<https://www.youtube.com/watch?v=5BDgKJFZMl8&list=PLu0W_9lII9ah7DDtYtflgwMwpT3xmjXY9&index=1>

**Installation and Getting Started:-+**

pip install Django

create new project textutils

and now create app in it

if not worked use,

python -m django startproject <projectname>

python -m django startproject testutils

create views.py file in testutils

**Views and URLs in DJANGO:-**

In urls.py

from . import views

from . import views  
  
urlpatterns = [  
 path('admin/', admin.site.urls),  
 path('', views.index, name ="index"),  
]

and in views.py

def index():  
 return "Hello"

now when we run we get error as 

because by default we have to get request argument to index

# File is created manually by Sunny  
from django.http import HttpResponse  
  
def index(request):  
 return HttpResponse("Hello")



Now add about

Urls.py

urlpatterns = [  
 path('admin/', admin.site.urls),  
 path('', views.index, name ="index"),  
 path('about', views.about, name ="about"),  
]

Views.py

from django.http import HttpResponse  
  
def index(request):  
 return HttpResponse("Hello")  
  
def about(request):  
 return HttpResponse("About")

//Django first come in Urls.py -> then Views – runs function in views

def index(request):  
 return HttpResponse('''<h1>Hello</h1> <a href="https://getbootstrap.com/docs/5.3/examples/sign-in/"> Sunny's World </a>''')



Urls.py

urlpatterns = [  
 path('admin/', admin.site.urls),  
 path('', views.index, name ="index"),  
 path("removepunc", views.removepunc, name="rempun"),  
 path("capitalizefirst", views.capitalizefirst, name="capfirst"),  
 path("newlineremove", views.newlineremove, name="newlineremove"),  
 path("spaceremove", views.spaceremove, name="spaceremove"),  
 path("charcount", views.charcount, name="charcount"),  
]

Views.py

def index(request):  
 return HttpResponse("Home Page")  
  
def removepunc(request):  
 return HttpResponse('''removepunc <a href = "/"> Go Back to Home Page </a>''')  
  
def capitalizefirst(request):  
 return HttpResponse("capitalizefirst")  
  
def newlineremove(request):  
 return HttpResponse("newlineremove")  
  
def spaceremove(request):  
 return HttpResponse("spaceremove")  
  
def charcount(request):  
 return HttpResponse("charcount")



Add templates in settings.py







IF We want to render elements from dictionary into html page use







/

//Creating zHome Page for TextUtils







When we click on text analyze where is data gone , the page is redirected to removepunc page but we need to store the data somewhere so we use,

Name=”text”



Now when we click submit we see text in url as.





If we click on button without adding the data, it will take default7 as data







Will create Analyze method with multiple checkboxes on page

Index.html

<body>  
 <h1>Welcome to Text Analyzer, Enter your text below</h1>  
<!-- {{ Name }} is from {{ Place }}-->  
<form action='/analyze' method="get">  
 <textarea name="text" style="width: 1376px; height: 202px;">  
 </textarea>  
 <br>  
  
 <input type="checkbox" name="removepunc"> Remove Punctuations </input>  
 <br>  
  
 <button type="submit">  
 Analyze Text  
 </button>  
</form>  
  
</body>

//here action will be analyze,it will goto analyze method

Views.py

def analyze(request):  
 djtext = request.GET.get("text","default")  
 removepunc = request.GET.get("removepunc", "off") #off means default will not be printed in console  
 print(djtext)  
 print(removepunc)  
 if removepunc == "on":  
 punctuations\_list = '''!()-{}[];"'"\,<>./?@#$%^&\*\_~'''  
 analyzed = ""  
 for char in djtext:  
 if char not in punctuations\_list:  
 analyzed = analyzed + char  
 params = {"purpose" : "Removed Punctuations from Sentence", "analyzed\_text" : analyzed}  
 return render(request, "analyze.html", params)  
 else:  
 return HttpResponse("Error")

analyze.html

<body>  
<!-- {{ Name }} is from {{ Place }}-->  
Analyzed Text - {{ purpose }}  
<br><br>  
Final sentence is as below,  
<p>  
 <strong>  
 {{ analyzed\_text }}  
 </strong>  
</p>  
</body>

Adding more featires to textutils website

Added feature of Upper case to full sentence

def analyze(request):  
 # Get the Text  
 djtext = request.GET.get("text","default")  
  
 # Check checkbox value  
 removepunc = request.GET.get("removepunc", "off") #off means default will not be printed in console  
 full\_caps = request.GET.get("full\_caps", "off")  
 print(djtext)  
 print(removepunc)  
  
 # CHeck which checkbox is on  
 if removepunc == "on":  
 punctuations\_list = '''!()-{}[];"'"\,<>./?@#$%^&\*\_~'''  
 analyzed = ""  
 for char in djtext:  
 if char not in punctuations\_list:  
 analyzed = analyzed + char  
 params = {"purpose" : "Removed Punctuations from Sentence", "analyzed\_text" : analyzed}  
 return render(request, "analyze.html", params)  
 elif (full\_caps == "on"):  
 analyzed = ""  
 for char in djtext:  
 analyzed = analyzed + char.upper()  
 params = {"purpose": "Text in Upper Case", "analyzed\_text": analyzed}  
 return render(request, "analyze.html", params)  
 else:  
 return HttpResponse("Error")

index.html

<form action='/analyze' method="get">  
 <textarea name="text" style="width: 1376px; height: 202px;">  
 </textarea>  
 <br>  
 <input type="checkbox" name="removepunc"> Remove Punctuations </input> <br>  
 <input type="checkbox" name="full\_caps"> Upper Case </input>  
 <br>  
  
 <button type="submit">  
 Analyze Text  
 </button>  
</form>

Index.html

<input type="checkbox" name="full\_caps"> Upper Case </input><br>  
<input type="checkbox" name="new\_line\_remover"> New Line Remover</input> <br>  
<input type="checkbox" name="extra\_space\_remover"> Extra Space Remover</input>

Views.py

def analyze(request):  
 # Get the Text  
 djtext = request.GET.get("text","default")  
  
 # Check checkbox value  
 removepunc = request.GET.get("removepunc", "off") #off means default will not be printed in console  
 full\_caps = request.GET.get("full\_caps", "off")  
 new\_line\_remover = request.GET.get("new\_line\_remover", "off")  
 extra\_space\_remover = request.GET.get("extra\_space\_remover", "off")  
 print(djtext)  
 print(removepunc)  
  
 # CHeck which checkbox is on  
 if removepunc == "on":  
 punctuations\_list = '''!()-{}[];"'"\,<>./?@#$%^&\*\_~'''  
 analyzed = ""  
 for char in djtext:  
 if char not in punctuations\_list:  
 analyzed = analyzed + char  
 params = {"purpose" : "Removed Punctuations from Sentence", "analyzed\_text" : analyzed}  
 return render(request, "analyze.html", params)  
 elif (full\_caps == "on"):  
 analyzed = ""  
 for char in djtext:  
 analyzed = analyzed + char.upper()  
 params = {"purpose": "Text in Upper Case", "analyzed\_text": analyzed}  
 return render(request, "analyze.html", params)  
 elif (new\_line\_remover == "on"):  
 analyzed = ""  
 for char in djtext :  
 if char != "\n":  
 analyzed = analyzed + char  
 params = {"purpose": "Remove New Line", "analyzed\_text": analyzed}  
 return render(request, "analyze.html", params)  
 elif (extra\_space\_remover == "on"):  
 analyzed = ""  
 for index, char in enumerate(djtext):  
 if djtext[index] == " " and djtext[index+1] == " ":  
 pass  
 else:  
 analyzed = analyzed + char  
 params = {"purpose": "Extra space removed from sentence", "analyzed\_text": analyzed}  
 return render(request, "analyze.html", params)  
 else:  
 return HttpResponse("Error")

Changed UI – Added Bootstrap

<!doctype html>  
<html lang="en">  
 <head>  
 <meta charset="utf-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>Text Utils</title>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-rbsA2VBKQhggwzxH7pPCaAqO46MgnOM80zW1RWuH61DGLwZJEdK2Kadq2F9CUG65" crossorigin="anonymous">  
 </head>  
 <body>  
<nav class="navbar navbar-expand-lg dark">  
 <div class="container-fluid">  
 <a class="navbar-brand" href="#">Text Utils</a>  
 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavDropdown" aria-controls="navbarNavDropdown" aria-expanded="false" aria-label="Toggle navigation">  
 <span class="navbar-toggler-icon"></span>  
 </button>  
 <div class="collapse navbar-collapse" id="navbarNavDropdown">  
 <ul class="navbar-nav">  
 <li class="nav-item">  
 <a class="nav-link active" aria-current="page" href="/">Home</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="#">About Us</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="#">Contact Us</a>  
 </li>  
  
 </ul>  
 </div>  
 </div>  
 <form class="d-flex" role="search">  
 <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">  
 <button class="btn btn-outline-success" type="submit">Search</button>  
 </form>  
</nav>  
  
<div class="alert alert-success alert-dismissible fade show" role="alert">  
 <strong>Welcome to Text Utils!</strong> You can do anything with text here.  
 <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>  
</div>  
  
  
<div class="container">  
 <form action='/analyze' method="get">  
 <div class="mb-3">  
 <label for="exampleFormControlTextarea1" class="form-label">Enter Your Text Here and let textutils do the magic:- </label>  
 <textarea name="text" class="form-control" id="exampleFormControlTextarea1" rows="9"></textarea>  
 </div>  
  
  
 <div class="form-check form-switch">  
 <input class="form-check-input" name="removepunc" type="checkbox" role="switch" id="flexSwitchCheckChecked1" \>  
 <label class="form-check-label" for="flexSwitchCheckChecked">Remove Punctuations</label>  
 </div>  
 <div class="form-check form-switch">  
 <input class="form-check-input" name="full\_caps" type="checkbox" role="switch" id="flexSwitchCheckChecked2" \>  
 <label class="form-check-label" for="flexSwitchCheckChecked">Upper Case</label>  
 </div>  
 <div class="form-check form-switch">  
 <input class="form-check-input" name="new\_line\_remover" type="checkbox" role="switch" id="flexSwitchCheckChecked3" >  
 <label class="form-check-label" for="flexSwitchCheckChecked">New Line Remover</label>  
 </div>  
 <div class="form-check form-switch">  
 <input class="form-check-input" name="extra\_space\_remover" type="checkbox" role="switch" id="flexSwitchCheckChecked4" >  
 <label class="form-check-label" for="flexSwitchCheckChecked">Extra Space Remover</label>  
 </div>  
 <div class="form-check form-switch">  
 <input class="form-check-input" name="char\_count" type="checkbox" role="switch" id="flexSwitchCheckChecked5" >  
 <label class="form-check-label" for="flexSwitchCheckChecked">Count Characters</label>  
 </div>  
 <button type = "submit" class = "btn btn-dark mt-2">Analyze Text</button>  
 </form>  
</div>  
  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-kenU1KFdBIe4zVF0s0G1M5b4hcpxyD9F7jL+jjXkk+Q2h455rYXK/7HAuoJl+0I4" crossorigin="anonymous"></script>  
 <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js" integrity="sha384-oBqDVmMz9ATKxIep9tiCxS/Z9fNfEXiDAYTujMAeBAsjFuCZSmKbSSUnQlmh/jp3" crossorigin="anonymous"></script>  
 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.min.js" integrity="sha384-cuYeSxntonz0PPNlHhBs68uyIAVpIIOZZ5JqeqvYYIcEL727kskC66kF92t6Xl2V" crossorigin="anonymous"></script>  
 </body>  
</html>

<!doctype html>  
<html lang="en">  
 <head>  
 <meta charset="utf-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>Text Utils</title>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-rbsA2VBKQhggwzxH7pPCaAqO46MgnOM80zW1RWuH61DGLwZJEdK2Kadq2F9CUG65" crossorigin="anonymous">  
 </head>  
 <body>  
<nav class="navbar navbar-expand-lg dark">  
 <div class="container-fluid">  
 <a class="navbar-brand" href="#">Text Utils</a>  
 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavDropdown" aria-controls="navbarNavDropdown" aria-expanded="false" aria-label="Toggle navigation">  
 <span class="navbar-toggler-icon"></span>  
 </button>  
 <div class="collapse navbar-collapse" id="navbarNavDropdown">  
 <ul class="navbar-nav">  
 <li class="nav-item">  
 <a class="nav-link active" aria-current="page" href="#">Home</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="#">About Us</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="#">Contact Us</a>  
 </li>  
  
 </ul>  
 </div>  
 </div>  
 <form class="d-flex" role="search">  
 <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">  
 <button class="btn btn-outline-success" type="submit">Search</button>  
 </form>  
</nav>  
  
<div class="alert alert-success alert-dismissible fade show" role="alert">  
 <strong>Success!!! Your text has been analyzed!</strong> You can do anything with text here.  
 <button type="button" class="btn-close" data-bs-dismiss="alert" aria-label="Close"></button>  
</div>  
  
  
<div class="container">  
Analyzed Text - {{ purpose }}  
<br><br>  
Final sentence is as below,  
 <p>  
 <strong>  
 {{ analyzed\_text }}  
 </strong>  
 </p>  
</div>  
  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js" integrity="sha384-kenU1KFdBIe4zVF0s0G1M5b4hcpxyD9F7jL+jjXkk+Q2h455rYXK/7HAuoJl+0I4" crossorigin="anonymous"></script>  
 <script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js" integrity="sha384-oBqDVmMz9ATKxIep9tiCxS/Z9fNfEXiDAYTujMAeBAsjFuCZSmKbSSUnQlmh/jp3" crossorigin="anonymous"></script>  
 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.min.js" integrity="sha384-cuYeSxntonz0PPNlHhBs68uyIAVpIIOZZ5JqeqvYYIcEL727kskC66kF92t6Xl2V" crossorigin="anonymous"></script>  
 </body>  
</html>

We need to remove BUG from website as when we add text and do remove punctuation





It removes new line also

But we don’t want it to remove new line

Cause is because HTML

As in analyze.html we r directly dumping it as

<div class="container">  
Analyzed Text - {{ purpose }}  
<br><br>  
Final sentence is as below,  
 <p>  
 <strong>  
 {{ analyzed\_text }}  
 </strong>  
 </p>  
</div>

So we have to use <pre> tag in html as it automatically adds same text

<strong>  
<pre>{{ analyzed\_text }} </pre>  
 </strong>

Output now



**Django CSRF Token and Post request:-**

Post request sends additional data to server in message body



When we send unlimited data



So we need to send this type of data as POST request

Change get method to POST in views.py

def analyze(request):  
 # Get the Text  
 djtext = request.POST.get("text","default")

and in index.html use post instead of get





<div class="container">  
 <form action='/analyze' method="post">  
 {% csrf\_token %}

Change GET to POST

# Check checkbox value  
removepunc = request.POST.get("removepunc", "off") #off means default will not be printed in console  
full\_caps = request.POST.get("full\_caps", "off")  
new\_line\_remover = request.POST.get("new\_line\_remover", "off")  
extra\_space\_remover = request.POST.get("extra\_space\_remover", "off")  
char\_count = request.POST.get("char\_count", "off")  
print(djtext)



Now working!!!



elif (new\_line\_remover == "on"):  
 analyzed = ""  
 for char in djtext :  
 if char != "\n" and char != "\r":

new line remover was not working so we add “\r”

now working!!

Now I want to use multiple features at once

def analyze(request):  
 # Get the Text  
 djtext = request.POST.get("text","default")  
  
 # Check checkbox value  
 removepunc = request.POST.get("removepunc", "off") #off means default will not be printed in console  
 full\_caps = request.POST.get("full\_caps", "off")  
 new\_line\_remover = request.POST.get("new\_line\_remover", "off")  
 extra\_space\_remover = request.POST.get("extra\_space\_remover", "off")  
 char\_count = request.POST.get("char\_count", "off")  
 # print(djtext)  
 # print(removepunc)  
  
 # CHeck which checkbox is on  
 if removepunc == "on":  
 punctuations\_list = '''!()-{}[];"'"\,<>./?@#$%^&\*\_~'''  
 analyzed = ""  
 for char in djtext:  
 if char not in punctuations\_list:  
 analyzed = analyzed + char  
 params = {"purpose" : "Removed Punctuations from Sentence", "analyzed\_text" : analyzed}  
 # return render(request, "analyze.html", params)  
 djtext = analyzed  
 # elif (full\_caps == "on"):  
 if (full\_caps == "on"):  
 analyzed = ""  
 for char in djtext:  
 analyzed = analyzed + char.upper()  
 params = {"purpose": "Text in Upper Case", "analyzed\_text": analyzed}  
 # return render(request, "analyze.html", params)  
 djtext = analyzed  
 if (new\_line\_remover == "on"):  
 analyzed = ""  
 for char in djtext :  
 if char != "\n" and char != "\r":  
 analyzed = analyzed + char  
 params = {"purpose": "Remove New Line", "analyzed\_text": analyzed}  
 # return render(request, "analyze.html", params)  
 djtext = analyzed  
 if (extra\_space\_remover == "on"):  
 analyzed = ""  
 for index, char in enumerate(djtext):  
 if djtext[index] == " " and djtext[index+1] == " ":  
 pass  
 else:  
 analyzed = analyzed + char  
 params = {"purpose": "Extra space removed from sentence", "analyzed\_text": analyzed}  
 # return render(request, "analyze.html", params)  
 djtext = analyzed  
 if (char\_count == 'on'):  
 analyzed= 0  
 count = 0  
 for i in djtext:  
 if i!=" ":  
 count += 1  
 analyzed = analyzed + count  
 params = {"purpose": "Counted Characters are, ", "analyzed\_text": analyzed}  
 # return render(request, "analyze.html", params)  
  
 if (removepunc != "on" and full\_caps != "on" and new\_line\_remover != "on" and extra\_space\_remover != "on" and char\_count != "on"):  
 return HttpResponse("Please select operation atleat select one")  
 # else:  
 # return HttpResponse("Error")  
 return render(request, "analyze.html", params)

Hosted on github - <https://github.com/yash9151/textutils_website>

Website 2 :- Ecommerce Website

Create new Project Named – MyAwesomeCart

python -m django startproject mac





http://127.0.0.1:8000/



python manage.py startapp shop

python manage.py startapp blog





Now add urls.py and views.py in both apps blog and shop

Shop\urls.py

from django.urls import path  
from . import views  
  
urlpatterns = [  
 path('', views.index, name="ShopHome"),  
]

Shop\views.py

from django.http import HttpResponse  
# Create your views here.  
def index(request):  
 return HttpResponse("Index Shop")

Blog\urls.py

from django.shortcuts import render  
from django.shortcuts import HttpResponse  
# Create your views here.  
def index(request):  
 return HttpResponse("In Blog Home Page")

Blog\views.py

from django.shortcuts import render  
from django.shortcuts import HttpResponse  
# Create your views here.  
def index(request):  
 return HttpResponse("In Blog Home Page")

Also create templates folder in both



Again create folder in templets in both name accrodinglt to appname

As,

same for Shop 

Now we need to add our app in installed apps in main project

Mac\Settings,py

INSTALLED\_APPS = [  
 'django.contrib.admin',  
 'django.contrib.auth',  
 'django.contrib.contenttypes',  
 'django.contrib.sessions',  
 'django.contrib.messages',  
 'django.contrib.staticfiles',  
 'blog',  
 'shop'  
]

Create index.html in templte\blog folder

Copy starter templte in it

Now in

Blog\views.py render it

from django.shortcuts import render  
from django.shortcuts import HttpResponse  
# Create your views here.  
def index(request):  
 return render(request, "blog/index.html")

Doing same for shop app

Shop\views.py

from django.shortcuts import render  
from django.http import HttpResponse  
# Create your views here.  
def index(request):  
 return render(request, "shop/index.html")

Create Static directory in both





Mystatic.txt

This is my\_static file in blog.

