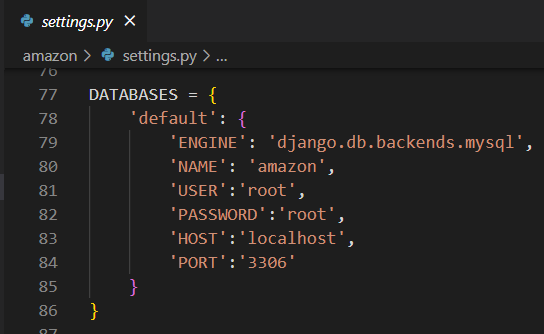
1. Create django project : amazon & open in VS code
2. Open VS code terminal and create Django app :product
3. Register application in settings.py
4. Create database “amazon” in workbench
5. Add database details in settings.py

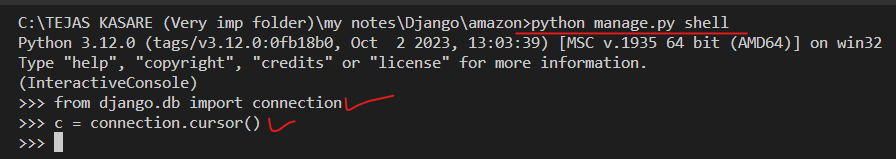


1. Start server
2. Check for database connection(open shell and do it)

>python manage.py shell

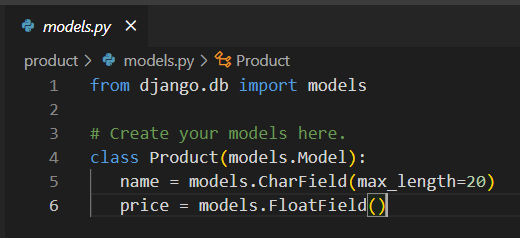
>>>from django.db import connection

>>>c = connection.cursor()

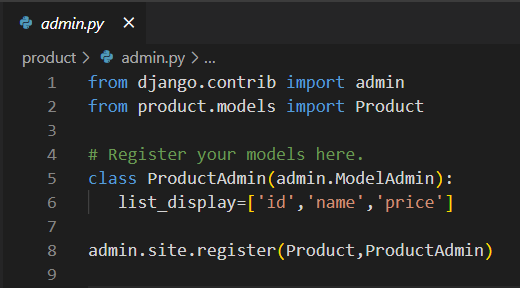


IF NO ERROR THEN GOOD TO GO

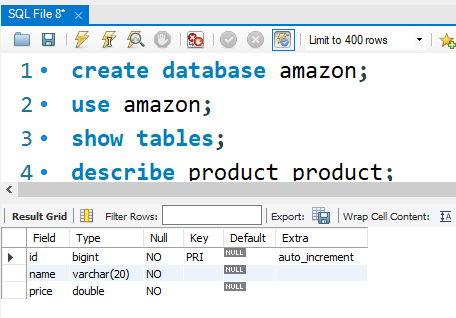
1. In product application, create model class : Product (in model.py) with following fields : name, price

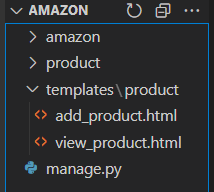


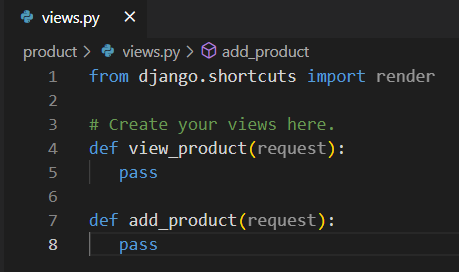
1. Register model class in admin.py



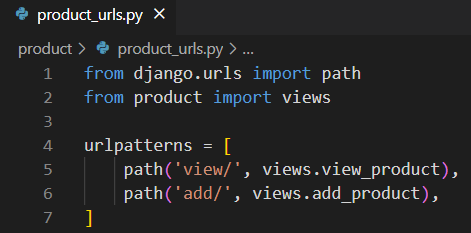
1. Makemigrations
2. Migrate
3. Check for table in workbench



