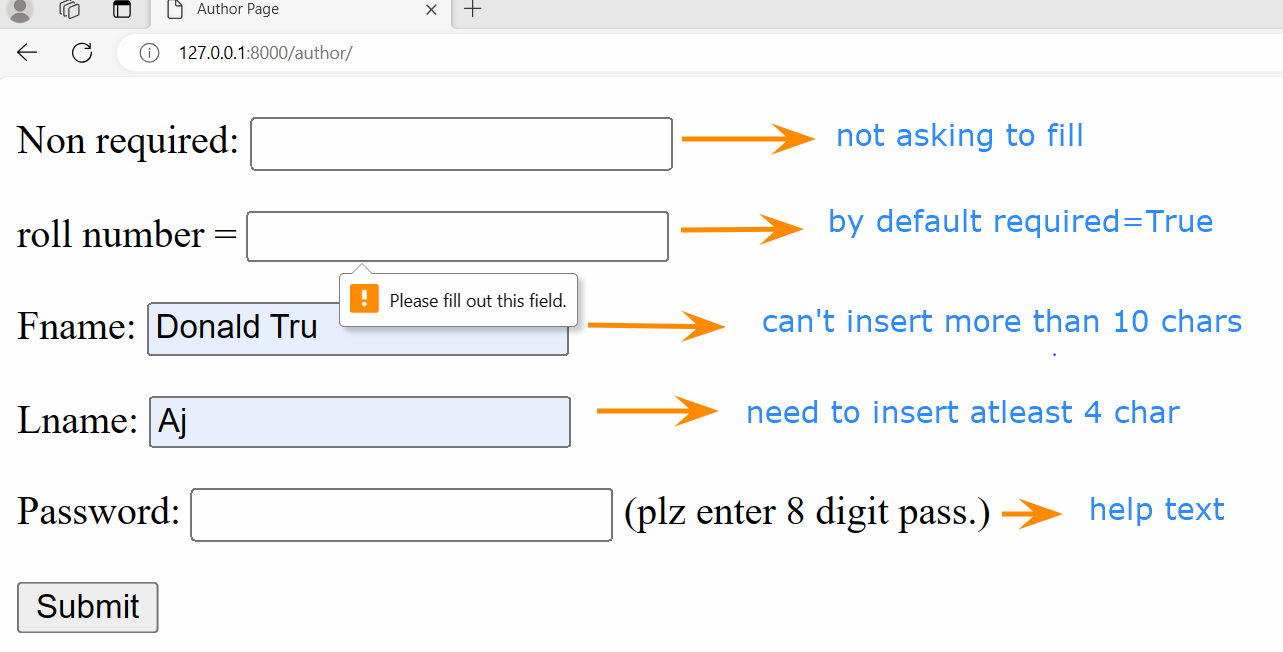
**Some important parameters**

* label --> to change the default label default label is field name.
* label\_suffix --> to change label suffix default suffix is(**:**).
* help\_text --> to provide additional information about input.
* max\_length --> can’t insert more than specified chars.
* min\_length --> can’t insert less than specified chars.
* required --> if assign to false browser will not force to fill this field.





**Widget:**

A widget is Django’s representation of an HTML input element.Whenever you specify a field on a form, Django will use a default widget that is appropriate to the type of data that is to be displayed.

These widgets will help to create more interactive form for any site.

### Default Widget in Form Fields

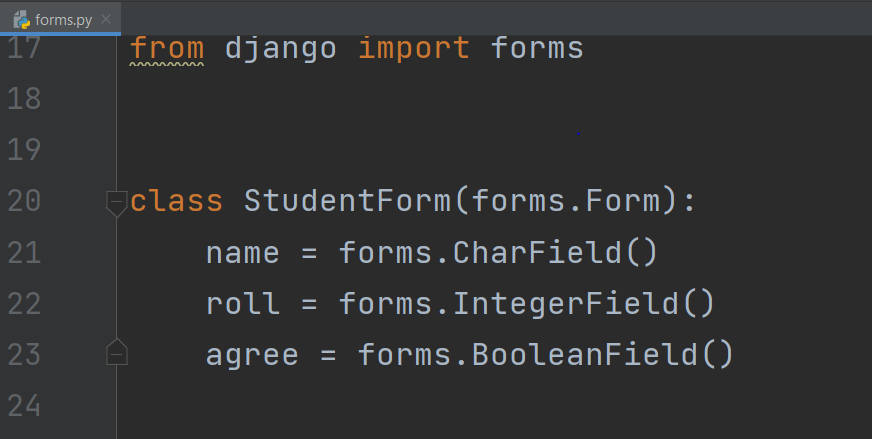
Every field has a default widget.