To load static files in Django we use

<h1>Hello, world!</h1>  
{% load static %}  
<a href="{%static 'blog/mystatic.txt' %}"> CLick Me</a>





Now created methods in shop\urls.py

from django.urls import path  
from . import views  
  
urlpatterns = [  
 path('', views.index, name="ShopHome"),  
 path('about/', views.about, name="AboutUs"),  
 path('contact', views.contact, name="ContactUs"),  
 path('tracker', views.tracker, name="TrackingStatus"),  
 path('search', views.search, name="Search"),  
 path('productview', views.prodView, name="ProductView"),  
 path('checkout', views.checkout, name="Checkout"),  
]

shop\views.py

from django.shortcuts import render  
from django.http import HttpResponse  
# Create your views here.  
def index(request):  
 return render(request, "shop/index.html")  
  
def about(request):  
 return HttpResponse("We are in about page of shop")  
  
def contact(request):  
 return HttpResponse("We are in contact page of shop")  
  
def tracker(request):  
 return HttpResponse("We are in tracker page of shop")  
  
def search(request):  
 return HttpResponse("We are in search page of shop")  
  
def prodView(request):  
 return HttpResponse("We are in productview page of shop")  
  
def checkout(request):  
 return HttpResponse("We are in checkout page of shop")

//Now will create templates for all pages

//Models in Django

python manage.py migrate

//this will create tables which are needed in project i.e, change we done in models will get saved without changing database

//if we apply migrations then it will change tables

Shop\Models.py

class Product(models.Model):  
 product\_id = models.AutoField  
 product\_name = models.CharField(max\_length=50)  
 desc = models.CharField(max\_length=300)  
 pub\_date = models.DateField()

as in apps.py of shop app we can see

class ShopConfig(AppConfig):  
 default\_auto\_field = 'django.db.models.BigAutoField'  
 name = 'shop'

so in main projects settings.py file we can add this,

INSTALLED\_APPS = [  
 'django.contrib.admin',  
 'django.contrib.auth',  
 'django.contrib.contenttypes',  
 'django.contrib.sessions',  
 'django.contrib.messages',  
 'django.contrib.staticfiles',  
 'blog',  
 'shop.apps.ShopConfig'  
]

> python manage.py makemigrations





//this is changed so store it in database, change is tables is added named Product



Now to check whether everthing is good in database , we need to be admin

SO will create SuperUserto see Models in database



Password = sunny

Now when we open admin website



There is no Product that we created

So to add it we need to register model in admin.py in shop

from django.contrib import admin  
  
# Register your models here.  
from .models import Product  
  
admin.site.register(Product)



Will create index page for shop app

<https://getbootstrap.com/docs/5.3/components/navbar/>



<!doctype html>  
<html lang="en">  
<head>  
 <meta charset="utf-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>Bootstrap demo</title>  
 <style>  
 .col-md-3  
 {  
 display:inline-block;  
 margin-left:-4px;  
 }  
  
 .col-md-3 img{  
  
 width:100%;  
 height:auto;  
 }  
  
 carousel-indicators .active{  
 background-color:blue;  
 }  
  
 body .carousel-indicator li{  
 background-color: blue;  
 }  
 body .carousel-indicators{  
 bottom:0;  
 }  
  
 body .carousel-control-prev-icon,  
 body .carousel-control-next-icon{  
 background-color: blue;  
 }  
 body .no-padding{  
 padding-left:0,  
 padding-right:0;  
 }  
  
  
 </style>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet"  
 integrity="sha384-rbsA2VBKQhggwzxH7pPCaAqO46MgnOM80zW1RWuH61DGLwZJEdK2Kadq2F9CUG65" crossorigin="anonymous">  
</head>  
<body>  
{% load static %}  
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">  
 <div class="container-fluid">  
 <a class="navbar-brand" href="#">My Awesome Cart</a>  
 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"  
 aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">  
 <span class="navbar-toggler-icon"></span>  
 </button>  
 <div class="collapse navbar-collapse" id="navbarSupportedContent">  
 <ul class="navbar-nav me-auto mb-2 mb-lg-0">  
 <li class="nav-item">  
 <a class="nav-link active" aria-current="page" href="#">Home</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="#">Link</a>  
 </li>  
 <li class="nav-item dropdown">  
 <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown"  
 aria-expanded="false">  
 Dropdown  
 </a>  
 <ul class="dropdown-menu">  
 <li><a class="dropdown-item" href="#">Action</a></li>  
 <li><a class="dropdown-item" href="#">Another action</a></li>  
 <li>  
 <hr class="dropdown-divider">  
 </li>  
 <li><a class="dropdown-item" href="#">Something else here</a></li>  
 </ul>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link disabled" aria-disabled="true">Disabled</a>  
 </li>  
 </ul>  
 <form class="d-flex" role="search">  
 <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">  
 <button class="btn btn-outline-success" type="submit">Search</button>  
 </form>  
 </div>  
 </div>  
</nav>  
  
<div class="container">  
  
 <div id="demo" class="carousel slide my-3" data-ride="carousel">  
 <ul class="carousel-indicators">  
 <li data-bs-target="#demo" data-bs-slide-to="0" class="active"></li>  
 <li data-bs-target="#demo" data-bs-slide-to="1"></li>  
 <li data-bs-target="#demo" data-bs-slide-to="2"></li>  
 </ul>  
  
 <div class="carousel-inner">  
 <div class="carousel-item active">  
 <div class="row">  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
  
  
 <div class="carousel-item">  
 <div class="row">  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
  
  
 <div class="carousel-item">  
  
 <div class="row">  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
</div>  
<a class="carousel-control-prev " href="#demo" role="button" data-bs-slide="prev">  
 <span class="carousel-control-prev-icon" aria-hidden="true"></span>  
 <span class="sr-only">Previous</span>  
</a>  
<a class="carousel-control-next" href="#demo" role="button" data-bs-slide="next">  
 <span class="carousel-control-next-icon" aria-hidden="true"></span>  
 <span class="sr-only">Next</span>  
</a>  
  
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js"  
 integrity="sha384-oBqDVmMz9ATKxIep9tiCxS/Z9fNfEXiDAYTujMAeBAsjFuCZSmKbSSUnQlmh/jp3"  
 crossorigin="anonymous"></script>  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.min.js"  
 integrity="sha384-cuYeSxntonz0PPNlHhBs68uyIAVpIIOZZ5JqeqvYYIcEL727kskC66kF92t6Xl2V"  
 crossorigin="anonymous"></script>  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js"  
 integrity="sha384-kenU1KFdBIe4zVF0s0G1M5b4hcpxyD9F7jL+jjXkk+Q2h455rYXK/7HAuoJl+0I4"  
 crossorigin="anonymous"></script>  
</body>  
</html>



//Adding images and changing models

Models.py in shop

from django.db import models  
  
# Create your models here.  
class Product(models.Model):  
 product\_id = models.AutoField  
 product\_name = models.CharField(max\_length=50)  
 category = models.CharField(max\_length=50, default="")  
 subcategory = models.CharField(max\_length=50, default="")  
 price = models.IntegerField(default=0)  
 desc = models.CharField(max\_length=300)  
 pub\_date = models.DateField()  
 image = models.ImageField(upload\_to = "shop/images", default="")

now make migrations





When we run admin



But we want names so we use

from django.db import models  
  
# Create your models here.  
class Product(models.Model):  
 product\_id = models.AutoField  
 product\_name = models.CharField(max\_length=50)  
 category = models.CharField(max\_length=50, default="")  
 subcategory = models.CharField(max\_length=50, default="")  
 price = models.IntegerField(default=0)  
 desc = models.CharField(max\_length=300)  
 pub\_date = models.DateField()  
 image = models.ImageField(upload\_to = "shop/images", default="")  
  
 def \_\_str\_\_(self):  
 return self.product\_name





**//Adding Media directory in Django**

settings.py in mac (main project)

# Managing Media by Sunny  
MEDIA\_ROOT = os.path.join(BASE\_DIR, 'media')  
MEDIA\_URL = '/media/'

Urls.py in main project\

Add

] + static(settings.MEDIA\_URL, document\_root=settings.MEDIA\_ROOT)

And import line 3 and 4

from django.contrib import admin  
from django.urls import path, include  
from django.conf import settings  
from django.conf.urls.static import static  
  
urlpatterns = [  
 path('admin/', admin.site.urls),  
 path('shop/', include("shop.urls")),  
 path('blog/', include("blog.urls")),  
] + static(settings.MEDIA\_URL, document\_root=settings.MEDIA\_ROOT)

now we can see image in admin panel and also it has created folder media, in base directory to store images,



It created shop/images folder becoz we have added this in models.py of shop app as,

image = models.ImageField(upload\_to = "shop/images", default="")

**/fetching products using python manage.py shell**



>>> from django.utils import timezone

>>> myprod = Product(product\_name = "mouse", category = "computer accessories", subcategory = "Devices", price = 500, desc = "mouse to be used", pub\_date = timezone.now())

>>> myprod.save()

>>> myprod

<Product: mouse>

>>> myprod.product\_id

<class 'django.db.models.fields.AutoField'>

>>> Product.objects.all()

<QuerySet [<Product: watch>, <Product: Ghadi>, <Product: df>, <Product: watch2>, <Product: mouse>]>

>>> Product.objects.get(product\_name="mouse")

<Product: mouse>

>>> a= Product.objects.get(product\_name="mouse")

>>> a.category

'computer accessories'

>>>

**//Template Inheritance**

****

**Basic.html**

<!doctype html>  
<html lang="en">  
<head>  
 <meta charset="utf-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>{%block title%} {% endblock %}</title>  
 <style>  
 {%block css%} {% endblock %}  
 </style>  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet"  
 integrity="sha384-rbsA2VBKQhggwzxH7pPCaAqO46MgnOM80zW1RWuH61DGLwZJEdK2Kadq2F9CUG65" crossorigin="anonymous">  
</head>  
<body>  
{% load static %}  
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">  
 <div class="container-fluid">  
 <a class="navbar-brand" href="#">My Awesome Cart</a>  
 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"  
 aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">  
 <span class="navbar-toggler-icon"></span>  
 </button>  
 <div class="collapse navbar-collapse" id="navbarSupportedContent">  
 <ul class="navbar-nav me-auto mb-2 mb-lg-0">  
 <li class="nav-item">  
 <a class="nav-link active" aria-current="page" href="#">Home</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="#">Link</a>  
 </li>  
 <li class="nav-item dropdown">  
 <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown"  
 aria-expanded="false">  
 Dropdown  
 </a>  
 <ul class="dropdown-menu">  
 <li><a class="dropdown-item" href="#">Action</a></li>  
 <li><a class="dropdown-item" href="#">Another action</a></li>  
 <li>  
 <hr class="dropdown-divider">  
 </li>  
 <li><a class="dropdown-item" href="#">Something else here</a></li>  
 </ul>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link disabled" aria-disabled="true">Disabled</a>  
 </li>  
 </ul>  
 <form class="d-flex" role="search">  
 <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">  
 <button class="btn btn-outline-success" type="submit">Search</button>  
 </form>  
 </div>  
 </div>  
</nav>  
  
{%block body%} {% endblock %}  
  
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js"  
 integrity="sha384-oBqDVmMz9ATKxIep9tiCxS/Z9fNfEXiDAYTujMAeBAsjFuCZSmKbSSUnQlmh/jp3"  
 crossorigin="anonymous"></script>  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.min.js"  
 integrity="sha384-cuYeSxntonz0PPNlHhBs68uyIAVpIIOZZ5JqeqvYYIcEL727kskC66kF92t6Xl2V"  
 crossorigin="anonymous"></script>  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js"  
 integrity="sha384-kenU1KFdBIe4zVF0s0G1M5b4hcpxyD9F7jL+jjXkk+Q2h455rYXK/7HAuoJl+0I4"  
 crossorigin="anonymous"></script>  
</body>  
</html>

**Index.html of shop app**

{% extends 'shop/basic.html' %}  
{%block css%} .col-md-3  
{  
display:inline-block;  
margin-left:-4px;  
}  
  
.col-md-3 img{  
  
width:100%;  
height:auto;  
}  
  
carousel-indicators .active{  
background-color:blue;  
}  
  
body .carousel-indicator li{  
background-color: blue;  
}  
body .carousel-indicators{  
bottom:0;  
}  
  
body .carousel-control-prev-icon,  
body .carousel-control-next-icon{  
background-color: blue;  
}  
body .no-padding{  
padding-left:0,  
padding-right:0;  
}  
  
{% endblock %}  
  
  
{%block body%}  
{% load static %}  
<div class="container">  
  
 <div id="demo" class="carousel slide my-3" data-ride="carousel">  
 <ul class="carousel-indicators">  
 <li data-bs-target="#demo" data-bs-slide-to="0" class="active"></li>  
 <li data-bs-target="#demo" data-bs-slide-to="1"></li>  
 <li data-bs-target="#demo" data-bs-slide-to="2"></li>  
 </ul>  
  
 <div class="carousel-inner">  
 <div class="carousel-item active">  
 <div class="row">  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
  
  
 <div class="carousel-item">  
 <div class="row">  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
  
  
 <div class="carousel-item">  
  
 <div class="row">  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">Card title</h5>  
 <p class="card-text">Some quick example text to build on the card title and make up the  
 bulk of the  
 card's content.</p>  
 <a href="#" class="btn btn-primary">Go somewhere</a>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
 </div>  
</div>  
<a class="carousel-control-prev " href="#demo" role="button" data-bs-slide="prev">  
 <span class="carousel-control-prev-icon" aria-hidden="true"></span>  
 <span class="sr-only">Previous</span>  
</a>  
<a class="carousel-control-next" href="#demo" role="button" data-bs-slide="next">  
 <span class="carousel-control-next-icon" aria-hidden="true"></span>  
 <span class="sr-only">Next</span>  
</a>  
  
  
{% endblock %}  
  
  
<!--<!doctype html>-->  
<!--<html lang="en">-->  
<!--<head>-->  
<!-- <meta charset="utf-8">-->  
<!-- <meta name="viewport" content="width=device-width, initial-scale=1">-->  
<!-- <title>Bootstrap demo</title>-->  
<!-- <style>-->  
<!-- .col-md-3-->  
<!-- {-->  
<!-- display:inline-block;-->  
<!-- margin-left:-4px;-->  
<!-- }-->  
  
<!-- .col-md-3 img{-->  
  
<!-- width:100%;-->  
<!-- height:auto;-->  
<!-- }-->  
  
<!-- carousel-indicators .active{-->  
<!-- background-color:blue;-->  
<!-- }-->  
  
<!-- body .carousel-indicator li{-->  
<!-- background-color: blue;-->  
<!-- }-->  
<!-- body .carousel-indicators{-->  
<!-- bottom:0;-->  
<!-- }-->  
  
<!-- body .carousel-control-prev-icon,-->  
<!-- body .carousel-control-next-icon{-->  
<!-- background-color: blue;-->  
<!-- }-->  
<!-- body .no-padding{-->  
<!-- padding-left:0,-->  
<!-- padding-right:0;-->  
<!-- }-->  
  
  
<!-- </style>-->  
<!-- <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet"-->  
<!-- integrity="sha384-rbsA2VBKQhggwzxH7pPCaAqO46MgnOM80zW1RWuH61DGLwZJEdK2Kadq2F9CUG65" crossorigin="anonymous">-->  
<!--</head>-->  
<!--<body>-->  
<!--{% load static %}-->  
<!--<nav class="navbar navbar-expand-lg navbar-dark bg-dark">-->  
<!-- <div class="container-fluid">-->  
<!-- <a class="navbar-brand" href="#">My Awesome Cart</a>-->  
<!-- <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"-->  
<!-- aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">-->  
<!-- <span class="navbar-toggler-icon"></span>-->  
<!-- </button>-->  
<!-- <div class="collapse navbar-collapse" id="navbarSupportedContent">-->  
<!-- <ul class="navbar-nav me-auto mb-2 mb-lg-0">-->  
<!-- <li class="nav-item">-->  
<!-- <a class="nav-link active" aria-current="page" href="#">Home</a>-->  
<!-- </li>-->  
<!-- <li class="nav-item">-->  
<!-- <a class="nav-link" href="#">Link</a>-->  
<!-- </li>-->  
<!-- <li class="nav-item dropdown">-->  
<!-- <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown"-->  
<!-- aria-expanded="false">-->  
<!-- Dropdown-->  
<!-- </a>-->  
<!-- <ul class="dropdown-menu">-->  
<!-- <li><a class="dropdown-item" href="#">Action</a></li>-->  
<!-- <li><a class="dropdown-item" href="#">Another action</a></li>-->  
<!-- <li>-->  
<!-- <hr class="dropdown-divider">-->  
<!-- </li>-->  
<!-- <li><a class="dropdown-item" href="#">Something else here</a></li>-->  
<!-- </ul>-->  
<!-- </li>-->  
<!-- <li class="nav-item">-->  
<!-- <a class="nav-link disabled" aria-disabled="true">Disabled</a>-->  
<!-- </li>-->  
<!-- </ul>-->  
<!-- <form class="d-flex" role="search">-->  
<!-- <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">-->  
<!-- <button class="btn btn-outline-success" type="submit">Search</button>-->  
<!-- </form>-->  
<!-- </div>-->  
<!-- </div>-->  
<!--</nav>-->  
  
<!--<div class="container">-->  
  
<!-- <div id="demo" class="carousel slide my-3" data-ride="carousel">-->  
<!-- <ul class="carousel-indicators">-->  
<!-- <li data-bs-target="#demo" data-bs-slide-to="0" class="active"></li>-->  
<!-- <li data-bs-target="#demo" data-bs-slide-to="1"></li>-->  
<!-- <li data-bs-target="#demo" data-bs-slide-to="2"></li>-->  
<!-- </ul>-->  
  