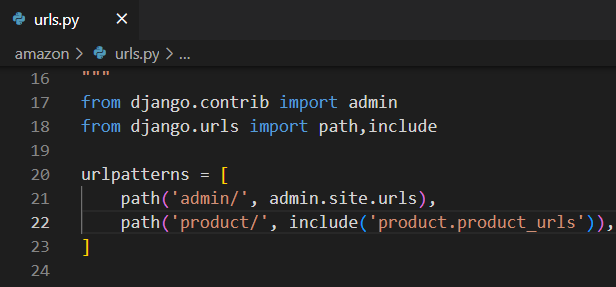
1. Create templates folder and register in settings.py
   1. Create product folder in templates folder
      1. Create add\_product.html and view\_product.html
      2. 
2. Create 2 views in product > views : view\_product , add\_product
   1. Initial code for views



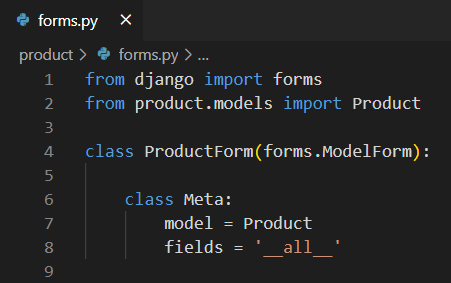
* 1. Create url pattern at application level for above view (create file bloodcamp\_urls.py)



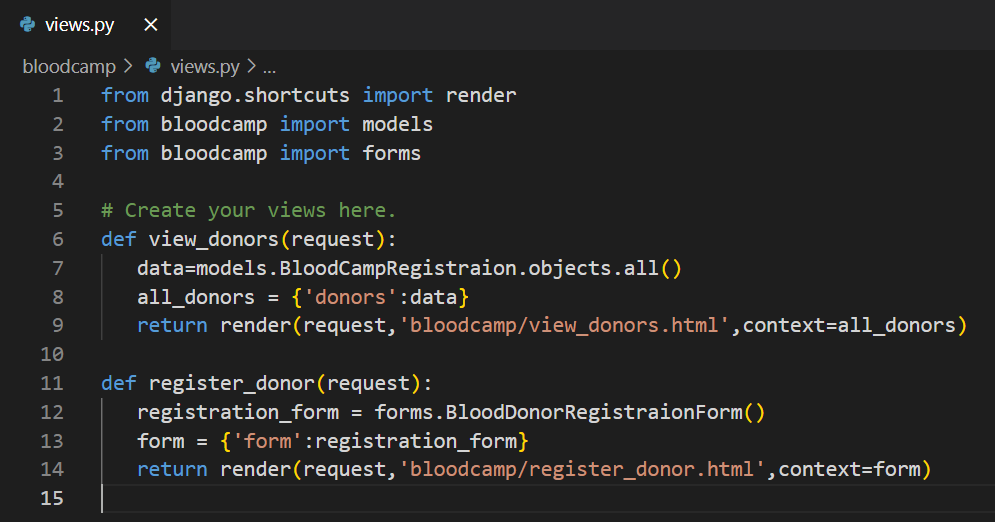
* 1. Include application level urls in project level urls.py



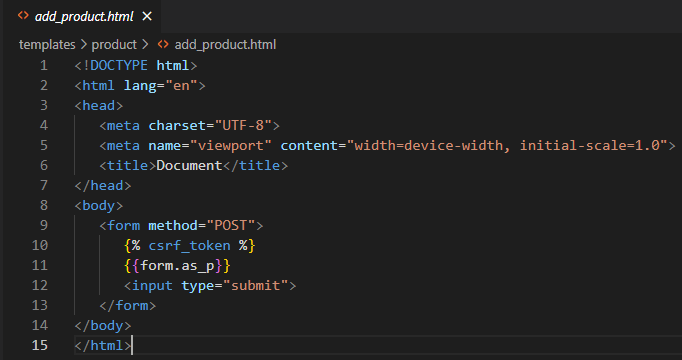
1. Create forms.py at application level (in product folder) and add below code

