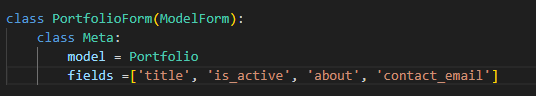
Create a new file called “forms.py”

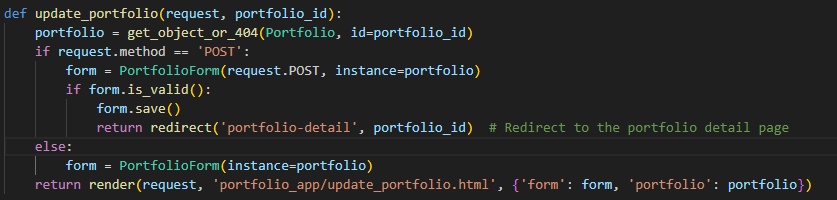


You will need to import everything you see above except for HiddenInput

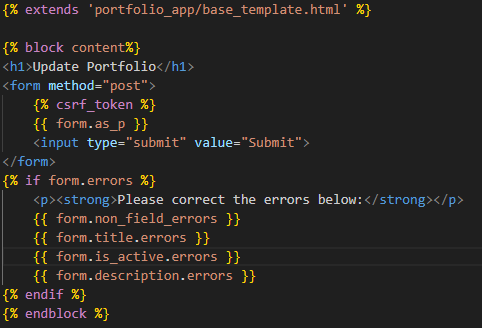
Update Portfolio Form:



This is the “form code” inside of forms.py for the portfolio update form, this can also be used as a way to create portfolios but for now we will just use it as an update form

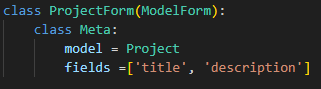


This is the function inside of views.py, this function checks to make sure the correct method is requested and validates which portfolio is being updated, updates the portfolio, then returns the user to the portfolio details page.

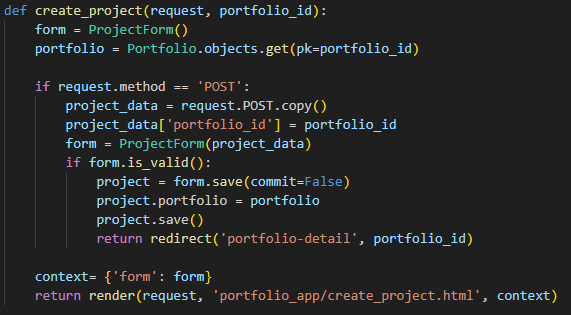


This is the entirety of the update\_portfolio.html template, it includes error checks and will not allow the user to proceed if any errors are present

**Create Project Form:**



This is the form code for both creating and updating projects. Same set-up as portfolio just using project data fields

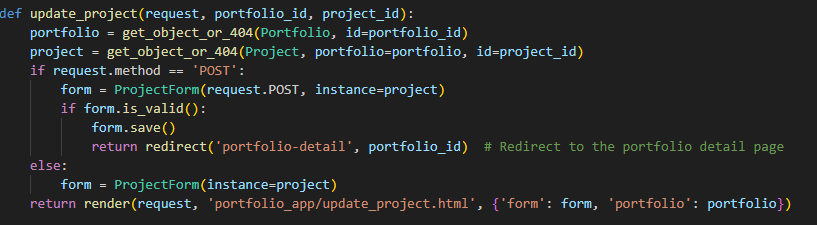


This is the function in views.py for creating a project, using a portfolio id and user-entered fields it will create a new project and assign it a new index # as well as immediately update the database so that it can be viewed on the portfolio details page. When the form is submitted it returns the user to the portfolio-detail page

(update/create/delete project all use the same HTML code as the update portfolio page, just with different text labels)

**Update Project Form:**

Update project uses the same ModelForm as create project

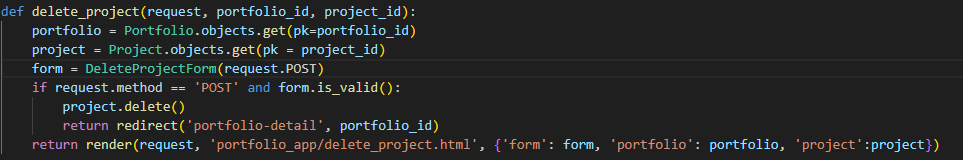


This is the function within views.py, updating the project using a project and portfolio id, it validates all fields before updating and returning the user to the portfolio detail page

**Project Delete Form:**



This is the form, note that it is note a ModelForm as we do not need to input any information or access information from an object



This is the function within views.py, validates the POST request and deletes the project, returning the user to the portfolio detail page afterwards.