<!-- <div class="carousel-inner">-->  
<!-- <div class="carousel-item active">-->  
<!-- <div class="row">-->  
<!-- <div class="col-xs-3 col-sm-3 col-md-3">-->  
<!-- <div class="card" style="width: 18rem;">-->  
<!-- <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">-->  
<!-- <div class="card-body">-->  
<!-- <h5 class="card-title">Card title</h5>-->  
<!-- <p class="card-text">Some quick example text to build on the card title and make up the-->  
<!-- bulk of the-->  
<!-- card's content.</p>-->  
<!-- <a href="#" class="btn btn-primary">Go somewhere</a>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- <div class="col-xs-3 col-sm-3 col-md-3">-->  
<!-- <div class="card" style="width: 18rem;">-->  
<!-- <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">-->  
<!-- <div class="card-body">-->  
<!-- <h5 class="card-title">Card title</h5>-->  
<!-- <p class="card-text">Some quick example text to build on the card title and make up the-->  
<!-- bulk of the-->  
<!-- card's content.</p>-->  
<!-- <a href="#" class="btn btn-primary">Go somewhere</a>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- <div class="col-xs-3 col-sm-3 col-md-3">-->  
<!-- <div class="card" style="width: 18rem;">-->  
<!-- <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">-->  
<!-- <div class="card-body">-->  
<!-- <h5 class="card-title">Card title</h5>-->  
<!-- <p class="card-text">Some quick example text to build on the card title and make up the-->  
<!-- bulk of the-->  
<!-- card's content.</p>-->  
<!-- <a href="#" class="btn btn-primary">Go somewhere</a>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- <div class="col-xs-3 col-sm-3 col-md-3">-->  
<!-- <div class="card" style="width: 18rem;">-->  
<!-- <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">-->  
<!-- <div class="card-body">-->  
<!-- <h5 class="card-title">Card title</h5>-->  
<!-- <p class="card-text">Some quick example text to build on the card title and make up the-->  
<!-- bulk of the-->  
<!-- card's content.</p>-->  
<!-- <a href="#" class="btn btn-primary">Go somewhere</a>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
  
  
<!-- <div class="carousel-item">-->  
<!-- <div class="row">-->  
<!-- <div class="col-xs-3 col-sm-3 col-md-3">-->  
<!-- <div class="card" style="width: 18rem;">-->  
<!-- <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">-->  
<!-- <div class="card-body">-->  
<!-- <h5 class="card-title">Card title</h5>-->  
<!-- <p class="card-text">Some quick example text to build on the card title and make up the-->  
<!-- bulk of the-->  
<!-- card's content.</p>-->  
<!-- <a href="#" class="btn btn-primary">Go somewhere</a>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- <div class="col-xs-3 col-sm-3 col-md-3">-->  
<!-- <div class="card" style="width: 18rem;">-->  
<!-- <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">-->  
<!-- <div class="card-body">-->  
<!-- <h5 class="card-title">Card title</h5>-->  
<!-- <p class="card-text">Some quick example text to build on the card title and make up the-->  
<!-- bulk of the-->  
<!-- card's content.</p>-->  
<!-- <a href="#" class="btn btn-primary">Go somewhere</a>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- <div class="col-xs-3 col-sm-3 col-md-3">-->  
<!-- <div class="card" style="width: 18rem;">-->  
<!-- <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">-->  
<!-- <div class="card-body">-->  
<!-- <h5 class="card-title">Card title</h5>-->  
<!-- <p class="card-text">Some quick example text to build on the card title and make up the-->  
<!-- bulk of the-->  
<!-- card's content.</p>-->  
<!-- <a href="#" class="btn btn-primary">Go somewhere</a>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- <div class="col-xs-3 col-sm-3 col-md-3">-->  
<!-- <div class="card" style="width: 18rem;">-->  
<!-- <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">-->  
<!-- <div class="card-body">-->  
<!-- <h5 class="card-title">Card title</h5>-->  
<!-- <p class="card-text">Some quick example text to build on the card title and make up the-->  
<!-- bulk of the-->  
<!-- card's content.</p>-->  
<!-- <a href="#" class="btn btn-primary">Go somewhere</a>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
  
  
<!-- <div class="carousel-item">-->  
  
<!-- <div class="row">-->  
<!-- <div class="col-xs-3 col-sm-3 col-md-3">-->  
<!-- <div class="card" style="width: 18rem;">-->  
<!-- <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">-->  
<!-- <div class="card-body">-->  
<!-- <h5 class="card-title">Card title</h5>-->  
<!-- <p class="card-text">Some quick example text to build on the card title and make up the-->  
<!-- bulk of the-->  
<!-- card's content.</p>-->  
<!-- <a href="#" class="btn btn-primary">Go somewhere</a>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- <div class="col-xs-3 col-sm-3 col-md-3">-->  
<!-- <div class="card" style="width: 18rem;">-->  
<!-- <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">-->  
<!-- <div class="card-body">-->  
<!-- <h5 class="card-title">Card title</h5>-->  
<!-- <p class="card-text">Some quick example text to build on the card title and make up the-->  
<!-- bulk of the-->  
<!-- card's content.</p>-->  
<!-- <a href="#" class="btn btn-primary">Go somewhere</a>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- <div class="col-xs-3 col-sm-3 col-md-3">-->  
<!-- <div class="card" style="width: 18rem;">-->  
<!-- <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">-->  
<!-- <div class="card-body">-->  
<!-- <h5 class="card-title">Card title</h5>-->  
<!-- <p class="card-text">Some quick example text to build on the card title and make up the-->  
<!-- bulk of the-->  
<!-- card's content.</p>-->  
<!-- <a href="#" class="btn btn-primary">Go somewhere</a>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- <div class="col-xs-3 col-sm-3 col-md-3">-->  
<!-- <div class="card" style="width: 18rem;">-->  
<!-- <img src="{%static 'shop/test.jpg' %}" class="card-img-top" alt="...">-->  
<!-- <div class="card-body">-->  
<!-- <h5 class="card-title">Card title</h5>-->  
<!-- <p class="card-text">Some quick example text to build on the card title and make up the-->  
<!-- bulk of the-->  
<!-- card's content.</p>-->  
<!-- <a href="#" class="btn btn-primary">Go somewhere</a>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!-- </div>-->  
<!--</div>-->  
<!--<a class="carousel-control-prev " href="#demo" role="button" data-bs-slide="prev">-->  
<!-- <span class="carousel-control-prev-icon" aria-hidden="true"></span>-->  
<!-- <span class="sr-only">Previous</span>-->  
<!--</a>-->  
<!--<a class="carousel-control-next" href="#demo" role="button" data-bs-slide="next">-->  
<!-- <span class="carousel-control-next-icon" aria-hidden="true"></span>-->  
<!-- <span class="sr-only">Next</span>-->  
<!--</a>-->  
  
<!--<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js"-->  
<!-- integrity="sha384-oBqDVmMz9ATKxIep9tiCxS/Z9fNfEXiDAYTujMAeBAsjFuCZSmKbSSUnQlmh/jp3"-->  
<!-- crossorigin="anonymous"></script>-->  
<!--<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.min.js"-->  
<!-- integrity="sha384-cuYeSxntonz0PPNlHhBs68uyIAVpIIOZZ5JqeqvYYIcEL727kskC66kF92t6Xl2V"-->  
<!-- crossorigin="anonymous"></script>-->  
<!--<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js"-->  
<!-- integrity="sha384-kenU1KFdBIe4zVF0s0G1M5b4hcpxyD9F7jL+jjXkk+Q2h455rYXK/7HAuoJl+0I4"-->  
<!-- crossorigin="anonymous"></script>-->  
<!--</body>-->  
<!--</html>-->

**About.html**

{% extends 'shop/basic.html' %}  
  
{%block title%} Title of About Page {% endblock %}  
  
{%block body%} Body bhai{% endblock %}

**//Logic for showing Products on HomePage**

****

****

**//Showing product on home page**

****

**Now we want to fetch images from database into home page**

**In Views.py in Shop app**

from .models import Product  
from math import ceil  
  
# Create your views here.  
def index(request):  
 # fetches products from database  
 products = Product.objects.all()  
 print(products)  
 n=len(products)  
 nSlides = n//4 + ceil((n/4) - (n//4))  
 params = {'product' : products, 'no\_of\_slides':nSlides, 'range' : range(1, nSlides)}  
 return render(request, "shop/index.html", params)

**//Now we want all variables from params to be included in index.html of shop app**

**Shop.index.html**

{% extends 'shop/basic.html' %}  
  
{% block css %}  
.col-md-3 {  
display: inline-block;  
margin-left: -4px;  
}  
  
.col-md-3 img {  
width: 255px;  
height: 255px;  
}  
  
.carousel-indicators .active {  
background-color: blue;  
}  
  
body .carousel-indicator li {  
background-color: blue;  
}  
  
body .carousel-indicators {  
bottom: 0;  
}  
  
body .carousel-control-prev-icon,  
body .carousel-control-next-icon {  
background-color: blue;  
}  
  
body .no-padding {  
padding-left: 0;  
padding-right: 0;  
}  
{% endblock %}  
  
{% block body %}  
{% load static %}  
<div class="container">  
 <h1>Mens Fashion</h1>  
 <div id="demo" class="carousel slide my-3" data-ride="carousel">  
 <ul class="carousel-indicators">  
 <li data-bs-target="#demo" data-bs-slide-to="0" class="active"></li>  
  
 {% for i in range %}  
 <li data-bs-target="#demo" data-bs-slide-to="{{ i }}"></li>  
 {% endfor %}  
 </ul>  
  
 <div class="carousel-inner">  
 <div class="carousel-item active">  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="/media/{{product.0.image}}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">{{product.0.product\_name}}</h5>  
 <p class="card-text">{{product.0.desc}}</p>  
 <a href="#" class="btn btn-primary">Add to Cart</a>  
 </div>  
 </div>  
 </div>  
  
 {% for product\_item in product|slice:"1:"%}  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="/media/{{product\_item.image}}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">{{product\_item.product\_name}}</h5>  
 <p class="card-text">{{product\_item.desc}}</p>  
 <a href="#" class="btn btn-primary">Add to Cart</a>  
 </div>  
 </div>  
 </div>  
 {% if forloop.counter|add:1|divisibleby:4 and forloop.counter > 0 and not forloop.last %}  
 </div>  
 <div class="carousel-item">  
 {% endif %}  
  
 {% endfor %}  
 </div>  
 </div>  
 </div>  
 <a class="carousel-control-prev" href="#demo" role="button" data-bs-slide="prev">  
 <span class="carousel-control-prev-icon" aria-hidden="true"></span>  
 <span class="sr-only">Previous</span>  
 </a>  
 <a class="carousel-control-next" href="#demo" role="button" data-bs-slide="next">  
 <span class="carousel-control-next-icon" aria-hidden="true"></span>  
 <span class="sr-only">Next</span>  
 </a>  
  
</div>  
{% endblock %}

**Views.py**

from .models import Product  
from math import ceil  
  
# Create your views here.  
def index(request):  
 # fetches products from database  
 products = Product.objects.all()  
 print(products)  
 n=len(products)  
 nSlides = n//4 + ceil((n/4) - (n//4))  
 params = {'product' : products, 'no\_of\_slides':nSlides, 'range' : range(1, nSlides)}  
 return render(request, "shop/index.html", params)

****

**//Showing multiple product slide on home page**

**Index.html**

def index(request):  
 # fetches products from database  
 products = Product.objects.all()  
 print(products)  
 n=len(products)  
 nSlides = n//4 + ceil((n/4) - (n//4))  
 # params = {'product' : products, 'no\_of\_slides':nSlides, 'range' : range(1, nSlides)}  
  
 allProds = [[products, range(1, nSlides), nSlides],  
 [products, range(1, nSlides), nSlides]]  
 params = {'allProds' : allProds}  
 return render(request, "shop/index.html", params)

**shop/index.html**

{% extends 'shop/basic.html' %}  
  
{% block css %}  
.col-md-3 {  
display: inline-block;  
margin-left: -4px;  
}  
  
.col-md-3 img {  
width: 255px;  
height: 255px;  
}  
  
.carousel-indicators .active {  
background-color: blue;  
}  
  
body .carousel-indicator li {  
background-color: blue;  
}  
  
body .carousel-indicators {  
bottom: 0;  
}  
  
body .carousel-control-prev-icon,  
body .carousel-control-next-icon {  
background-color: blue;  
}  
  
body .carousel-control-prev,  
body .carousel-control-next {  
top: auto;  
bottom: auto;  
}  
  
body .no-padding {  
padding-left: 0;  
padding-right: 0;  
}  
{% endblock %}  
  
{% block body %}  
{% load static %}  
<div class="container">  
  
<!-- Slide Show starts here-->  
 {% for product, range, nSlides in allProds %}  
 <h1>{{product.0.category}}</h1>  
 <div id="demo{{forloop.counter}}" class="carousel slide my-3" data-ride="carousel">  
 <ul class="carousel-indicators">  
 <li data-bs-target="#demo{{forloop.counter}}" data-bs-slide-to="0" class="active"></li>  
  
 {% for i in range %}  
 <li data-bs-target="#demo{{forloop.parentloop.counter}}" data-bs-slide-to="{{i}}"></li>  
 {% endfor %}  
 </ul>  
  
 <div class="container carousel-inner no-padding">  
 <div class="carousel-item active">  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="/media/{{product.0.image}}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">{{product.0.product\_name}}</h5>  
 <p class="card-text">{{product.0.desc}}</p>  
 <a href="#" class="btn btn-primary">Add to Cart</a>  
 </div>  
 </div>  
 </div>  
  
 {% for i in product|slice:"1:"%}  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="/media/{{i.image}}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">{{i.product\_name}}</h5>  
 <p class="card-text">{{i.desc}}</p>  
 <a href="#" class="btn btn-primary">Add to Cart</a>  
 </div>  
 </div>  
 </div>  
 {% if forloop.counter|add:1|divisibleby:4 and forloop.counter > 0 and not forloop.last %}  
 </div>  
 <div class="carousel-item">  
 {% endif %}  
  
 {% endfor %}  
 </div>  
 </div>  
 </div>  
 <a class="carousel-control-prev" href="#demo{{forloop.counter}}" role="button" data-bs-slide="prev">  
 <span class="carousel-control-prev-icon" aria-hidden="true"></span>  
 <span class="sr-only">Previous</span>  
 </a>  
 <a class="carousel-control-next" href="#demo{{forloop.counter}}" role="button" data-bs-slide="next">  
 <span class="carousel-control-next-icon" aria-hidden="true"></span>  
 <span class="sr-only">Next</span>  
 </a>  
 {% endfor %}  
</div>  
{% endblock %}

****

**Category wise product display(as we see shoes two times, so it must be separated by category)**

def index(request):  
 # fetches products from database  
 products = Product.objects.all()  
 print(products)  
 n=len(products)  
 nSlides = n//4 + ceil((n/4) - (n//4))  
 # params = {'product' : products, 'no\_of\_slides':nSlides, 'range' : range(1, nSlides)}  
 #  
 # allProds = [[products, range(1, nSlides), nSlides],  
 # [products, range(1, nSlides), nSlides]]  
  
 allProds = []  
 catprods = Product.objects.values('category','id')  
 cats = {item['category'] for item in catprods}  
 for cat in cats:  
 prod = Product.objects.filter(category = cat)  
 n = len(prod)  
 nSlides = n//4 + ceil((n/4) - (n//4))  
 allProds.append([prod, range(1, nSlides), nSlides])  
  
 params = {'allProds' : allProds}  
 return render(request, "shop/index.html", params)

**Index.html change**

.col-md-3 img {  
width: 227px;  
max-height: 242px;  
}

<h4 class="my-4">{{product.0.category}} - Recommended Items For You</h4>

**now we cn see categories displayed correctly**

**Creating Online Shopping Cart Using Javascript**

****

****

**We create local storage for website. And session storage works until website session ens**

**In basic.html**

****

**Index.html**

{%block js%}  
<script>  
 console.log("working");  
</script>  
{% endblock %}

****

**Index.html**

{%block js%}  
<script>  
 console.log("working");  
 if(localStorage.getItem('cart') == null){  
 var cart = {};  
 }  
  
 else  
 {  
 cart = JSON.parse(localStorage.getItem('cart'));  
 }  
<!--wherever class is .cart call this function-->  
 $('.cart').click(function(){  
 console.log('clicked”);  
 });  
</script>  
{% endblock %}

{%block js%}  
<script>  
 console.log("working");  
 if(localStorage.getItem('cart') == null){  
 var cart = {};  
 }  
  
 else  
 {  
 cart = JSON.parse(localStorage.getItem('cart'));  
 }  
<!--wherever class is .cart call this function-->  
 $('.cart').click(function(){  
  
 console.log('clicked');  
 var idstr = this.id.toString();  
 console.log(idstr);  
  
 if(cart[idstr] != undefined){  
 cart[idstr] = cart[idstr] + 1;  
 }  
 else  
 {  
 cart[idstr]=1;  
 }  
 console.log(cart);  
 localStorage.setItem('cart',JSON.stringify(cart));  
 });  
</script>  
{% endblock %}

**Also change anchor tag to button and add class and id as ,**

<div class="container carousel-inner no-padding">  
 <div class="carousel-item active">  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="/media/{{product.0.image}}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">{{product.0.product\_name}}</h5>  
 <p class="card-text">{{product.0.desc}}</p>  
 <button id ="pr{{product.0.id}}" class="btn btn-primary cart">Add to Cart</button>  
 </div>  
 </div>  
 </div>  
  
 {% for i in product|slice:"1:"%}  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="/media/{{i.image}}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">{{i.product\_name}}</h5>  
 <p class="card-text">{{i.desc}}</p>  
 <button id ="pr{{i.id}}" class="btn btn-primary cart">Add to Cart</button>  
 </div>  
 </div>  
 </div>  
 {% if forloop.counter|add:1|divisibleby:4 and forloop.counter > 0 and not forloop.last %}  
 </div>

**Now in inspect even when we refresh we can see cart items added**

****

****

**Imporving Slider Logic & Adding Cart Items Count**

**Basic.html**

**Adding cart button**

<li class="nav-item">  
 <a class="nav-link" href="#">Cart(<span id="cart">0</span>)</a>  
</li>

**Index.html add**

document.getElementById('cart').innerHTML = Object.keys(cart).length;

**in else and at last**

{%block js%}  
<script>  
 console.log("working");  
  
 if(localStorage.getItem('cart') == null){  
 var cart = {};  
 }  
  
 else  
 {  
 cart = JSON.parse(localStorage.getItem('cart'));  
 document.getElementById('cart').innerHTML = Object.keys(cart).length;  
 }  
<!--wherever class is .cart call this function-->  
 $('.cart').click(function(){  
  
 console.log('clicked');  
 var idstr = this.id.toString();  
 console.log(idstr);  
  
 if(cart[idstr] != undefined){  
 cart[idstr] = cart[idstr] + 1;  
 }  
 else  
 {  
 cart[idstr]=1;  
 }  
 console.log(cart);  
 localStorage.setItem('cart',JSON.stringify(cart));  
 document.getElementById('cart').innerHTML = Object.keys(cart).length;  
 });  
</script>  
{% endblock %}