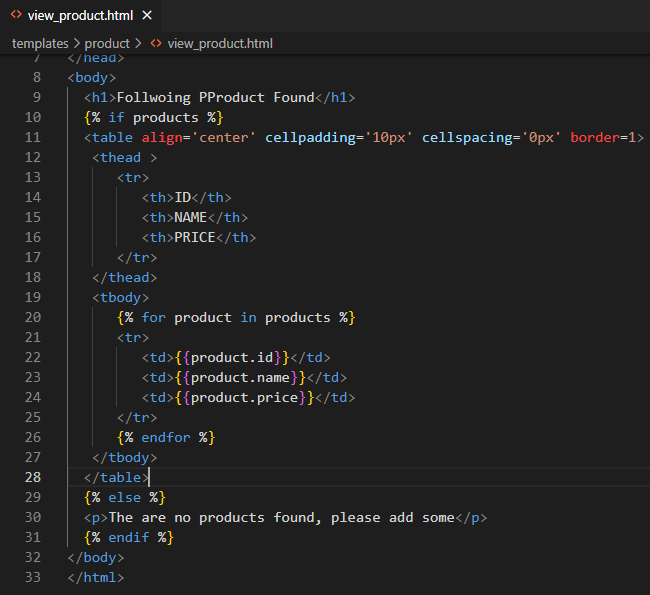


1. Code for above both views (to show products and to add product)

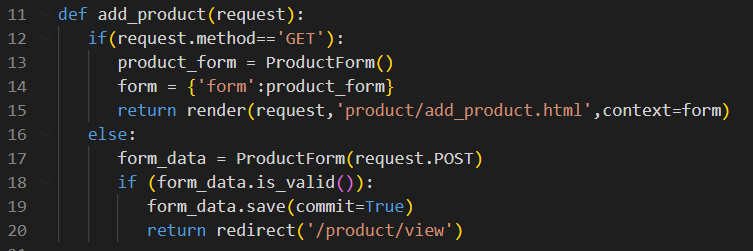


1. Code for above both html files (add\_product.html and view\_product.html)

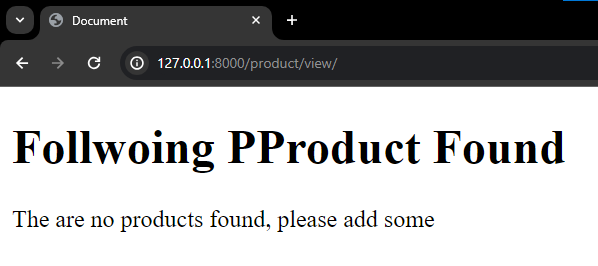
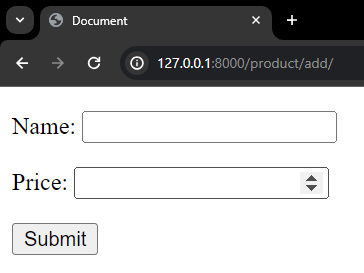




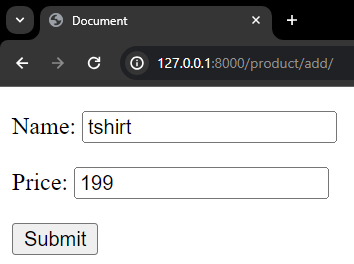
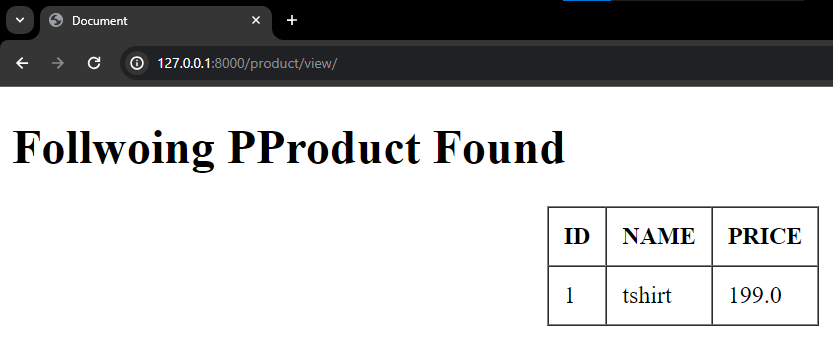
1. Restart sever
2. Write a code to collect data from form in add\_product() view