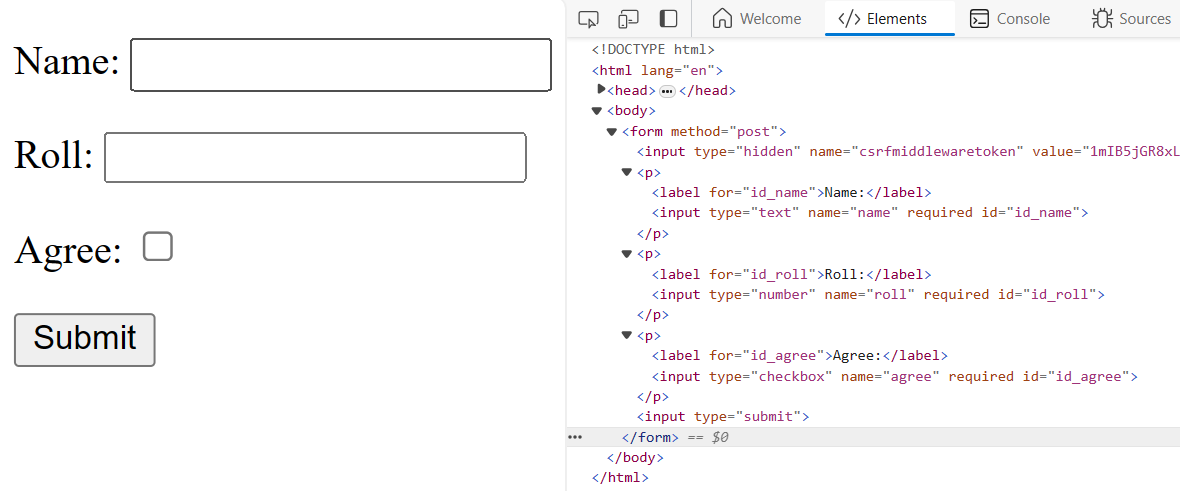
for example:

* **IntegerField** has a default widget of **NumberInput**. and used to create an HTML code <input type = "number"> in the form.
* **Charfield** has a default widget of **TextInput**. and used to create an HTML code <input type = "text"> the form.