**Also, Adding Bootstrap Row and Some CSS**

**Index.html**

{% extends 'shop/basic.html' %}  
  
{% block css %}  
.col-md-3 {  
display: inline-block;  
margin-left: -4px;  
}  
  
.col-md-3 img {  
width: 170px;  
height: 200px;  
}  
  
.carousel-indicators .active {  
background-color: blue;  
}  
  
body .carousel-indicator li {  
background-color: blue;  
}  
  
body .carousel-indicators {  
bottom: 0;  
}  
  
body .carousel-control-prev-icon,  
body .carousel-control-next-icon {  
background-color: blue;  
}  
  
body .carousel-control-prev,  
body .carousel-control-next {  
top: auto;  
bottom: auto;  
padding-top:222px;  
}  
  
body .no-padding {  
padding-left: 0;  
padding-right: 0;  
}  
{% endblock %}  
  
{% block body %}  
{% load static %}  
<div class="container">  
  
<!-- Slide Show starts here-->  
 {% for product, range, nSlides in allProds %}  
 <h4 class="my-4">{{product.0.category}} - Recommended Items For You</h4>  
 <div class="row">  
 <div id="demo{{forloop.counter}}" class="col carousel slide my-3" data-ride="carousel">  
 <ul class="carousel-indicators">  
 <li data-bs-target="#demo{{forloop.counter}}" data-bs-slide-to="0" class="active"></li>  
  
 {% for i in range %}  
 <li data-bs-target="#demo{{forloop.parentloop.counter}}" data-bs-slide-to="{{i}}"></li>  
 {% endfor %}  
 </ul>  
  
 <div class="container carousel-inner no-padding">  
 <div class="carousel-item active">  
  
 {% for i in product %}  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="/media/{{i.image}}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">{{i.product\_name}}</h5>  
 <p class="card-text">{{i.desc}}</p>  
 <button id ="pr{{i.id}}" class="btn btn-primary cart">Add to Cart</button>  
 </div>  
 </div>  
 </div>  
 {% if forloop.counter|divisibleby:4 and forloop.counter > 0 and not forloop.last %}  
 </div>  
 <div class="carousel-item">  
 {% endif %}  
  
 {% endfor %}  
 </div>  
 </div>  
 </div>  
 <a class="carousel-control-prev" href="#demo{{forloop.counter}}" role="button" data-bs-slide="prev">  
 <span class="carousel-control-prev-icon" aria-hidden="true"></span>  
 <span class="sr-only">Previous</span>  
 </a>  
 <a class="carousel-control-next" href="#demo{{forloop.counter}}" role="button" data-bs-slide="next">  
 <span class="carousel-control-next-icon" aria-hidden="true"></span>  
 <span class="sr-only">Next</span>  
 </a>  
 </div>  
 {% endfor %}  
</div>  
{% endblock %}  
  
  
{%block js%}  
<script>  
 console.log("working");  
  
 if(localStorage.getItem('cart') == null){  
 var cart = {};  
 }  
  
 else  
 {  
 cart = JSON.parse(localStorage.getItem('cart'));  
 document.getElementById('cart').innerHTML = Object.keys(cart).length;  
 }  
<!--wherever class is .cart call this function-->  
 $('.cart').click(function(){  
  
 console.log('clicked');  
 var idstr = this.id.toString();  
 console.log(idstr);  
  
 if(cart[idstr] != undefined){  
 cart[idstr] = cart[idstr] + 1;  
 }  
 else  
 {  
 cart[idstr]=1;  
 }  
 console.log(cart);  
 localStorage.setItem('cart',JSON.stringify(cart));  
 document.getElementById('cart').innerHTML = Object.keys(cart).length;  
 });  
</script>  
{% endblock %}

**Adding Cart PopOver & other pages using bootstrap**

**Index.html**

</script>  
<script type="text/javascript">  
 var popoverTriggerList = [].slice.call(document.querySelectorAll('[data-bs-toggle="popover"]'));  
 var popoverList = popoverTriggerList.map(function (popoverTriggerEl) {  
 return new bootstrap.Popover(popoverTriggerEl);  
 });  
</script>  
{% endblock %}

**Basic.html**

<button type="button" class="btn btn-secondary mx-2" data-bs-toggle="popover" title="Popover title"  
 data-html='true' data-bs-content="Cart for your items in shopping cart">Cart(<span  
 id="cart">0</span>)  
 </button>  
</div>

****

Creating Product Page \\

Index.html

Add Quick view

{% for i in product %}  
<div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="/media/{{i.image}}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">{{i.product\_name}}</h5>  
 <p class="card-text">{{i.desc|slice:"0:23"}}...</p>  
 <button id ="pr{{i.id}}" class="btn btn-primary cart">Add to Cart</button>  
 <a href="/shop/products/{{i.id}}"><button id="qv{{i.id}}" class="btn btn-primary cart">Quick View</button></a>  
 </div>  
 </div>  
</div>

Urls.py

urlpatterns = [  
 path('', views.index, name="ShopHome"),  
 path('about/', views.about, name="AboutUs"),  
 path('contact/', views.contact, name="ContactUs"),  
 path('tracker/', views.tracker, name="TrackingStatus"),  
 path('search/', views.search, name="Search"),  
 path('products/<int:myid>', views.prodView, name="ProductView"),  
 path('checkout/', views.checkout, name="Checkout"),  
]

Views.py

def prodView(request, myid):  
 # fetch the product using id  
 product = Product.objects.filter(id=myid)  
 print(product)  
  
 return render(request, "shop/prodView.html", {'product': product[0]})

prodView.html

{% extends 'shop/basic.html' %}  
  
{%block title%} {{product.product\_name}} - My Awesome Cart {% endblock %}  
  
{%block body%}  
  
<div class="container">  
 <div class="row">  
 <div class="col-md-4">  
 <div class="row">  
 <img src="/media/{{product.image}}" width="233px" height="385px">  
 </div>  
 <div class="row">  
 <button class="btn btn-primary my-3">Buy Now</button>  
 <button class="btn btn-primary my-3">Add to Cart</button>  
 </div>  
 </div>  
 <div class="col-md-8">  
 <h5>{{product.product\_name}}</h5>  
 <p><b>Rs. {{product.price}}</b></p>  
 <p>{{product.desc}}</p>  
 </div>  
 </div>  
</div>  
  
  
{% endblock %}  
{%block js%}  
<script>  
 console.log("working");  
  
 if(localStorage.getItem('cart') == null){  
 var cart = {};  
 }  
  
 else  
 {  
 cart = JSON.parse(localStorage.getItem('cart'));  
 document.getElementById('cart').innerHTML = Object.keys(cart).length;  
 }  
<!--wherever class is .cart call this function-->  
 $('.cart').click(function(){  
  
 console.log('clicked');  
 var idstr = this.id.toString();  
 console.log(idstr);  
  
 if(cart[idstr] != undefined){  
 cart[idstr] = cart[idstr] + 1;  
 }  
 else  
 {  
 cart[idstr]=1;  
 }  
 console.log(cart);  
 localStorage.setItem('cart',JSON.stringify(cart));  
 document.getElementById('cart').innerHTML = Object.keys(cart).length;  
 });  
 $('#popcart').popover('show')  
</script>  
<script type="text/javascript">  
 var popoverTriggerList = [].slice.call(document.querySelectorAll('[data-bs-toggle="popover"]'));  
 var popoverList = popoverTriggerList.map(function (popoverTriggerEl) {  
 return new bootstrap.Popover(popoverTriggerEl);  
 });  
</script>  
{% endblock %}

Index.html added slicer to show description of limited characters on index page

<p class="card-text">{{i.desc|slice:"0:23"}}...</p>





**Creating Contact Page + Database**

**Ccontact.html**

{% extends 'shop/basic.html' %}  
  
{% block title%} Contact Us - My Awesome Cart{% endblock %}  
{% block body %}  
<div class="container my-3">  
 <h3>Contact Us</h3>  
 <form action="/shop/contact/" method="post">{% csrf\_token %}  
 <div class="form-group">  
 <label for="name">Name</label>  
 <input type="text" class="form-control" id="name" name='name' placeholder="Enter Your Name">  
 </div>  
 <div class="form-group">  
 <label for="name">Email</label>  
 <input type="email" class="form-control" id="email" name='email' placeholder="Enter Your Email">  
 </div>  
  
 <div class="form-group">  
 <label for="name">Phone</label>  
 <input type="tel" class="form-control" id="phone" name='phone' placeholder="Enter Your Phone Number">  
 </div>  
  
  
 <div class="form-group">  
 <label for="desc">How May We Help You?</label>  
 <textarea class="form-control" id="desc" name='desc' rows="3"></textarea>  
 </div>  
 <br>  
 <button type="submit" class="btn btn-success">Submit</button>  
 </form>  
  
  
</div>  
{% endblock %}

**Views.py**

****

Views.py add contact method

def contact(request):  
 if request.method == 'POST':  
 name = request.POST.get('name', '')  
 email = request.POST.get('email', '')  
 phone = request.POST.get('phone', '')  
 desc = request.POST.get('desc', '')  
 print(name,email,phone,desc)  
 print(request)  
 return render(request, "shop/contact.html")

models.py

# contact  
class Contact(models.Model):  
 msg\_id = models.AutoField(primary\_key=True)  
 name = models.CharField(max\_length=50)  
 email = models.CharField(max\_length=70, default="")  
 phone = models.CharField(max\_length=70, default="")  
 desc = models.CharField(max\_length=500, default="")  
  
 def \_\_str\_\_(self):  
 return self.name

Python manage.py makemigrations

Python manage.py migrate

Register in admin.py

from django.contrib import admin  
  
# Register your models here.  
from .models import Product, Contact  
  
admin.site.register(Product)  
admin.site.register(Contact)

in admin we can see contacts also



Views.py

def contact(request):  
 if request.method == 'POST':  
 name = request.POST.get('name', '')  
 email = request.POST.get('email', '')  
 phone = request.POST.get('phone', '')  
 desc = request.POST.get('desc', '')  
 contact = Contact(name=name, email=email, phone=phone, desc=desc)  
 contact.save()  
 return render(request, "shop/contact.html")

now when we submit form in conact page

we can see it in admin page also



So we create one more model here!!!

Improving Cart

* localStorage.clear()

Creating Add and Remove Buttons for Items in Cart:

<div class="card align-items-center" style="width: 18rem;">

align to center

**Creatign add and Remove buttons for items in cart**

Index.html

{% extends 'shop/basic.html' %}  
  
{% block css %}  
.col-md-3 {  
display: inline-block;  
margin-left: -4px;  
}  
  
.col-md-3 img {  
width: 170px;  
height: 200px;  
}  
  
.carousel-indicators .active {  
background-color: blue;  
}  
  
body .carousel-indicator li {  
background-color: blue;  
}  
  
body .carousel-indicators {  
bottom: 0;  
}  
  
body .carousel-control-prev-icon,  
body .carousel-control-next-icon {  
background-color: blue;  
}  
  
body .carousel-control-prev,  
body .carousel-control-next {  
top: auto;  
bottom: auto;  
padding-top:222px;  
}  
  
body .no-padding {  
padding-left: 0;  
padding-right: 0;  
}  
{% endblock %}  
  
{% block body %}  
{% load static %}  
<div class="container">  
  
<!-- Slide Show starts here-->  
 {% for product, range, nSlides in allProds %}  
 <h4 class="my-4">{{product.0.category}} - Recommended Items For You</h4>  
 <div class="row">  
 <div id="demo{{forloop.counter}}" class="col carousel slide my-3" data-ride="carousel">  
 <ul class="carousel-indicators">  
 <li data-bs-target="#demo{{forloop.counter}}" data-bs-slide-to="0" class="active"></li>  
  
 {% for i in range %}  
 <li data-bs-target="#demo{{forloop.parentloop.counter}}" data-bs-slide-to="{{i}}"></li>  
 {% endfor %}  
 </ul>  
  
 <div class="container carousel-inner no-padding">  
 <div class="carousel-item active">  
  
 {% for i in product %}  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="/media/{{i.image}}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title">{{i.product\_name}}</h5>  
 <p class="card-text">{{i.desc|slice:"0:53"}}...</p>  
 <span id="divpr{{i.id}}" class="divpr">  
 <button id="pr{{i.id}}" class="btn btn-primary cart">Add To Cart</button>  
 </span>  
 <a href="/shop/products/{{i.id}}"><button id="qv{{i.id}}" class="btn btn-primary cart">QuickView</button></a>  
 </div>  
 </div>  
 </div>  
 {% if forloop.counter|divisibleby:4 and forloop.counter > 0 and not forloop.last %}  
 </div>  
 <div class="carousel-item">  
 {% endif %}  
 {% endfor %}  
 </div>  
 </div>  
 </div>  
 <a class="carousel-control-prev" href="#demo{{forloop.counter}}" role="button" data-bs-slide="prev">  
 <span class="carousel-control-prev-icon" aria-hidden="true"></span>  
 <span class="sr-only">Previous</span>  
 </a>  
 <a class="carousel-control-next" href="#demo{{forloop.counter}}" role="button" data-bs-slide="next">  
 <span class="carousel-control-next-icon" aria-hidden="true"></span>  
 <span class="sr-only">Next</span>  
 </a>  
 </div>  
 {% endfor %}  
</div>  
{% endblock %}  
  
  
{% block js %}  
<script type="text/javascript">  
 console.log("working");  
 var cart = {};  
  
 if(localStorage.getItem('cart') == null){  
 cart = {};  
 }  
 else  
 {  
 cart = JSON.parse(localStorage.getItem('cart'));  
 document.getElementById('cart').innerHTML = Object.keys(cart).length;  
 updateCart(cart);  
 }  
// <!--wherever class is .cart call this function-->  
 $('.cart').click(function() {  
 var idstr = this.id.toString();  
 if (cart[idstr] != undefined) {  
 cart[idstr] = cart[idstr] + 1;  
 } else {  
 cart[idstr] = 1;  
 updateCart(cart);  
 }  
 });  
  
 function updateCart(cart) {  
 for (var item in cart) {  
 document.getElementById('div' + item).innerHTML = "<button id='minus" + item + "' class='btn btn-primary minus'>-</button> <span id='val" + item + "''>" + cart[item] + "</span> <button id='plus" + item + "' class='btn btn-primary plus'> + </button>";  
 }  
 localStorage.setItem('cart', JSON.stringify(cart));  
 document.getElementById('cart').innerHTML = Object.keys(cart).length;  
 console.log(cart);  
 }  
<!-- // If plus or minus button is clicked, change the cart as well as the display value-->  
 $('.divpr').on("click", "button.minus", function() {  
 console.log("Minus Clicked");  
 a = this.id.slice(7, );  
 cart['pr' + a] = cart['pr' + a] - 1;  
 cart['pr' + a] = Math.max(0, cart['pr' + a]);  
 document.getElementById('valpr' + a).innerHTML = cart['pr' + a];  
 updateCart(cart);  
 });  
 $('.divpr').on("click", "button.plus", function() {  
 console.log("Plus Clicked");  
 a = this.id.slice(6, );  
 cart['pr' + a] = cart['pr' + a] + 1;  
 document.getElementById('valpr' + a).innerHTML = cart['pr' + a];  
 updateCart(cart);  
 });  
  
  
  
</script>  
<script type="text/javascript">  
 var popoverTriggerList = [].slice.call(document.querySelectorAll('[data-bs-toggle="popover"]'));  
 var popoverList = popoverTriggerList.map(function (popoverTriggerEl) {  
 return new bootstrap.Popover(popoverTriggerEl);  
 });  
</script>  
{% endblock %}

//here we added logic to plus and minus button

Also added span here

<span id="divpr{{i.id}}" class="divpr">  
 <button id="pr{{i.id}}" class="btn btn-primary cart">Add To Cart</button>  
 </span>



Creating a Cartin Navigation Bar as popover

Bcoz cart at top is not getting updated



Now to update this will create function

In index.html will create updatepopOver

{% block body %}  
{% load static %}  
<div class="container">  
  
<!-- Slide Show starts here-->  
 {% for product, range, nSlides in allProds %}  
 <h4 class="my-4">{{product.0.category}} - Recommended Items For You</h4>  
 <div class="row">  
 <div id="demo{{forloop.counter}}" class="col carousel slide my-3" data-ride="carousel">  
 <ul class="carousel-indicators">  
 <li data-bs-target="#demo{{forloop.counter}}" data-bs-slide-to="0" class="active"></li>  
  
 {% for i in range %}  
 <li data-bs-target="#demo{{forloop.parentloop.counter}}" data-bs-slide-to="{{i}}"></li>  
 {% endfor %}  
 </ul>  
  
 <div class="container carousel-inner no-padding">  
 <div class="carousel-item active">  
  
 {% for i in product %}  
 <div class="col-xs-3 col-sm-3 col-md-3">  
 <div class="card" style="width: 18rem;">  
 <img src="/media/{{i.image}}" class="card-img-top" alt="...">  
 <div class="card-body">  
 <h5 class="card-title" id ="namepr{{i.id}}">{{i.product\_name}}</h5>  
 <p class="card-text">{{i.desc|slice:"0:53"}}...</p>  
 <span id="divpr{{i.id}}" class="divpr">  
 <button id="pr{{i.id}}" class="btn btn-primary cart">Add To Cart</button>  
 </span>  
 <a href="/shop/products/{{i.id}}"><button id="qv{{i.id}}" class="btn btn-primary cart">QuickView</button></a>  
 </div>  
 </div>  
 </div>  
 {% if forloop.counter|divisibleby:4 and forloop.counter > 0 and not forloop.last %}  
 </div>  
 <div class="carousel-item">  
 {% endif %}  
 {% endfor %}  
 </div>  
 </div>  
 </div>  
 <a class="carousel-control-prev" href="#demo{{forloop.counter}}" role="button" data-bs-slide="prev">  
 <span class="carousel-control-prev-icon" aria-hidden="true"></span>  
 <span class="sr-only">Previous</span>  
 </a>  
 <a class="carousel-control-next" href="#demo{{forloop.counter}}" role="button" data-bs-slide="next">  
 <span class="carousel-control-next-icon" aria-hidden="true"></span>  
 <span class="sr-only">Next</span>  
 </a>  
 </div>  
 {% endfor %}  
</div>  
{% endblock %}  
  
  
{% block js %}  
<script type="text/javascript">  
 console.log("working");  
 var cart = {};  
  
 if(localStorage.getItem('cart') == null){  
 cart = {};  
 }  
 else  
 {  
 cart = JSON.parse(localStorage.getItem('cart'));  
 document.getElementById('cart').innerHTML = Object.keys(cart).length;  
 updateCart(cart);  
 }  
// <!--wherever class is .cart call this function-->  
 $('.cart').click(function() {  
 var idstr = this.id.toString();  
 if (cart[idstr] != undefined) {  
 cart[idstr] = cart[idstr] + 1;  
 } else {  
 cart[idstr] = 1;  
 }  
 updateCart(cart);  
 });  
  