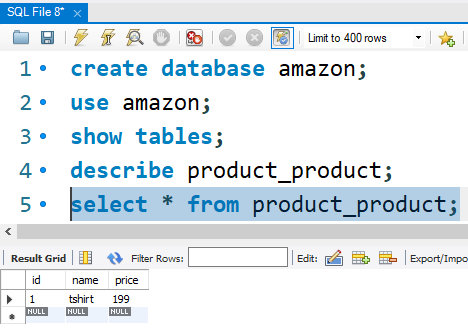
1. Final output

1. DONE
2. Lets add some products

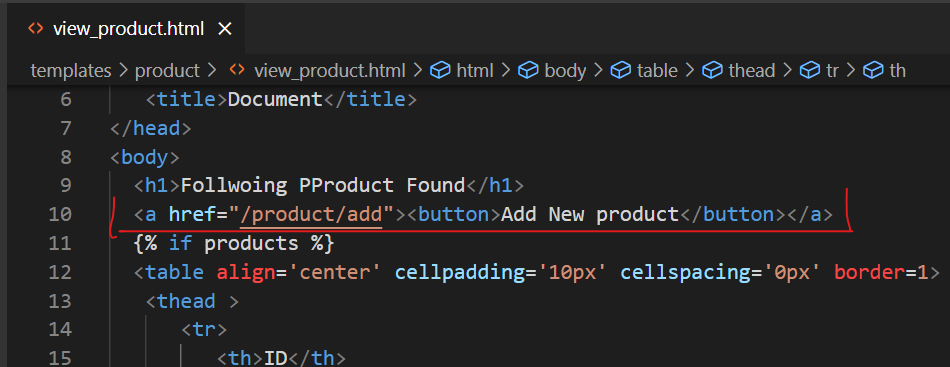
 

**Also Check in workbech :**

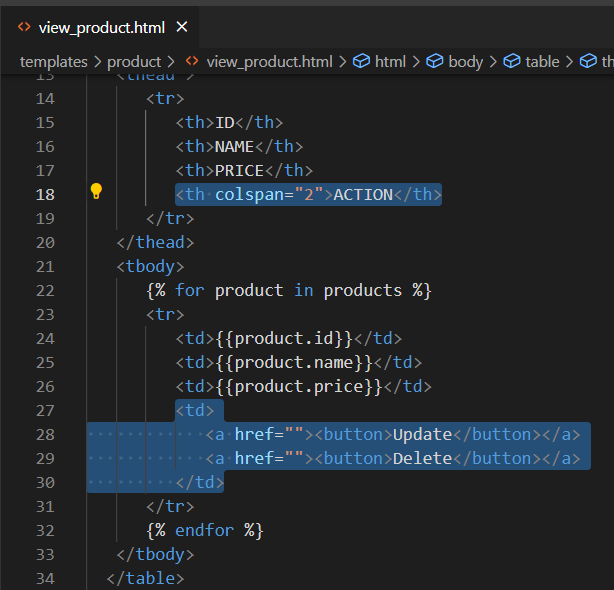


**Implementing CRUD**

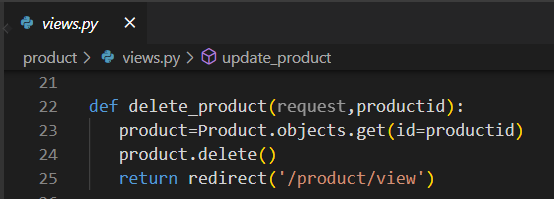
1. Creating add product button in view\_product.html



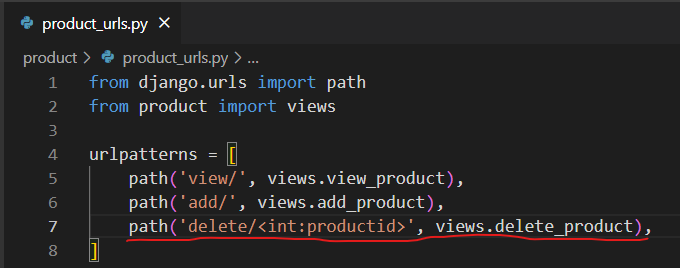
1. Adding update and delete button in the table