**Note**: Django does not provide fields for input type submit and form tag.

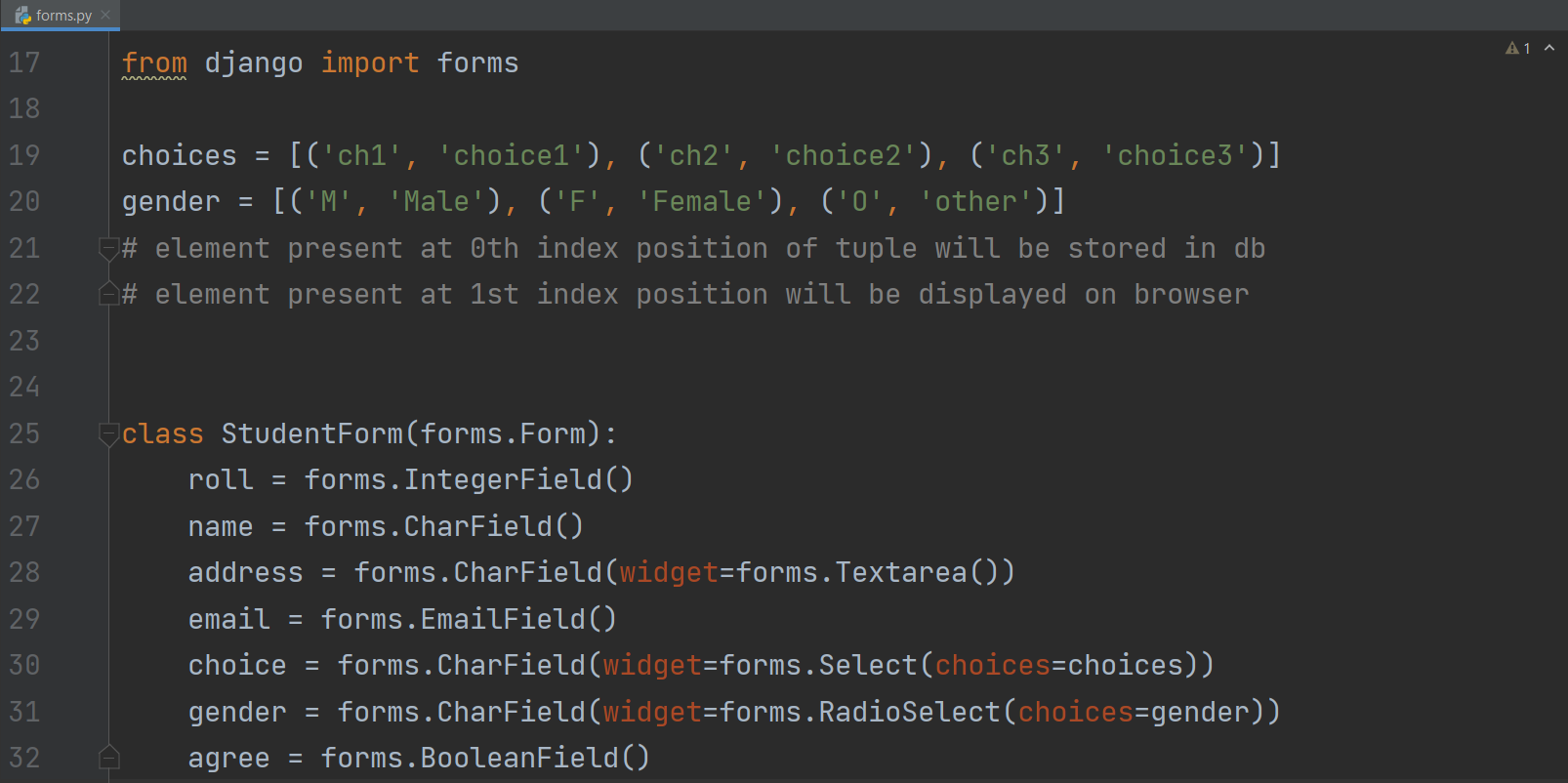
So we have to add them explicitly inside html file.