<!-- //Add Popover to cart-->  
 updatePopover(cart);  
 function updatePopover(cart) {  
 var popStr = "<div class='row mx-2 my-2'>";  
 var i = 1;  
 for (var item in cart){  
 popStr += i + '. ';  
 popStr += document.getElementById('name' + item).innerHTML.slice(0, 19) + "... Qty: " + cart[item] + '<br>';  
 i++;  
 }  
 popStr = popStr + "</div>";  
  
 console.log(popStr);  
  
 document.getElementById('popcart').setAttribute('data-bs-content', popStr);  
 var popoverTriggerList = [].slice.call(document.querySelectorAll('[data-bs-toggle="popover"]'));  
 var popoverList = popoverTriggerList.map(function (popoverTriggerEl) {  
 return new bootstrap.Popover(popoverTriggerEl, {  
 html: true // Enable HTML content  
 });  
 });  
 }  
  
  
 function updateCart(cart) {  
 for (var item in cart) {  
 document.getElementById('div' + item).innerHTML = "<button id='minus" + item + "' class='btn btn-primary minus'>-</button> <span id='val" + item + "''>" + cart[item] + "</span> <button id='plus" + item + "' class='btn btn-primary plus'> + </button>";  
 }  
 localStorage.setItem('cart', JSON.stringify(cart));  
 document.getElementById('cart').innerHTML = Object.keys(cart).length;  
 console.log(cart);  
 updatePopover(cart);  
 }  
<!-- // If plus or minus button is clicked, change the cart as well as the display value-->  
 $('.divpr').on("click", "button.minus", function() {  
 console.log("Minus Clicked");  
 a = this.id.slice(7, );  
 cart['pr' + a] = cart['pr' + a] - 1;  
 cart['pr' + a] = Math.max(0, cart['pr' + a]);  
 document.getElementById('valpr' + a).innerHTML = cart['pr' + a];  
 updateCart(cart);  
 });  
 $('.divpr').on("click", "button.plus", function() {  
 console.log("Plus Clicked");  
 a = this.id.slice(6, );  
 cart['pr' + a] = cart['pr' + a] + 1;  
 document.getElementById('valpr' + a).innerHTML = cart['pr' + a];  
 updateCart(cart);  
 });  
  
  
  
</script>  
<!--<script type="text/javascript">-->  
<!-- var popoverTriggerList = [].slice.call(document.querySelectorAll('[data-bs-toggle="popover"]'));-->  
<!-- var popoverList = popoverTriggerList.map(function (popoverTriggerEl) {-->  
<!-- return new bootstrap.Popover(popoverTriggerEl);-->  
<!-- });-->  
<!--</script>-->  
{% endblock %}

Button of it in basic.html

<button type="button" class="btn btn-secondary mx-2" data-bs-toggle="popover" title="Cart for your items in my shopping cart"  
 data-html='true' id ='popcart' data-bs-content="Cart for your items in shopping cart">Cart(<span  
 id="cart">0</span>)  
</button>

To display total cart items on button, as its not working correctly



function updateCart(cart) {  
var sum=0;  
for (var item in cart) {  
 sum= sum+cart[item];  
 document.getElementById('div' + item).innerHTML = "<button id='minus" + item + "' class='btn btn-primary minus'>-</button> <span id='val" + item + "''>" + cart[item] + "</span> <button id='plus" + item + "' class='btn btn-primary plus'> + </button>";  
}  
localStorage.setItem('cart', JSON.stringify(cart));  
document.getElementById('cart').innerHTML = sum;  
console.log(cart);  
updatePopover(cart);  
}

will add sum inupdate\_cart method



Will Create Clear Cart button and checkout button

<!--// If the add to cart button is clicked, add/increment the item-->  
<!--// $('.cart').click(function() {-->  
 $('.divpr').on("click", "button.cart", function() {  
 var idstr = this.id.toString();  
 if (cart[idstr] != undefined) {  
 cart[idstr] = cart[idstr] + 1;  
 } else {  
 cart[idstr] = 1;  
 }  
 updateCart(cart);  
 });  
  
  
 var myDefaultAllowList = bootstrap.Tooltip.Default.allowList  
 myDefaultAllowList.button = ['id', 'class' , 'type' , 'onclick']  
<!-- //Add Popover to cart-->  
 updatePopover(cart);

//add this before update popover method so that will display buttons in popover

var myDefaultAllowList = bootstrap.Tooltip.Default.allowList  
myDefaultAllowList.button = ['id', 'class' , 'type' , 'onclick']

add this in button

popStr = popStr + "</div> <a href='/shop/checkout'><button class='btn btn-primary' id ='checkout'>Checkout</button></a> <button class='btn btn-primary' onclick='clearCart()' id ='clearCart'>Clear Cart</button> "

updatePopover(cart);  
function updatePopover(cart) {  
 var popStr = "<div class='row mx-2 my-2'>";  
 var i = 1;  
 for (var item in cart){  
 popStr += i + '. ';  
 popStr += document.getElementById('name' + item).innerHTML.slice(0, 19) + "... Qty: " + cart[item] + '<br>';  
 i++;  
 }  
 popStr = popStr + "</div> <a href='/shop/checkout'><button class='btn btn-primary' id ='checkout'>Checkout</button></a> <button class='btn btn-primary' onclick='clearCart()' id ='clearCart'>Clear Cart</button> "  
  
 console.log(popStr);  
  
 document.getElementById('popcart').setAttribute('data-bs-content', popStr);  
 var popoverTriggerList = [].slice.call(document.querySelectorAll('[data-bs-toggle="popover"]'));  
 var popoverList = popoverTriggerList.map(function (popoverTriggerEl) {  
 return new bootstrap.Popover(popoverTriggerEl, {  
 html: true // Enable HTML content  
 });  
 });  
}

function clearCart() {  
 cart = JSON.parse(localStorage.getItem('cart'));  
 for (var item in cart) {  
 document.getElementById('div' + item).innerHTML = '<button id="' + item + '" class="btn btn-primary cart">Add To Cart</button>'  
 }  
 localStorage.clear();  
 cart = {};  
 console.log("In CLearCart method");  
 updateCart(cart);  
}

 when clicked on clear cart it will clear

Creating Checkout Page

{% extends 'shop/basic.html' %}  
{% block title%} Checkout - My Awesome Cart{% endblock %}  
{% block body %}  
<div class="container">  
 <div class="col my-4">  
 <h2>Step 1 - My Awesome Cart Express Checkout - Review Your Cart Items</h2>  
 <div id="items">  
 <ol class="list-group list-group-numbered">  
 <li class="list-group-item d-flex justify-content-between align-items-start">  
 <div class="ms-2 me-auto">  
 <div class="fw-bold">Subheading</div>  
 Content for list item  
 </div>  
 <span class="badge bg-primary rounded-pill">14</span>  
 </li>  
 <li class="list-group-item d-flex justify-content-between align-items-start">  
 <div class="ms-2 me-auto">  
 <div class="fw-bold">Subheading</div>  
 Content for list item  
 </div>  
 <span class="badge bg-primary rounded-pill">14</span>  
 </li>  
 <li class="list-group-item d-flex justify-content-between align-items-start">  
 <div class="ms-2 me-auto">  
 <div class="fw-bold">Subheading</div>  
 Content for list item  
 </div>  
 <span class="badge bg-primary rounded-pill">14</span>  
 </li>  
 </ol>  
 </div>  
 </div>  
 <div class="col my-4">  
 <h2>Step 2 - Enter Address & Other Details:</h2>  
 <form class="row g-3">  
 <div class="col-md-6">  
 <label for="inputname" class="form-label">Name</label>  
 <input type="name" class="form-control" id="inputname" placeholder="Name">  
 </div>  
 <div class="col-md-6">  
 <label for="inputEmail4" class="form-label">Email</label>  
 <input type="email" class="form-control" id="inputEmail4" placeholder="Email">  
 </div>  
 <div class="col-12">  
 <label for="inputAddress" class="form-label">Address</label>  
 <input type="text" class="form-control" id="inputAddress" placeholder="1234 Main St">  
 </div>  
 <div class="col-12">  
 <label for="inputAddress2" class="form-label">Address Line 2</label>  
 <input type="text" class="form-control" id="inputAddress2" placeholder="Apartment, studio, or floor">  
 </div>  
 <div class="col-md-6">  
 <label for="inputCity" class="form-label">City</label>  
 <input type="text" class="form-control" id="inputCity" placeholder="Enter City">  
 </div>  
 <div class="col-md-4">  
 <label for="inputState" class="form-label">State</label>  
 <input type="text" class="form-control" id="inputState" placeholder="Enter State">  
 </div>  
 <div class="col-md-2">  
 <label for="inputZip" class="form-label">Zip</label>  
 <input type="text" class="form-control" id="inputZip" placeholder="Enter Zip Code">  
 </div>  
 <div class="col-md-2">  
 <label for="inputPhone" class="form-label">Phone No.</label>  
 <input type="text" class="form-control" id="inputPhone" placeholder="Enter Phone">  
 </div>  
 <div class="col-12">  
 <button type="submit" class="btn btn-primary">Place Order</button>  
 </div>  
 </form>  
 </div>  
</div>  
{% endblock %}



Adding Items on checkout Page

Checkout.html

{% extends 'shop/basic.html' %}  
{% block title%} Checkout - My Awesome Cart{% endblock %}  
{% block body %}  
<div class="container">  
 <div class="col my-4">  
 <h2>Step 1 - My Awesome Cart Express Checkout - Review Your Cart Items</h2>  
 <div>  
 <ul class="list-group list-group-numbered" id="items">  
 </ul>  
 </div>  
 </div>  
 <div class="col my-4">  
 <h2>Step 2 - Enter Address & Other Details:</h2>  
 <form class="row g-3">  
 <div class="col-md-6">  
 <label for="inputname" class="form-label">Name</label>  
 <input type="name" class="form-control" id="inputname" placeholder="Name">  
 </div>  
 <div class="col-md-6">  
 <label for="inputEmail4" class="form-label">Email</label>  
 <input type="email" class="form-control" id="inputEmail4" placeholder="Email">  
 </div>  
 <div class="col-12">  
 <label for="inputAddress" class="form-label">Address</label>  
 <input type="text" class="form-control" id="inputAddress" placeholder="1234 Main St">  
 </div>  
 <div class="col-12">  
 <label for="inputAddress2" class="form-label">Address Line 2</label>  
 <input type="text" class="form-control" id="inputAddress2" placeholder="Apartment, studio, or floor">  
 </div>  
 <div class="col-md-6">  
 <label for="inputCity" class="form-label">City</label>  
 <input type="text" class="form-control" id="inputCity" placeholder="Enter City">  
 </div>  
 <div class="col-md-4">  
 <label for="inputState" class="form-label">State</label>  
 <input type="text" class="form-control" id="inputState" placeholder="Enter State">  
 </div>  
 <div class="col-md-2">  
 <label for="inputZip" class="form-label">Zip</label>  
 <input type="text" class="form-control" id="inputZip" placeholder="Enter Zip Code">  
 </div>  
 <div class="col-md-2">  
 <label for="inputPhone" class="form-label">Phone No.</label>  
 <input type="text" class="form-control" id="inputPhone" placeholder="Enter Phone">  
 </div>  
 <div class="col-12">  
 <button type="submit" class="btn btn-primary">Place Order</button>  
 </div>  
 </form>  
 </div>  
</div>  
{% endblock %}  
  
{% block js %}  
<script type="text/javascript">  
 console.log("working");  
 var cart = {};  
  
 if(localStorage.getItem('cart') == null){  
 cart = {};  
 }  
 else  
 {  
 cart = JSON.parse(localStorage.getItem('cart'));  
 }  
  
 var sum=0;  
 if ($.isEmptyObject(cart)){  
<!-- if object is empty-->  
 mystr = `<p>Your Cart is empty, please add some items in cart before checking out!!</p>`  
 $('#items').append(mystr);  
 }  
 else{  
 for (item in cart) {  
 let name = cart[item][1];  
 let qty = cart[item][0];  
 sum = sum+qty;  
 mystr = ` <li class="list-group-item d-flex justify-content-between align-items-start">  
 <div class="ms-2 me-auto">  
 <div class="fw-bold">${name}</div>  
 </div>  
 <span class="badge bg-primary rounded-pill">${qty}</span>  
 </li>`  
 $('#items').append(mystr);  
 }  
 }  
  
 document.getElementById('cart').innerHTML = sum;  
  
</script>  
{% endblock %}

Js of indes.html

{% block js %}  
<script type="text/javascript">  
 console.log("working");  
 var cart = {};  
  
 if(localStorage.getItem('cart') == null){  
 cart = {};  
 }  
 else  
 {  
 cart = JSON.parse(localStorage.getItem('cart'));  
<!-- document.getElementById('cart').innerHTML = Object.keys(cart).length;-->  
 updateCart(cart);  
 }  
// <!--wherever class is .cart call this function-->  
<!-- $('.cart').click(function() {-->  
<!--// If the add to cart button is clicked, add/increment the item-->  
<!--// $('.cart').click(function() {-->  
 // If the add to cart button is clicked, add/increment the item  
//$('.cart').click(function() {  
 $('.divpr').on('click', 'button.cart', function() {  
 var idstr = this.id.toString();  
 if (cart[idstr] != undefined) {  
 qty = cart[idstr][0] + 1;  
 } else {  
 qty = 1;  
 name = document.getElementById('name'+idstr).innerHTML  
 cart[idstr] = [qty, name];  
 }  
 updateCart(cart);  
 });  
  
  
 var myDefaultAllowList = bootstrap.Tooltip.Default.allowList  
 myDefaultAllowList.button = ['id', 'class' , 'type' , 'onclick']  
<!-- //Add Popover to cart-->  
 updatePopover(cart);  
 function updatePopover(cart) {  
 var popStr = "<div class='row mx-2 my-2'>";  
 var i = 1;  
 for (var item in cart){  
 popStr += i + '. ';  
 popStr += document.getElementById('name' + item).innerHTML.slice(0, 19) + "... Qty: " + cart[item][0] + '<br>';  
 i++;  
 }  
 popStr = popStr + "</div> <a href='/shop/checkout'><button class='btn btn-primary' id ='checkout'>Checkout</button></a> <button class='btn btn-primary' onclick='clearCart()' id ='clearCart'>Clear Cart</button> "  
  
 console.log(popStr);  
  
 document.getElementById('popcart').setAttribute('data-bs-content', popStr);  
 var popoverTriggerList = [].slice.call(document.querySelectorAll('[data-bs-toggle="popover"]'));  
 var popoverList = popoverTriggerList.map(function (popoverTriggerEl) {  
 return new bootstrap.Popover(popoverTriggerEl, {  
 html: true // Enable HTML content  
 });  
 });  
 }  
  
 function clearCart() {  
 cart = JSON.parse(localStorage.getItem('cart'));  
 for (var item in cart) {  
 document.getElementById('div' + item).innerHTML = '<button id="' + item + '" class="btn btn-primary cart">Add To Cart</button>'  
 }  
 localStorage.clear();  
 cart = {};  
 console.log("In CLearCart method");  
 updateCart(cart);  
 }  
  
  
 function updateCart(cart) {  
 var sum=0;  
 for (var item in cart) {  
 sum= sum+cart[item][0];  
 document.getElementById('div' + item).innerHTML = "<button id='minus" + item + "' class='btn btn-primary minus'>-</button> <span id='val" + item + "''>" + cart[item][0] + "</span> <button id='plus" + item + "' class='btn btn-primary plus'> + </button>";  
 }  
 localStorage.setItem('cart', JSON.stringify(cart));  
 document.getElementById('cart').innerHTML = sum;  
 console.log(cart);  
 updatePopover(cart);  
 }  
<!-- // If plus or minus button is clicked, change the cart as well as the display value-->  
 $('.divpr').on("click", "button.minus", function() {  
 console.log("Minus Clicked");  
 a = this.id.slice(7, );  
 cart['pr' + a][0] = cart['pr' + a][0] - 1;  
 cart['pr' + a][0] = Math.max(0, cart['pr' + a][0]);  
 document.getElementById('valpr' + a).innerHTML = cart['pr' + a][0];  
 updateCart(cart);  
 });  
 $('.divpr').on("click", "button.plus", function() {  
 console.log("Plus Clicked");  
 a = this.id.slice(6, );  
 cart['pr' + a][0] = cart['pr' + a][0] + 1;  
 document.getElementById('valpr' + a).innerHTML = cart['pr' + a][0];  
 updateCart(cart);  
 });  
  
  
  
</script>  
<!--<script type="text/javascript">-->  
<!-- var popoverTriggerList = [].slice.call(document.querySelectorAll('[data-bs-toggle="popover"]'));-->  
<!-- var popoverList = popoverTriggerList.map(function (popoverTriggerEl) {-->  
<!-- return new bootstrap.Popover(popoverTriggerEl);-->  
<!-- });-->  
<!--</script>-->  
{% endblock %}

**Creating Database Entry For Orders**

Models.py

class Orders(models.Model):  
 order\_id = models.AutoField(primary\_key=True)  
 items\_json = models.CharField(max\_length=5000)  
 name = models.CharField(max\_length=90)  
 email = models.CharField(max\_length=111)  
 address = models.CharField(max\_length=111)  
 city = models.CharField(max\_length=111)  
 state = models.CharField(max\_length=111)  
 zip\_code = models.CharField(max\_length=111)

phone=models.CharField(max\_length=111)

phone=models.CharField(max\_length=111, default="")

admin.py

register in admin

# Register your models here.  
from .models import Product, Contact, Orders  
  
admin.site.register(Product)  
admin.site.register(Contact)  
admin.site.register(Orders)

Now migrate

Python manage.py makemigrations

Python manage.py migrate

Checkout.html

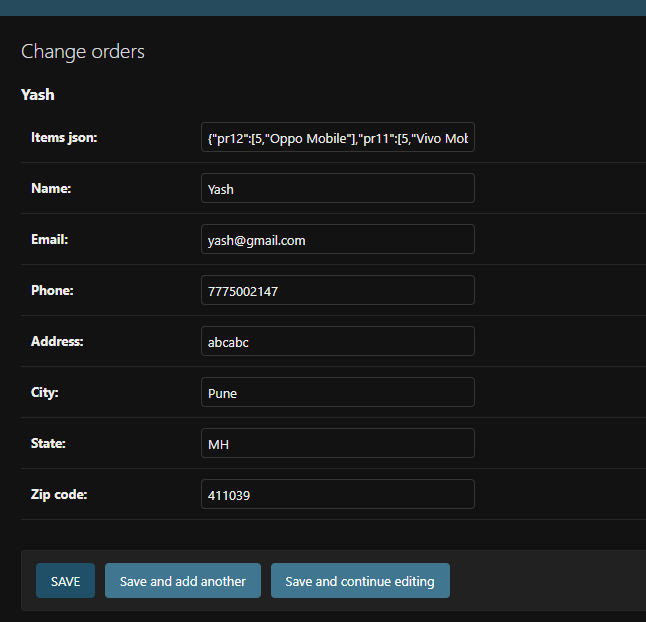
Add pop up to display form submitted using javascript and change ids also