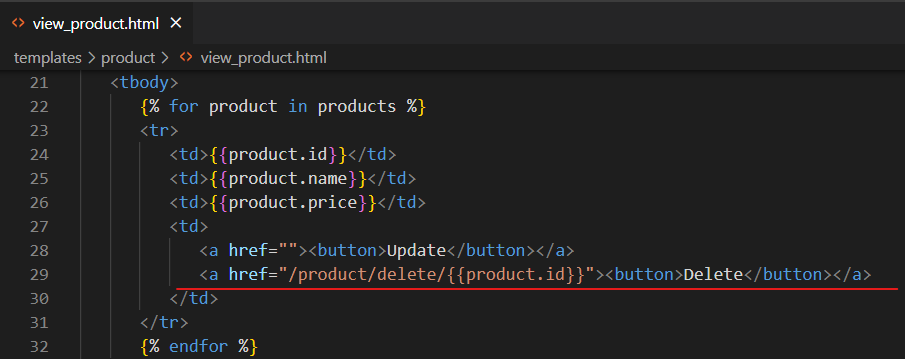
1. Implementing delete logic
   1. Write logic for deleting product : by creating delete\_product() in views.py



* 1. Create url for above view

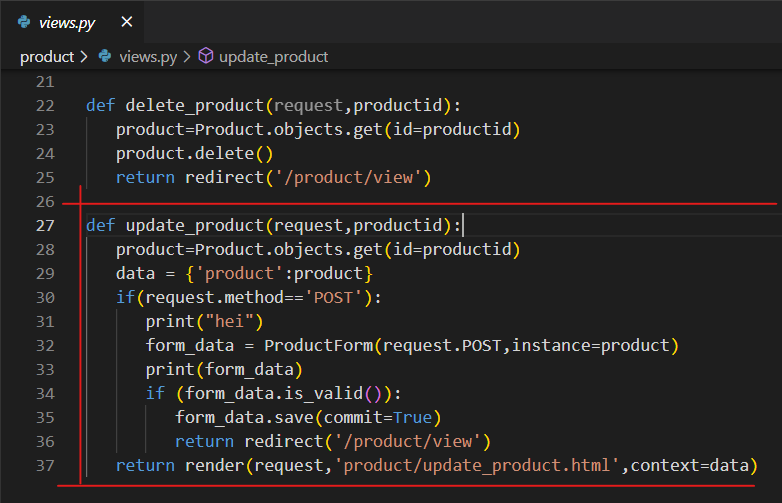


* 1. Use above url in view\_product.html on delete button :

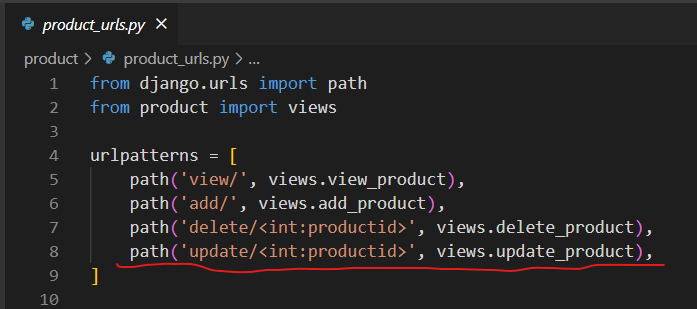


DONE !!!

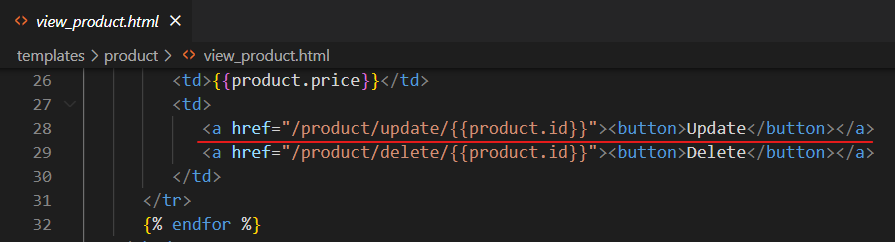
1. Implememting update logic
   1. Write logic for update product by creating update\_product() in views.py