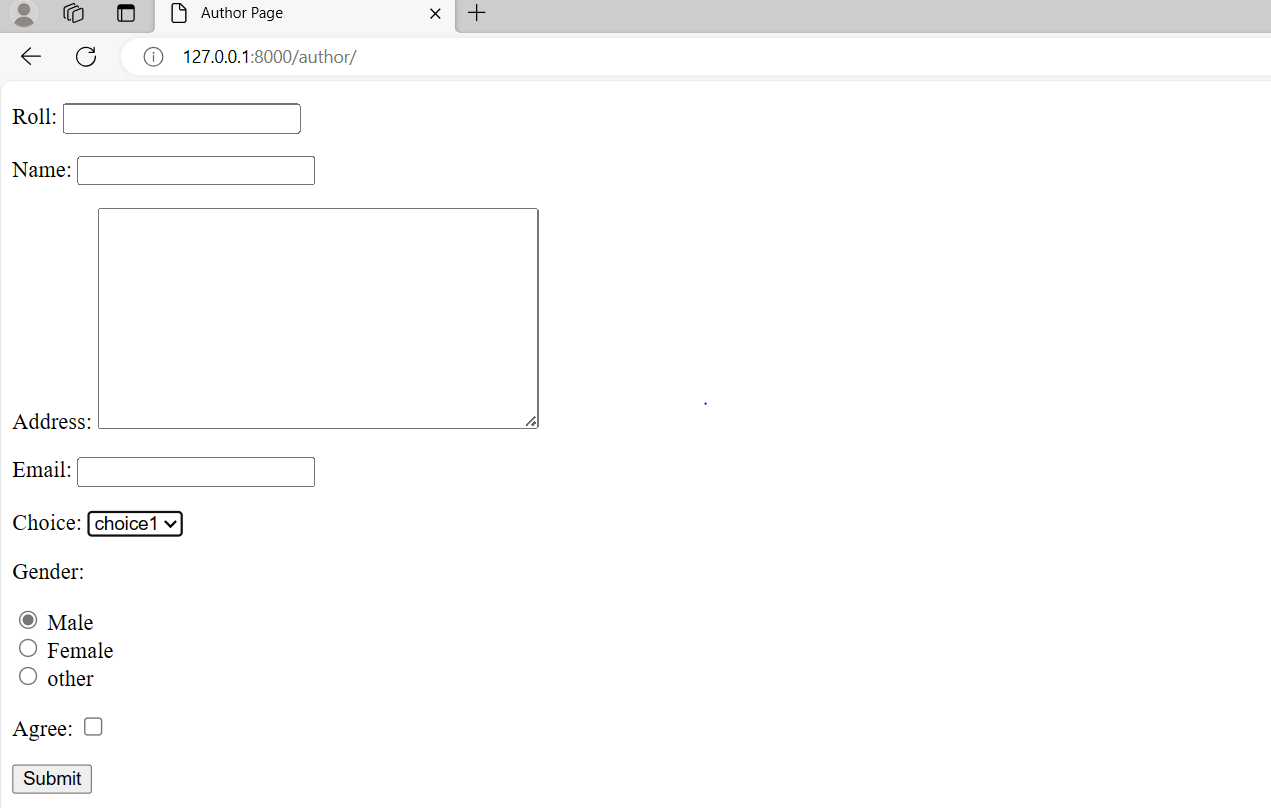


Here in above code snippet you can see input type for respective fields.

We can **override the default widget** of each field for various uses and add attributes.

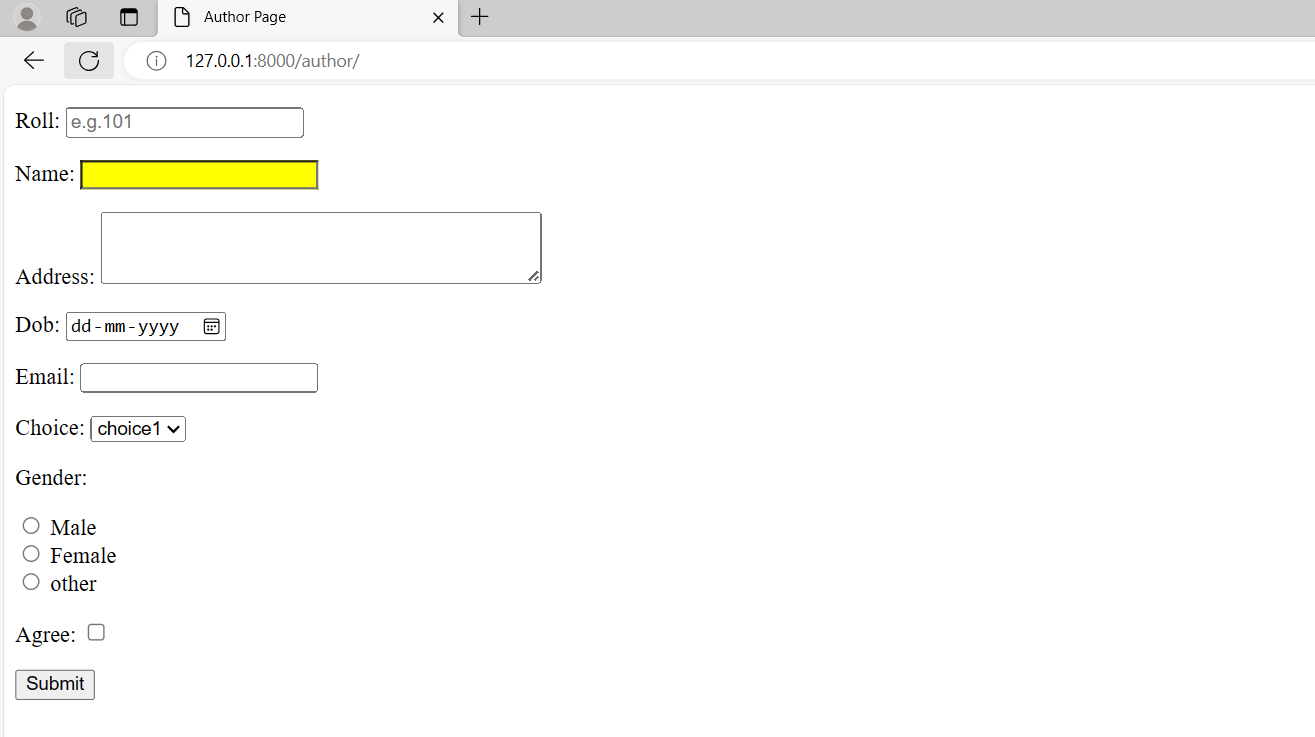


Right click on page and select view page source to see changes in html code



**Adding Attributes:**





lets apply bootstrap classes to make our form more attractive.