{% extends 'shop/basic.html' %}  
{% block title%} Checkout - My Awesome Cart{% endblock %}  
{% block body %}  
<div class="container">  
 <div class="col my-4">  
 <h2>Step 1 - My Awesome Cart Express Checkout - Review Your Cart Items</h2>  
 <div>  
 <ul class="list-group list-group-numbered" id="items">  
 </ul>  
 </div>  
 </div>  
 <div class="col my-4">  
 <h2>Step 2 - Enter Address & Other Details:</h2>  
 <form method="post" action='/shop/checkout/' class="row g-3">{% csrf\_token %}  
 <input type="hidden" name="itemsJson" id="itemsJson">  
 <div class="col-md-6">  
 <label for="inputname" class="form-label">Name</label>  
 <input type="text" class="form-control" id="name" name="name" placeholder="Name">  
 </div>  
 <div class="col-md-6">  
 <label for="inputEmail4" class="form-label">Email</label>  
 <input type="email" class="form-control" id="email" name="email" placeholder="Email">  
 </div>  
 <div class="col-12">  
 <label for="inputAddress" class="form-label">Address</label>  
 <input type="text" class="form-control" id="address1" name="address1" placeholder="1234 Main St">  
 </div>  
 <div class="col-12">  
 <label for="inputAddress2" class="form-label">Address Line 2</label>  
 <input type="text" class="form-control" id="address2" name="address2" placeholder="Apartment, studio, or floor">  
 </div>  
 <div class="col-md-6">  
 <label for="inputCity" class="form-label">City</label>  
 <input type="text" class="form-control" id="city" name="city" placeholder="Enter City">  
 </div>  
 <div class="col-md-4">  
 <label for="inputState" class="form-label">State</label>  
 <input type="text" class="form-control" id="state" name="state" placeholder="Enter State">  
 </div>  
 <div class="col-md-2">  
 <label for="inputZip" class="form-label">Zip</label>  
 <input type="text" class="form-control" id="zip\_code" name="zip\_code" placeholder="Enter Zip Code">  
 </div>  
 <div class="col-md-2">  
 <label for="inputPhone" class="form-label">Phone No.</label>  
 <input type="text" class="form-control" id="phone" name="phone" placeholder="Enter Phone">  
 </div>  
 <div class="col-12">  
 <button type="submit" class="btn btn-primary">Place Order</button>  
 </div>  
 </form>  
 </div>  
</div>  
{% endblock %}  
  
{% block js %}  
<script type="text/javascript">  
 console.log("working");  
 var cart = {};  
  
 if(localStorage.getItem('cart') == null){  
 cart = {};  
 }  
 else  
 {  
 cart = JSON.parse(localStorage.getItem('cart'));  
 }  
  
 var sum=0;  
 if ($.isEmptyObject(cart)){  
<!-- if object is empty-->  
 mystr = `<p>Your Cart is empty, please add some items in cart before checking out!!</p>`  
 $('#items').append(mystr);  
 }  
 else{  
 for (item in cart) {  
 let name = cart[item][1];  
 let qty = cart[item][0];  
 sum = sum+qty;  
 mystr = ` <li class="list-group-item d-flex justify-content-between align-items-start">  
 <div class="ms-2 me-auto">  
 <div class="fw-bold">${name}</div>  
 </div>  
 <span class="badge bg-primary rounded-pill">${qty}</span>  
 </li>`  
 $('#items').append(mystr);  
 }  
 }  
  
 document.getElementById('cart').innerHTML = sum;  
 $('#itemsJson').val(JSON.stringify(cart));  
 {% if thank %}  
 alert("Thanks for ordering with us. Your order id is {{id}}. Use it to track your order using our order tracker")  
 localStorage.clear();  
 document.location="/shop";  
 {%endif%}  
</script>  
{% endblock %}

Views.py

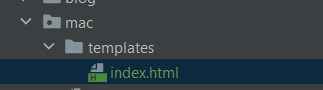
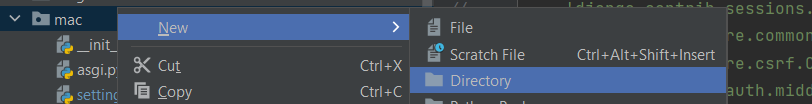
def checkout(request):  
 if request.method == 'POST':  
 items\_json = request.POST.get('itemsJson', '')  
 name = request.POST.get('name', '')  
 email = request.POST.get('email', '')  
 address = request.POST.get('address1', '') + request.POST.get('address2', '')  
 city = request.POST.get('city', '')  
 state = request.POST.get('state', '')  
 zip\_code = request.POST.get('zip\_code', '')  
 phone = request.POST.get('phone', '')  
  
 order = Orders(name=name, email=email, address=address, city=city, state=state, zip\_code=zip\_code, phone=phone, items\_json=items\_json)  
 order.save()  
 thank = True  
 id = order.order\_id  
 return render(request, "shop/checkout.html", {'thank':thank, 'id':id})  
 return render(request, "shop/checkout.html")

//now whe we checkout we can see all details in admin panel with order details in json format

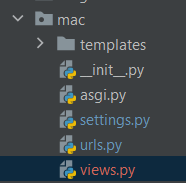


Will add mac/templates folder path in settings.py in main project

TEMPLATES = [  
 {  
 'BACKEND': 'django.template.backends.django.DjangoTemplates',  
 'DIRS': ['mac/templates'],



Views.py create it

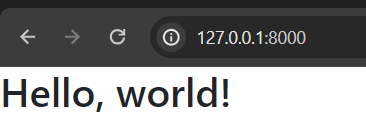


Urls.py add ‘’ to views.index

urlpatterns = [  
 path('admin/', admin.site.urls),  
 path('shop/', include("shop.urls")),  
 path('blog/', include("blog.urls")),  
 path('', views.index),  
] + static(settings.MEDIA\_URL, document\_root=settings.MEDIA\_ROOT)

views.py

from django.shortcuts import render  
from django.http import HttpResponse  
  
# Create your views here.  
def index(request):  
 return render(request, "index.html")



In Models.py of shop app

# to track the order we wrote below class  
class OrderUpdate(models.Model):  
 update\_id= models.AutoField(primary\_key=True)  
 order\_id= models.IntegerField(default="")  
 update\_desc= models.CharField(max\_length=5000)  
 timestamp= models.DateField(auto\_now\_add= True)  
  
 def \_\_str\_\_(self):  
 return self.update\_desc[0:7] + "..."

Add this

And alos register in admin.py

from django.contrib import admin  
  
# Register your models here.  
from .models import Product, Contact, Orders, OrderUpdate  
  
admin.site.register(Product)  
admin.site.register(Contact)  
admin.site.register(Orders)  
admin.site.register(OrderUpdate)

now make mogrations and migrate

in views.py of shop app

shop/views.py

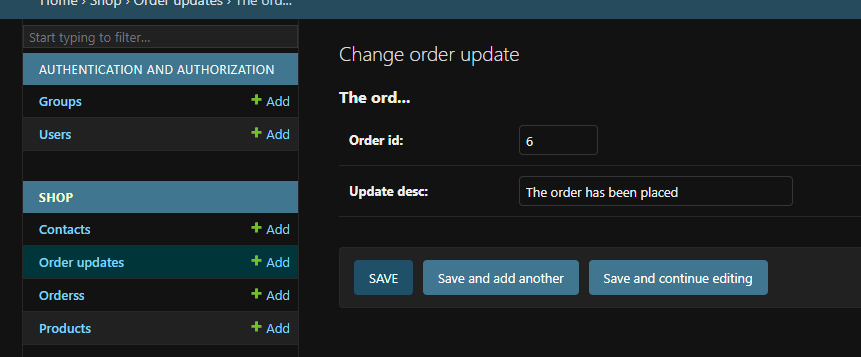
from .models import Product, Contact, Orders, OrderUpdate

add this in checkout method

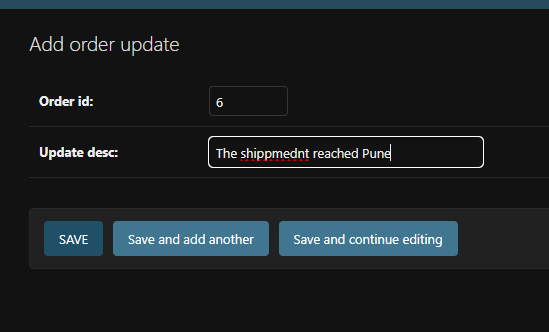
update = OrderUpdate(order\_id = order.order\_id, update\_desc="The order has been placed")  
update.save()

def checkout(request):  
 if request.method == 'POST':  
 items\_json = request.POST.get('itemsJson', '')  
 name = request.POST.get('name', '')  
 email = request.POST.get('email', '')  
 address = request.POST.get('address1', '') + request.POST.get('address2', '')  
 city = request.POST.get('city', '')  
 state = request.POST.get('state', '')  
 zip\_code = request.POST.get('zip\_code', '')  
 phone = request.POST.get('phone', '')  
  
 order = Orders(name=name, email=email, address=address, city=city, state=state, zip\_code=zip\_code, phone=phone, items\_json=items\_json)  
 order.save()  
 update = OrderUpdate(order\_id = order.order\_id, update\_desc="The order has been placed")  
 update.save()  
 thank = True  
 id = order.order\_id  
 return render(request, "shop/checkout.html", {'thank':thank, 'id':id})  
 return render(request, "shop/checkout.html")

//now when we checkout and see admin login we can see



Now add orderUpdate new

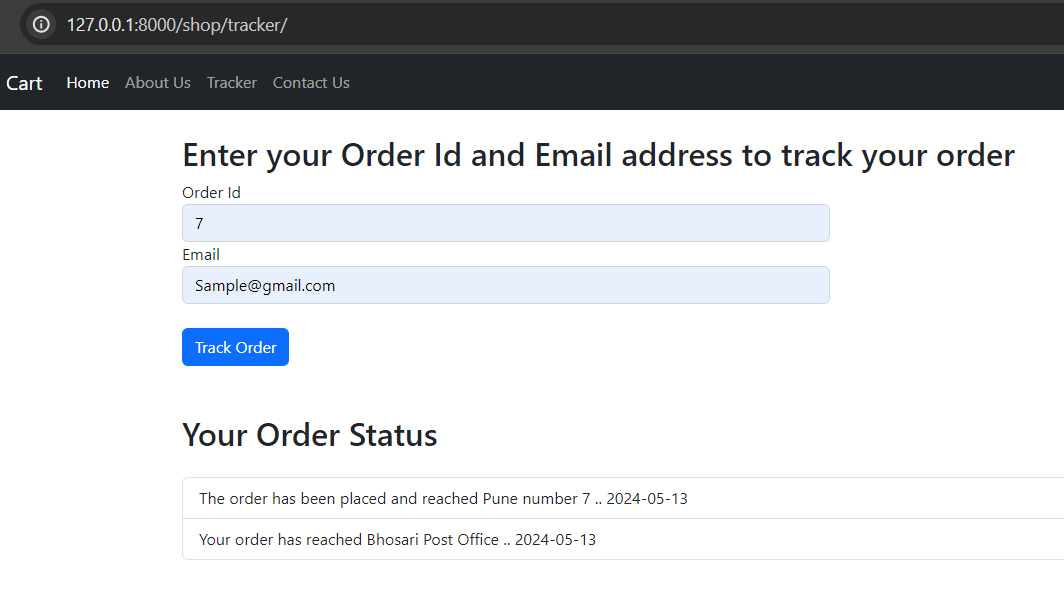


Shops/Views.py add

def tracker(request):  
 if request.method=="POST":  
 orderId = request.POST.get('orderId', '')  
 email = request.POST.get('email', '')  
 try:  
 order = Orders.objects.filter(order\_id=orderId, email=email)  
 if len(order)>0:  
 update = OrderUpdate.objects.filter(order\_id=orderId)  
 updates = []  
 for item in update:  
 updates.append({'text': item.update\_desc, 'time': item.timestamp})  
 response = json.dumps(updates, default=str)  
 return HttpResponse(response)  
 else:  
 return HttpResponse('{}')  
 except Exception as e:  
 return HttpResponse('{}')  
  
 return render(request, 'shop/tracker.html')

Tracker.html

{% extends 'shop/basic.html' %}  
{% block title%} My Awesome Cart Tracker{% endblock %}  
{% block body %}  
<div class="container">  
 <div class="col my-4">  
 <h2>Enter your Order Id and Email address to track your order</h2>  
 <form method="post" action="#" id="trackerForm">{% csrf\_token %}  
 <div class="form-row">  
 <div class="form-group col-md-6">  
 <label for="inputname">Order Id</label>  
 <input type="text" class="form-control" id="orderId" name="orderId" placeholder="Order Id">  
 </div>  
 <div class="form-group col-md-6">  
 <label for="inputEmail4">Email</label>  
 <input type="email" class="form-control" id="email" name="email" placeholder="Email">  
 </div>  
 <button type="submit" class="btn btn-primary my-4">Track Order</button>  
 </div>  
 </div>  
 <div class="col my-4">  
 <h2>Your Order Status</h2>  
 <div class="my-4">  
 <ul class="list-group" id="items">  
 Enter your order Id and Email and click Track Order to find details about your order!  
 </ul>  
 </div>  
 </div>  
</div>  
{% endblock %}  
{% block js %}  
<script type="text/javascript">  
$('#trackerForm').submit(function(event) {  
 $('#items').empty();  
 var formData = {  
 'orderId': $('input[name=orderId]').val(),  
 'email': $('input[name=email]').val(),  
 'csrfmiddlewaretoken': $('input[name=csrfmiddlewaretoken]').val()  
 };  
 $.ajax({  
 type: 'POST',  
 url: '/shop/tracker/',  
 data: formData,  
 encode: true  
 })  
 .done(function(data) {  
 console.log(data)  
 updates = JSON.parse(data);  
 if (updates.length > 0 & updates != {}) {  
 for (i = 0; i < updates.length; i++) {  
 let text = updates[i]['text'];  
 let time = updates[i]['time'];  
 mystr = `<li class="list-group-item d-flex justify-content-between align-items-center">  
 ${text} .. ${time}  
 <span class="badge badge-primary badge-pill">${time}</span>  
 </li>`  
 $('#items').append(mystr);  
 }  
 } else {  
 mystr = `<li class="list-group-item d-flex justify-content-between align-items-center">  
 Sorry, We are not able to fetch this order id and email. Make sure to type correct order Id and email</li>`  
 $('#items').append(mystr);  
 }  
 });  
 event.preventDefault();  
});  
</script>  
{% endblock %}



**Displaying Order Items in Tracker**

Tracker .html

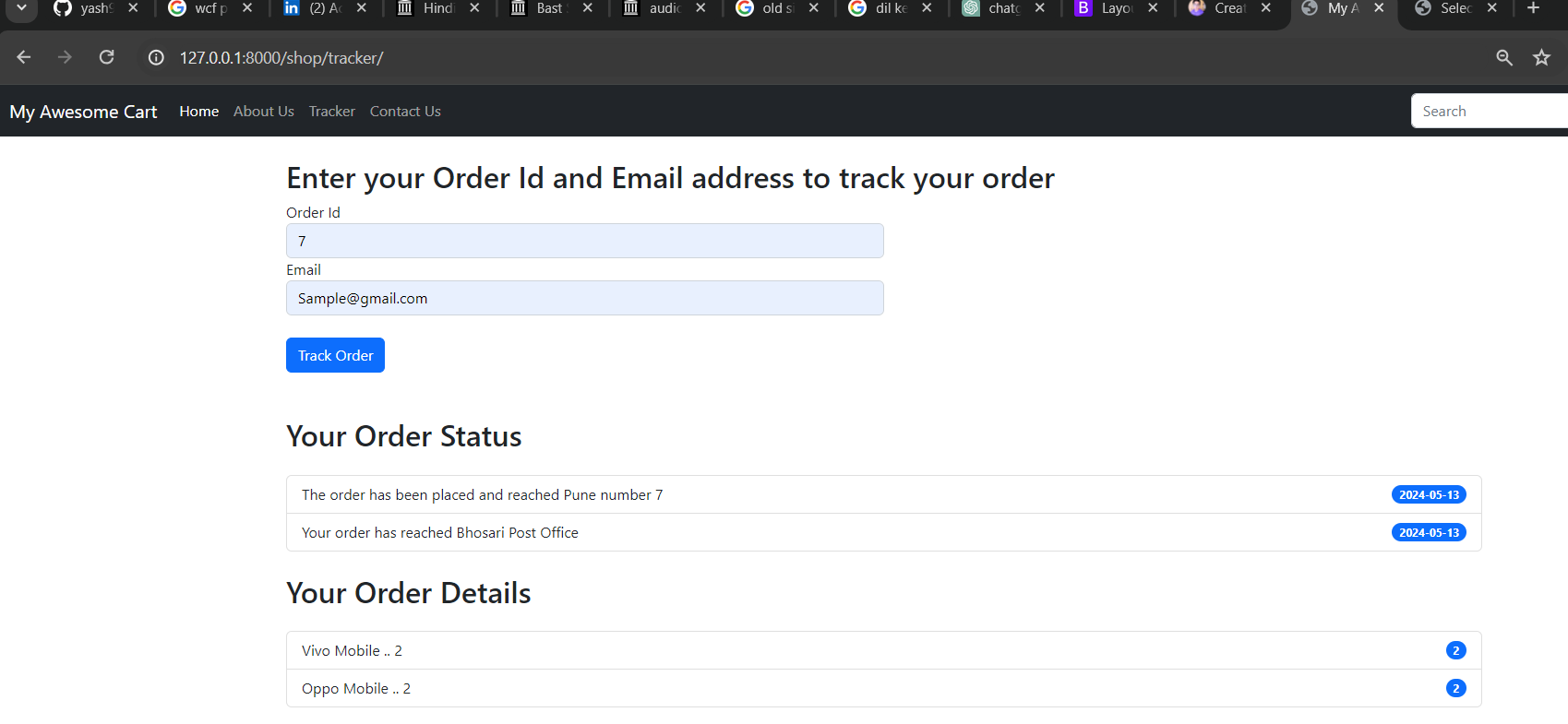
{% extends 'shop/basic.html' %}  
{% block title%} My Awesome Cart Tracker{% endblock %}  
{% block body %}  
<div class="container">  
 <div class="col my-4">  
 <h2>Enter your Order Id and Email address to track your order</h2>  
 <form method="post" action="#" id="trackerForm">{% csrf\_token %}  
 <div class="form-row">  
 <div class="form-group col-md-6">  
 <label for="inputname">Order Id</label>  
 <input type="text" class="form-control" id="orderId" name="orderId" placeholder="Order Id">  
 </div>  
 <div class="form-group col-md-6">  
 <label for="inputEmail4">Email</label>  
 <input type="email" class="form-control" id="email" name="email" placeholder="Email">  
 </div>  
 <button type="submit" class="btn btn-primary my-4">Track Order</button>  
 </div>  
 </div>  
 <div class="col my-4">  
 <h2>Your Order Status</h2>  
 <div class="my-4">  
 <ul class="list-group" id="items">  
 Enter your order Id and Email and click Track Order to find details about your order!  
 </ul>  
 </div>  
 <h2>Your Order Details</h2>  
 <div class="my-4">  
 <ul class="list-group" id="citems">  
 </ul>  
 </div>  
 </div>  
</div>  
{% endblock %}  
{% block js %}  
<script type="text/javascript">  
$('#trackerForm').submit(function(event) {  
 $('#items').empty();  
 var formData = {  
 'orderId': $('input[name=orderId]').val(),  
 'email': $('input[name=email]').val(),  
 'csrfmiddlewaretoken': $('input[name=csrfmiddlewaretoken]').val()  
 };  
 $.ajax({  
 type: 'POST',  
 url: '/shop/tracker/',  
 data: formData,  
 encode: true  
 })  
 .done(function(data) {  
 console.log(data)  
 data = JSON.parse(data)  
 updates = data[0];  
 if (updates.length > 0 & updates != {}) {  
 for (i = 0; i < updates.length; i++) {  
 let text = updates[i]['text'];  
 let time = updates[i]['time'];  
 mystr = `<li class="list-group-item d-flex justify-content-between align-items-center">  
 ${text}  
 <span class="badge bg-primary rounded-pill">${time}</span>  
 </li>`  
 $('#items').append(mystr);  
 }  
 } else {  
 mystr = `<li class="list-group-item d-flex justify-content-between align-items-center">  
 Sorry, We are not able to fetch this order id and email. Make sure to type correct order Id and email</li>`  
 $('#items').append(mystr);  
 }  
<!-- fill in the order details-->  
 cart=JSON.parse(data[1]);  
 console.log(cart);  
  
 for (item in cart) {  
  
 let name = cart[item][1];  
 let qty = cart[item][0];  
 mystr = ` <li class="list-group-item d-flex justify-content-between align-items-center">  
 ${name} .. ${qty}  
 <span class="badge bg-primary rounded-pill">${qty}</span>  
 </li>`  
 $('#citems').append(mystr);  
 }  
  