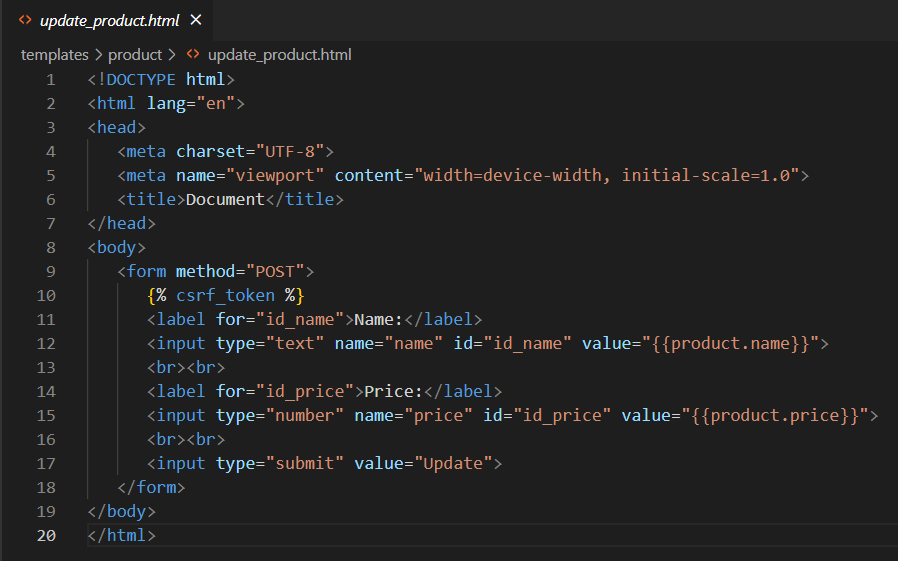
* 1. Create url for above view



* 1. Use above url in view\_product.html on update button



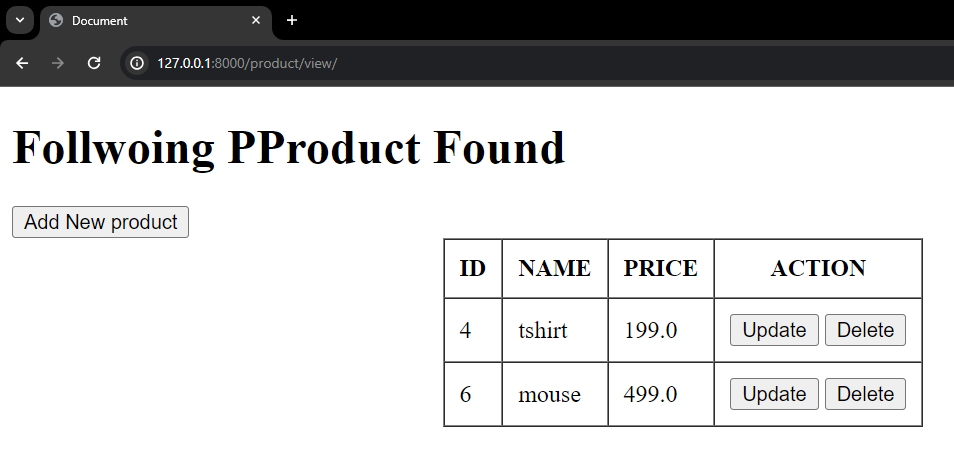
* 1. Create update\_product.html in templates -> product folder



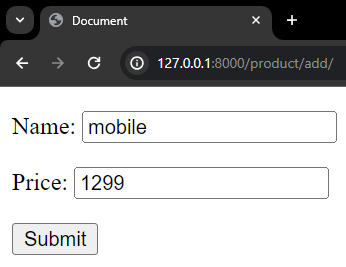
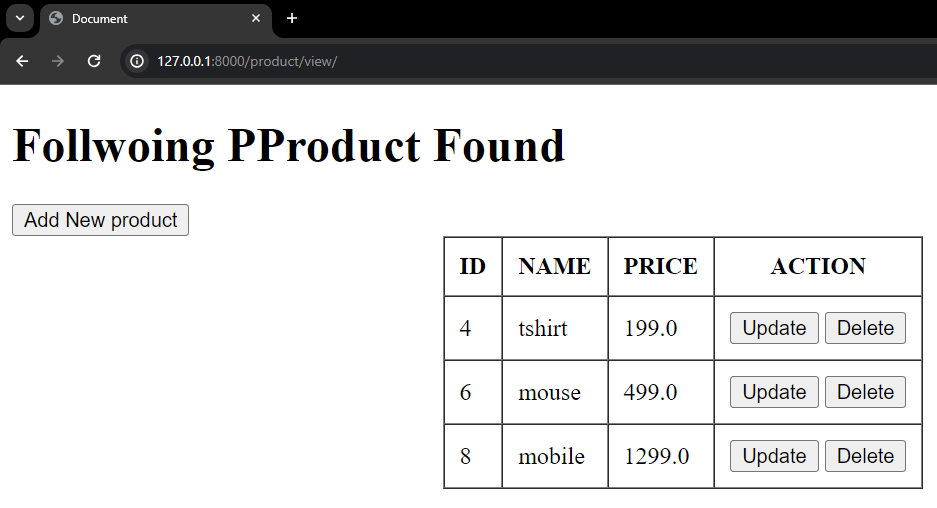
Here, for name attribute, the value must be same as column name in table