 });  
 event.preventDefault();  
});  
</script>  
{% endblock %}

Add

response = json.dumps([updates, order[0].items\_json], default=str)

in views.py

def tracker(request):  
 if request.method=="POST":  
 orderId = request.POST.get('orderId', '')  
 email = request.POST.get('email', '')  
 try:  
 order = Orders.objects.filter(order\_id=orderId, email=email)  
 if len(order)>0:  
 update = OrderUpdate.objects.filter(order\_id=orderId)  
 updates = []  
 for item in update:  
 updates.append({'text': item.update\_desc, 'time': item.timestamp})  
 response = json.dumps([updates, order[0].items\_json], default=str)  
 return HttpResponse(response)  
 else:  
 return HttpResponse('{}')  
 except Exception as e:  
 return HttpResponse('{}')  
  
 return render(request, 'shop/tracker.html')



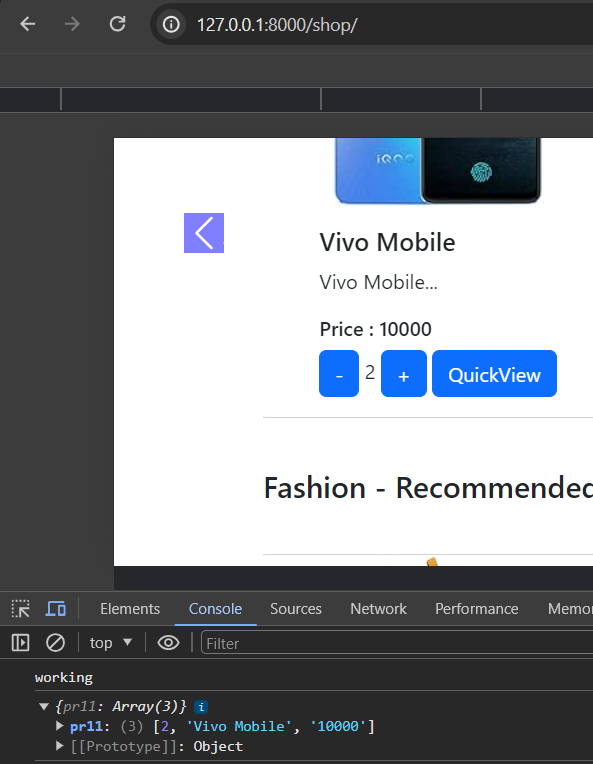
To Display Proces in Page

Add in dex.html of shop app

<h6 class="card-title" >Price : <span id="pricepr{{i.id}}">{{i.price}}</span></h6>

price = document.getElementById('price'+idstr).innerHTML  
cart[idstr] = [qty, name, price];

//$('.cart').click(function() {  
 $('.divpr').on('click', 'button.cart', function() {  
 var idstr = this.id.toString();  
 if (cart[idstr] != undefined) {  
 qty = cart[idstr][0] + 1;  
 } else {  
 qty = 1;  
 name = document.getElementById('name'+idstr).innerHTML  
 price = document.getElementById('price'+idstr).innerHTML  
 cart[idstr] = [qty, name, price];  
 }  
 updateCart(cart);  
 });



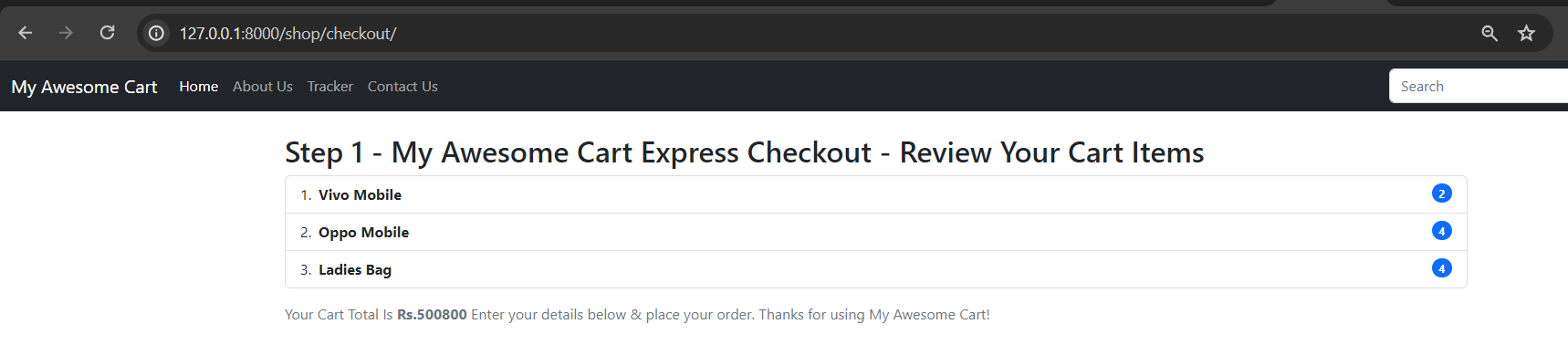
Checkout.html

{% extends 'shop/basic.html' %}  
{% block title%} Checkout - My Awesome Cart{% endblock %}  
{% block body %}  
<div class="container">  
 <div class="col my-4">  
 <h2>Step 1 - My Awesome Cart Express Checkout - Review Your Cart Items</h2>  
 <div>  
 <ul class="list-group list-group-numbered" id="items">  
 </ul>  
 <nav aria-label="breadcrumb">  
 <ol class="breadcrumb mt-3">  
 <li class="breadcrumb-item active" aria-current="page">Your Cart Total Is <b>Rs.<span id="totalPrice"></b> </span>Enter your details below & place your order. Thanks for using My Awesome Cart!</li>  
 </ol>  
 </nav>  
 </div>  
 </div>  
 <div class="col my-4">  
 <h2>Step 2 - Enter Address & Other Details:</h2>  
 <form method="post" action='/shop/checkout/' class="row g-3">{% csrf\_token %}  
 <input type="hidden" name="itemsJson" id="itemsJson">  
 <div class="col-md-6">  
 <label for="inputname" class="form-label">Name</label>  
 <input type="text" class="form-control" id="name" name="name" placeholder="Name">  
 </div>  
 <div class="col-md-6">  
 <label for="inputEmail4" class="form-label">Email</label>  
 <input type="email" class="form-control" id="email" name="email" placeholder="Email">  
 </div>  
 <div class="col-12">  
 <label for="inputAddress" class="form-label">Address</label>  
 <input type="text" class="form-control" id="address1" name="address1" placeholder="1234 Main St">  
 </div>  
 <div class="col-12">  
 <label for="inputAddress2" class="form-label">Address Line 2</label>  
 <input type="text" class="form-control" id="address2" name="address2" placeholder="Apartment, studio, or floor">  
 </div>  
 <div class="col-md-6">  
 <label for="inputCity" class="form-label">City</label>  
 <input type="text" class="form-control" id="city" name="city" placeholder="Enter City">  
 </div>  
 <div class="col-md-4">  
 <label for="inputState" class="form-label">State</label>  
 <input type="text" class="form-control" id="state" name="state" placeholder="Enter State">  
 </div>  
 <div class="col-md-2">  
 <label for="inputZip" class="form-label">Zip</label>  
 <input type="text" class="form-control" id="zip\_code" name="zip\_code" placeholder="Enter Zip Code">  
 </div>  
 <div class="col-md-2">  
 <label for="inputPhone" class="form-label">Phone No.</label>  
 <input type="text" class="form-control" id="phone" name="phone" placeholder="Enter Phone">  
 </div>  
 <div class="col-12">  
 <button type="submit" class="btn btn-primary">Place Order</button>  
 </div>  
 </form>  
 </div>  
</div>  
{% endblock %}  
  
{% block js %}  
<script type="text/javascript">  
 console.log("working");  
 var cart = {};  
  
 if(localStorage.getItem('cart') == null){  
 cart = {};  
 }  
 else  
 {  
 cart = JSON.parse(localStorage.getItem('cart'));  
 }  
  
 var sum=0;  
 var totalPrice=0;  
 if ($.isEmptyObject(cart)){  
<!-- if object is empty-->  
 mystr = `<p>Your Cart is empty, please add some items in cart before checking out!!</p>`  
 $('#items').append(mystr);  
 }  
 else{  
 for (item in cart) {  
 let name = cart[item][1];  
 let qty = cart[item][0];  
 let itemPrice = cart[item][2];  
 sum = sum+qty;  
 totalPrice = totalPrice + qty\* itemPrice  
 mystr = ` <li class="list-group-item d-flex justify-content-between align-items-start">  
 <div class="ms-2 me-auto">  
 <div class="fw-bold">${name}</div>  
 </div>  
 <span class="badge bg-primary rounded-pill">${qty}</span>  
 </li>`  
 $('#items').append(mystr);  
 }  
 }  
  
 document.getElementById('cart').innerHTML = sum;  
 document.getElementById('totalPrice').innerHTML = totalPrice;  
 $('#itemsJson').val(JSON.stringify(cart));  
 {% if thank %}  
 alert("Thanks for ordering with us. Your order id is {{id}}. Use it to track your order using our order tracker")  
 localStorage.clear();  
 document.location="/shop";  
 {%endif%}  
</script>  
{% endblock %}

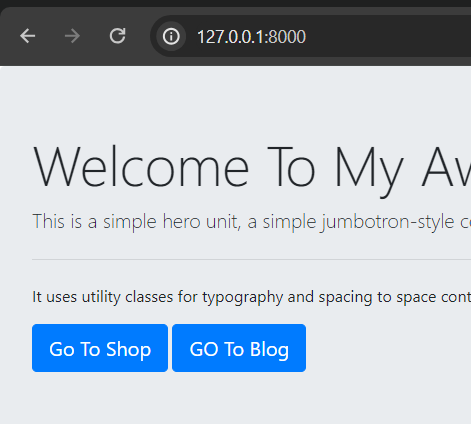
In index.html shopapp change price to integr

//$('.cart').click(function() {  
 $('.divpr').on('click', 'button.cart', function() {  
 var idstr = this.id.toString();  
 if (cart[idstr] != undefined) {  
 qty = cart[idstr][0] + 1;  
 } else {  
 qty = 1;  
 name = document.getElementById('name'+idstr).innerHTML  
 price = document.getElementById('price'+idstr).innerHTML  
 cart[idstr] = [qty, name, parseInt(price)];  
 }  
 updateCart(cart);  
 });

Now we can see totalprice in page



Merging Blog app to Shop UI

****

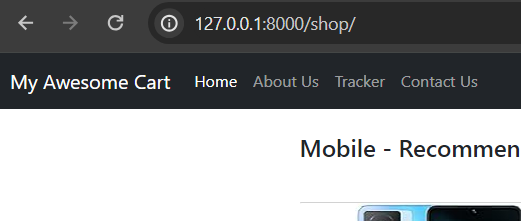
**When we goto this url we goto this page , but I want to redirect it to shop app**

**So use**

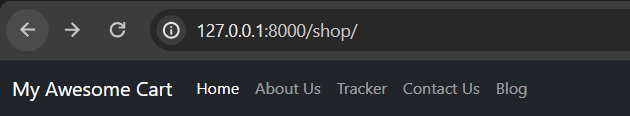
<script> window.location="/shop";</script>

**In index.html in templates of mac app**

**Now will auto redirect to**

****

**Add blog in navbar in basic.html in shop app**

****

<li class="nav-item">  
 <a class="nav-link" href="/blog">Blog</a>  
</li>

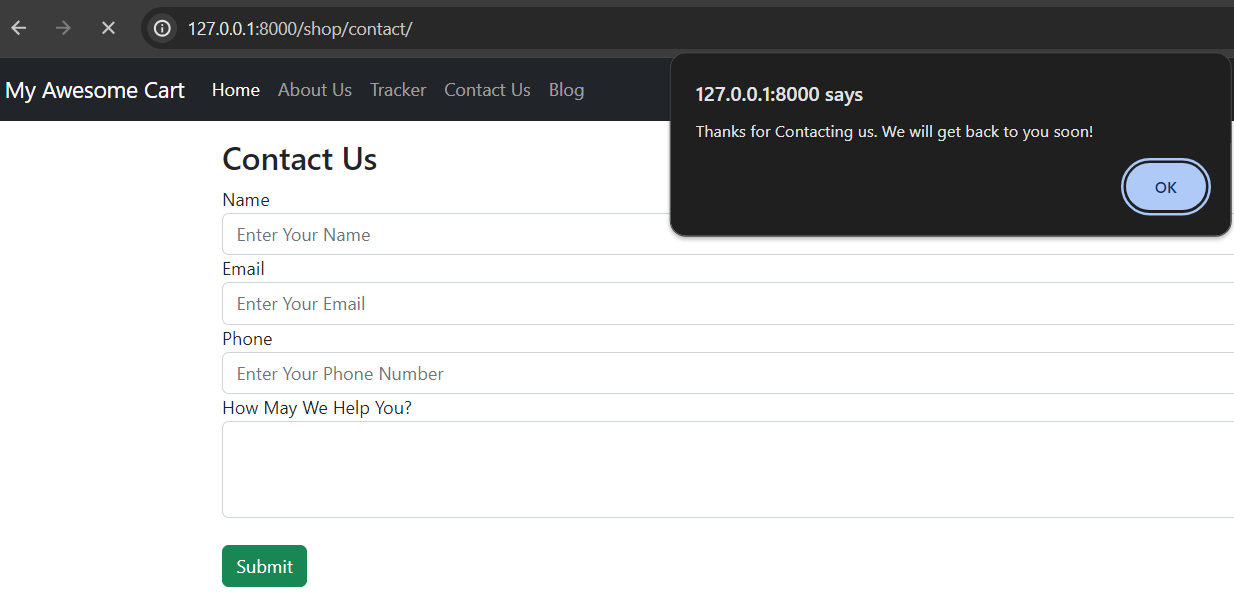
**In views.py of shop app**

def contact(request):  
 thank = False  
 if request.method == 'POST':  
 name = request.POST.get('name', '')  
 email = request.POST.get('email', '')  
 phone = request.POST.get('phone', '')  
 desc = request.POST.get('desc', '')  
 contact = Contact(name=name, email=email, phone=phone, desc=desc)  
 contact.save()  
 thank=True  
 return render(request, "shop/contact.html", {'thank': thank})

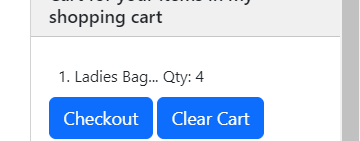
**added thank so that whever anyone add contact us on website he will see popup**

**in contact.html of shop app**

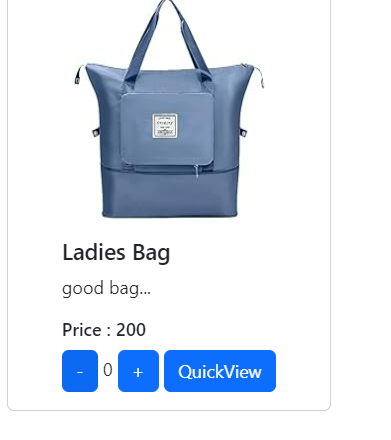
{% endblock %}  
{% block js %}  
<script type="text/javascript">  
 {% if thank %}  
 alert("Thanks for Contacting us. We will get back to you soon!")  
 document.location="/shop/contact";  
 {%endif%}  
</script>  
{% endblock %}

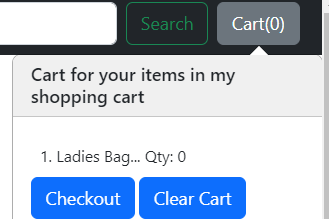
****

**When we add something in cart**

****

**And if we make it zero again**

**here**

****

**It remains zero but it must get removed from list**

**Shop/index.html change this and add if else block**

<!-- // If plus or minus button is clicked, change the cart as well as the display value-->  
 $('.divpr').on("click", "button.minus", function() {  
 console.log("Minus Clicked");  
 a = this.id.slice(7, );  
 cart['pr' + a][0] = cart['pr' + a][0] - 1;  
 cart['pr' + a][0] = Math.max(0, cart['pr' + a][0]);  
<!-- document.getElementById('valpr' + a).innerHTML = cart['pr' + a][0];-->  
 if (cart['pr' + a][0] == 0) {  
 for (var item in cart) {  
 document.getElementById('div' + item).innerHTML = '<button id = "' + item + '" class="btn btn-primary cart">Add to Cart</button>'  
 }  
 delete cart['pr' + a];  
 } else {  
 document.getElementById('valpr' + a).innerHTML = cart['pr' + a][0];  
 }  
 updateCart(cart);  
 });