Ex : name=”name” , name=”price” then there must be name and price column in the table

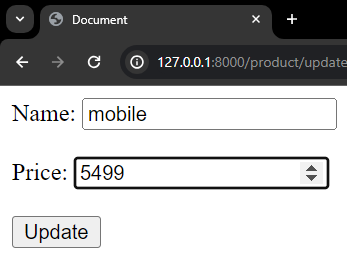
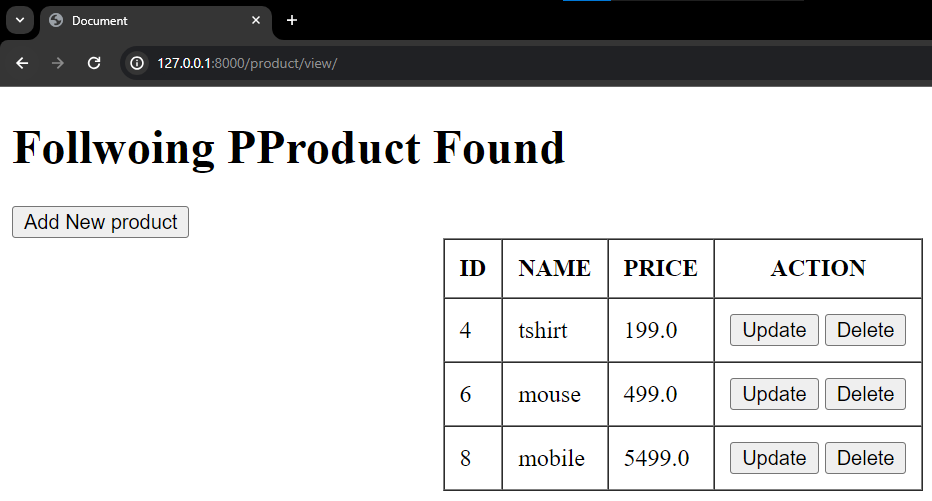
1. Final output



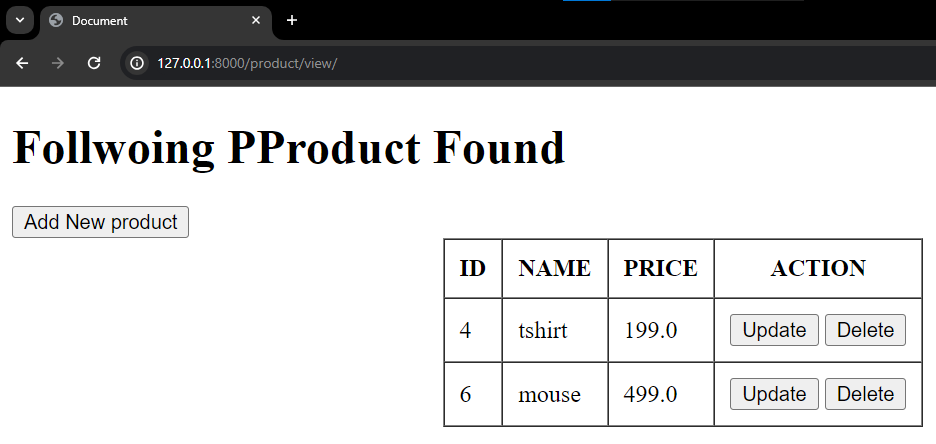
Add New Product :

Update for id = 8 (mobile) => change mobile price from 1299 to 5499

Delete mobile :



***PROJECT DONE !!!!!!!!!!!!!!!!!***