**Creating the blog**

**Created blogpost.html page**

{%extends 'blog/basic.html'%}  
{%block body%}  
<div class="container">  
 <div class="row mt-4">  
 <div class="col-md-8 blog-main">  
 <h3 class="pb-4 mb-4 font-italic border-bottom">  
 Title  
 </h3>  
 <div class="blog-post">  
 <h2 class="blog-post-title">Heading 0</h2>  
 <p class="blog-post-meta">January 1, 2014 by <a href="#">CodeWithHarry</a></p>  
 <p>Heading 0 content</p>  
 <h2>Heading1</h2>  
 <p> Heading 1 content</p>  
 <h2>Heading2</h2>  
 <p>Heading 2 content</p>  
 </div><!-- /.blog-post -->  
 </div><!-- /.blog-main -->  
 <aside class="col-md-4 blog-sidebar">  
 <div class="p-4 mb-3 bg-light rounded">  
 <h4 class="font-italic">About</h4>  
 <p class="mb-0">My Awesome Cart is an <em>ecommerce website</em>. You can use it for buying and searching items online. </p>  
 </div>  
 <div class="p-4">  
 <h4 class="font-italic">Contact Us On: </h4>  
 <ol class="list-unstyled">  
 <li><a href="#">GitHub</a></li>  
 <li><a href="#">Twitter</a></li>  
 <li><a href="#">Facebook</a></li>  
 </ol>  
 </div>  
 </aside><!-- /.blog-sidebar -->  
 </div>  
</div>  
{%endblock%}

**In blog app urls.py**

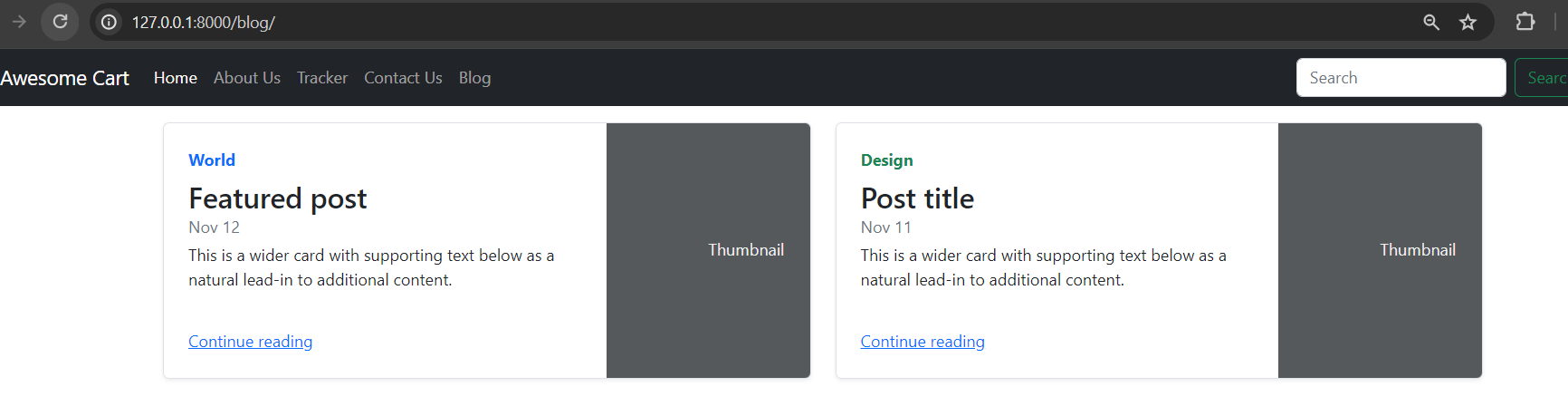
from django.urls import path, include  
from . import views  
  
urlpatterns = [  
 path('', views.index, name="BlogHome"),  
 path('blogpost/', views.blogpost, name="blogPost"),  
]

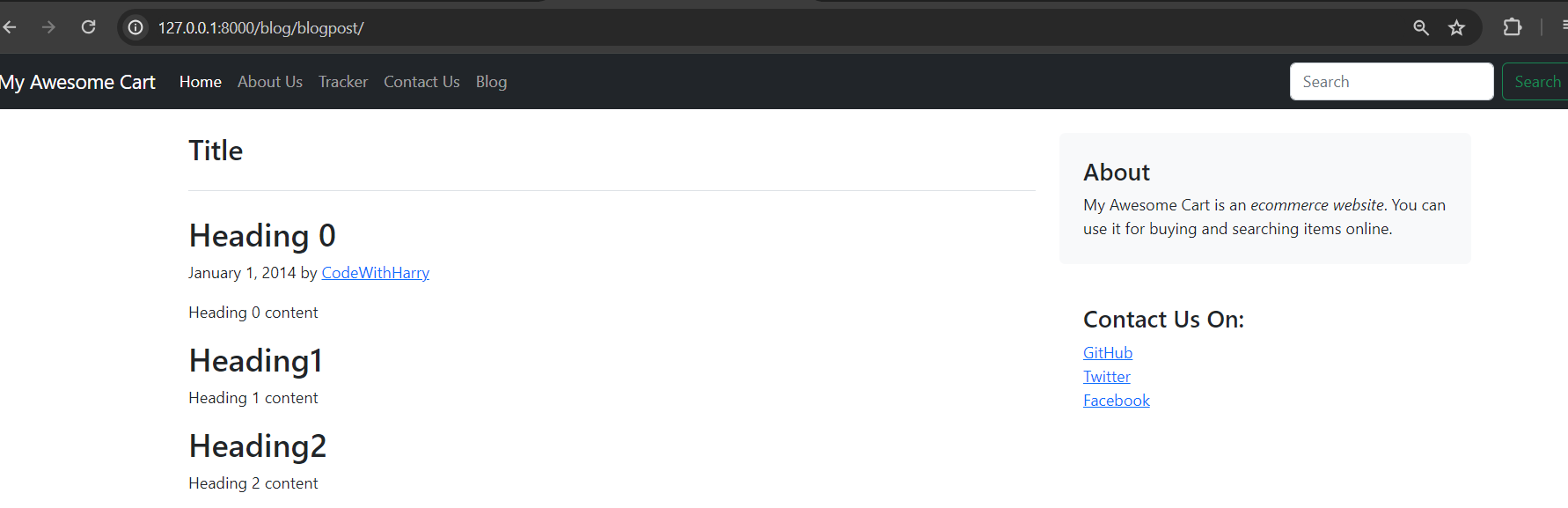
**views.py**

def blogpost(request):  
 return render(request, "blog/blogpost.html")

**also changed index.html of blog**

{%extends 'blog/basic.html'%}  
  
{%block body%}  
<div class="container mt-3">  
 <div class="row my-2">  
 <div class="col-md-6">  
 <div class="row g-0 border rounded overflow-hidden flex-md-row mb-4 shadow-sm h-md-250 position-relative">  
 <div class="col p-4 d-flex flex-column position-static">  
 <strong class="d-inline-block mb-2 text-primary">World</strong>  
 <h3 class="mb-0">Featured post</h3>  
 <div class="mb-1 text-muted">Nov 12</div>  
 <p class="card-text mb-auto">This is a wider card with supporting text below as a natural lead-in to  
 additional content.</p>  
 <a href="#" class="stretched-link">Continue reading</a>  
 </div>  
 <div class="col-auto d-none d-lg-block">  
 <svg class="bd-placeholder-img" width="200" height="250" xmlns="http://www.w3.org/2000/svg"  
 role="img"  
 aria-label="Placeholder: Thumbnail" preserveAspectRatio="xMidYMid slice" focusable="false">  
 <title>  
 Placeholder</title>  
 <rect width="100%" height="100%" fill="#55595c"></rect>  
 <text x="50%" y="50%" fill="#eceeef" dy=".3em">Thumbnail</text>  
 </svg>  
  
 </div>  
 </div>  
 </div>  
 <div class="col-md-6">  
 <div class="row g-0 border rounded overflow-hidden flex-md-row mb-4 shadow-sm h-md-250 position-relative">  
 <div class="col p-4 d-flex flex-column position-static">  
 <strong class="d-inline-block mb-2 text-success">Design</strong>  
 <h3 class="mb-0">Post title</h3>  
 <div class="mb-1 text-muted">Nov 11</div>  
 <p class="mb-auto">This is a wider card with supporting text below as a natural lead-in to  
 additional  
 content.</p>  
 <a href="#" class="stretched-link">Continue reading</a>  
 </div>  
 <div class="col-auto d-none d-lg-block">  
 <svg class="bd-placeholder-img" width="200" height="250" xmlns="http://www.w3.org/2000/svg"  
 role="img"  
 aria-label="Placeholder: Thumbnail" preserveAspectRatio="xMidYMid slice" focusable="false">  
 <title>  
 Placeholder</title>  
 <rect width="100%" height="100%" fill="#55595c"></rect>  
 <text x="50%" y="50%" fill="#eceeef" dy=".3em">Thumbnail</text>  
 </svg>  
  
 </div>  
 </div>  
 </div>  
 </div>  
</div>  
{%endblock%}

****

****

**In models.py of blog app**

from django.db import models  
  
# Create your models here.  
  
class Blogpost(models.Model):  
 post\_id = models.AutoField(primary\_key= True)  
 tilte = models.CharField(max\_length=50)  
 head0 = models.CharField(max\_length=500, default="")  
 chead0 = models.CharField(max\_length=5000, default="")  
 head1 = models.CharField(max\_length=500, default="")  
 chead1 = models.CharField(max\_length=5000, default="")  
 head2 = models.CharField(max\_length=500, default="")  
 chead2 = models.CharField(max\_length=5000, default="")  
 pub\_date = models.DateField()  
 thumbnail = models.ImageField(upload\_to='shop/images', default="")  
  
 def \_\_str\_\_(self):  
 return self.tilte

admin.py

from django.contrib import admin  
  
# Register your models here.  
from django.contrib import admin  
from .models import Blogpost  
admin.site.register(Blogpost)

//registered model

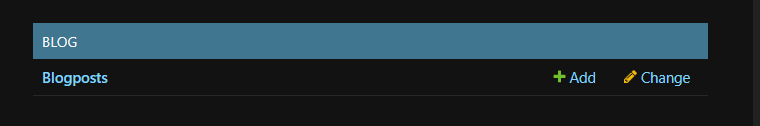
The next step is to store the changes in the database. So, fire the below commands one by one:

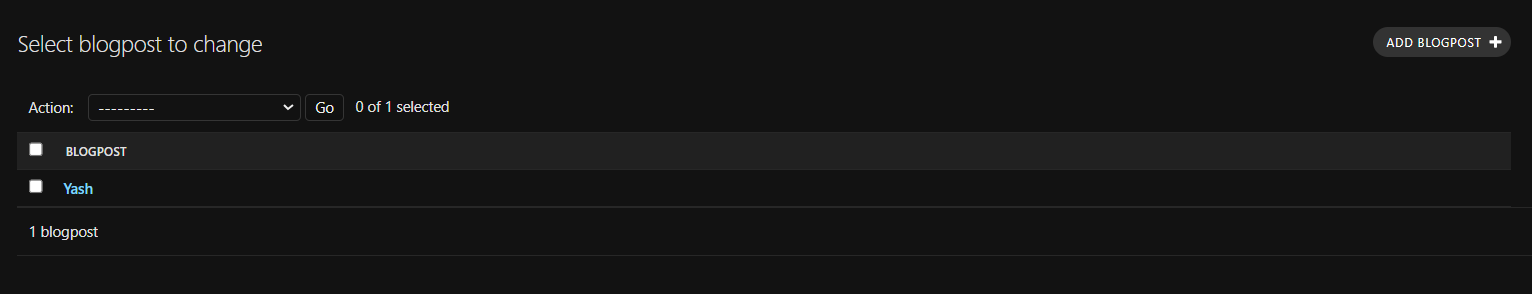
python manage.py makemigrations

Copy

python manage.py migrate

**/admin**

****

****

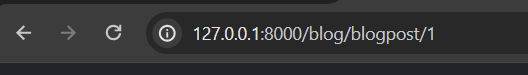
In urls.py of blog app

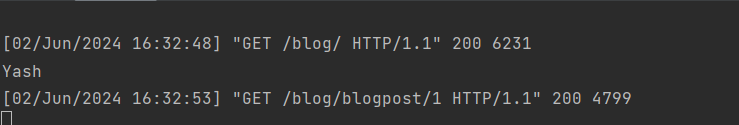
from django.urls import path  
from . import views  
  
urlpatterns = [  
 path("", views.index, name="blogHome"),  
 path("blogpost/<int:id>", views.blogpost, name="blogHome")  
]

**this id will get in views.blogpost will capture it in it**

**views.py**

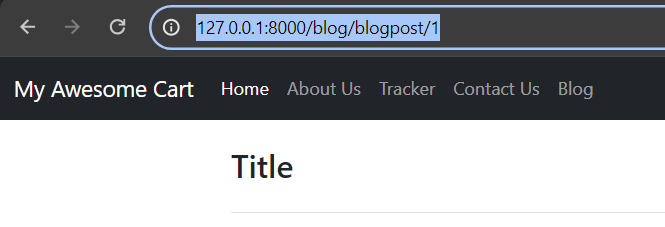
from django.shortcuts import render  
from django.shortcuts import HttpResponse  
from .models import Blogpost  
# Create your views here.  
def index(request):  
 return render(request, "blog/index.html")  
  
def blogpost(request, id):  
 post = Blogpost.objects.filter(post\_id = id)[0]  
 print(post)  
 return render(request, 'blog/blogpost.html',{'post':post})

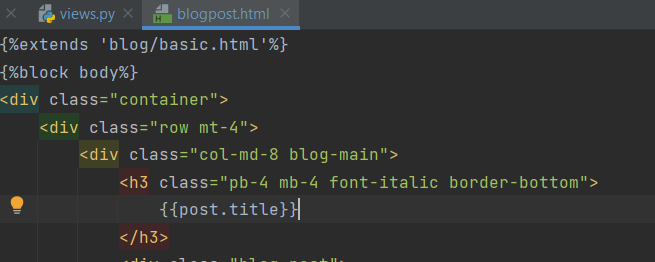
****

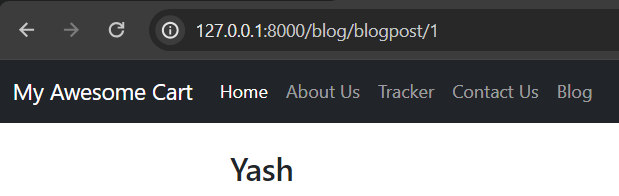
****

**This means we have got the blog post now will send blogpost to template**

**Before**

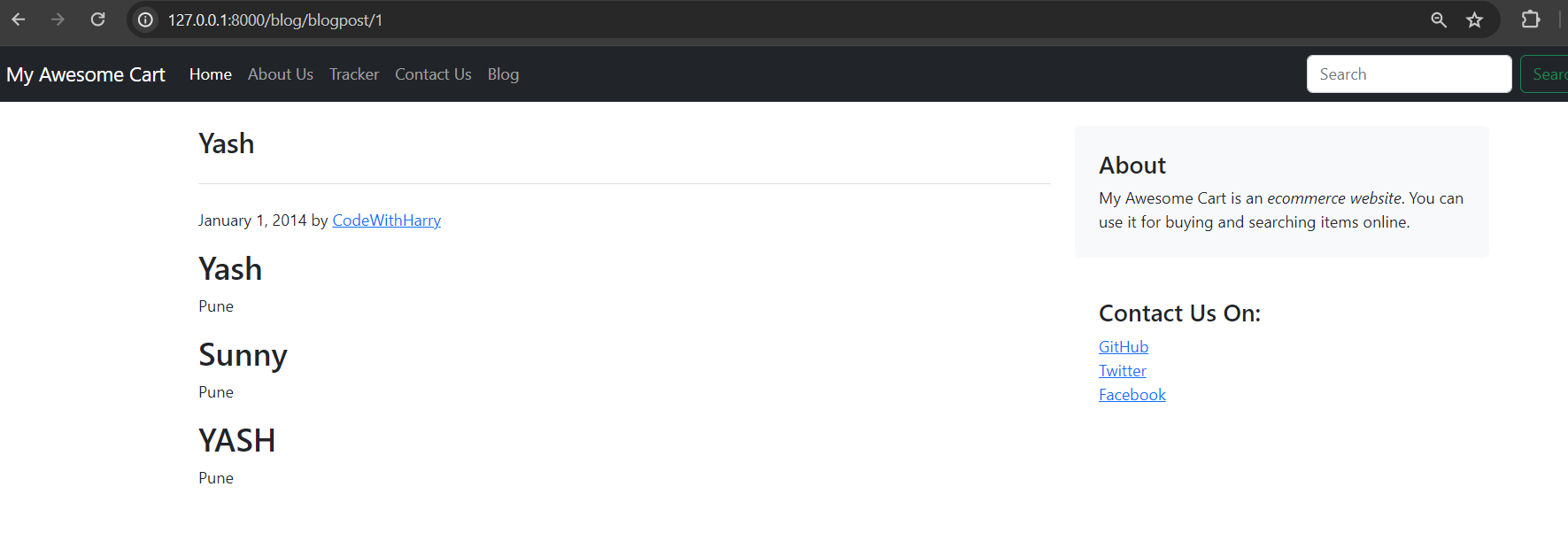
****

****

****

**Blogpost.html**

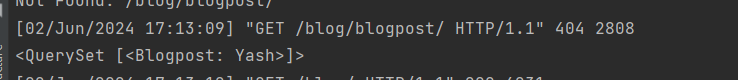
{%extends 'blog/basic.html'%}  
{%block body%}  
<div class="container">  
 <div class="row mt-4">  
 <div class="col-md-8 blog-main">  
 <h3 class="pb-4 mb-4 font-italic border-bottom">  
 {{post.title}}  
 </h3>  
 <div class="blog-post">  
 <h2 class="blog-post-title">{{post.heading0}}</h2>  
 <p class="blog-post-meta">January 1, 2014 by <a href="#">CodeWithHarry</a></p>  
 <h2>{{post.head0}}</h2>  
 <p>{{post.chead0}}</p>  
 <h2>{{post.head1}}</h2>  
 <p>{{post.chead1}}</p>  
 <h2>{{post.head2}}</h2>  
 <p>{{post.chead2}}</p>  
 </div><!-- /.blog-post -->  
 </div><!-- /.blog-main -->

****

**Designing the Blog Home Page**

**Views.py**

from django.shortcuts import render  
from django.shortcuts import HttpResponse  
from .models import Blogpost  
# Create your views here.  
def index(request):  
 myposts= Blogpost.objects.all()  
 print(myposts)  
 return render(request, 'blog/index.html', {'myposts': myposts})

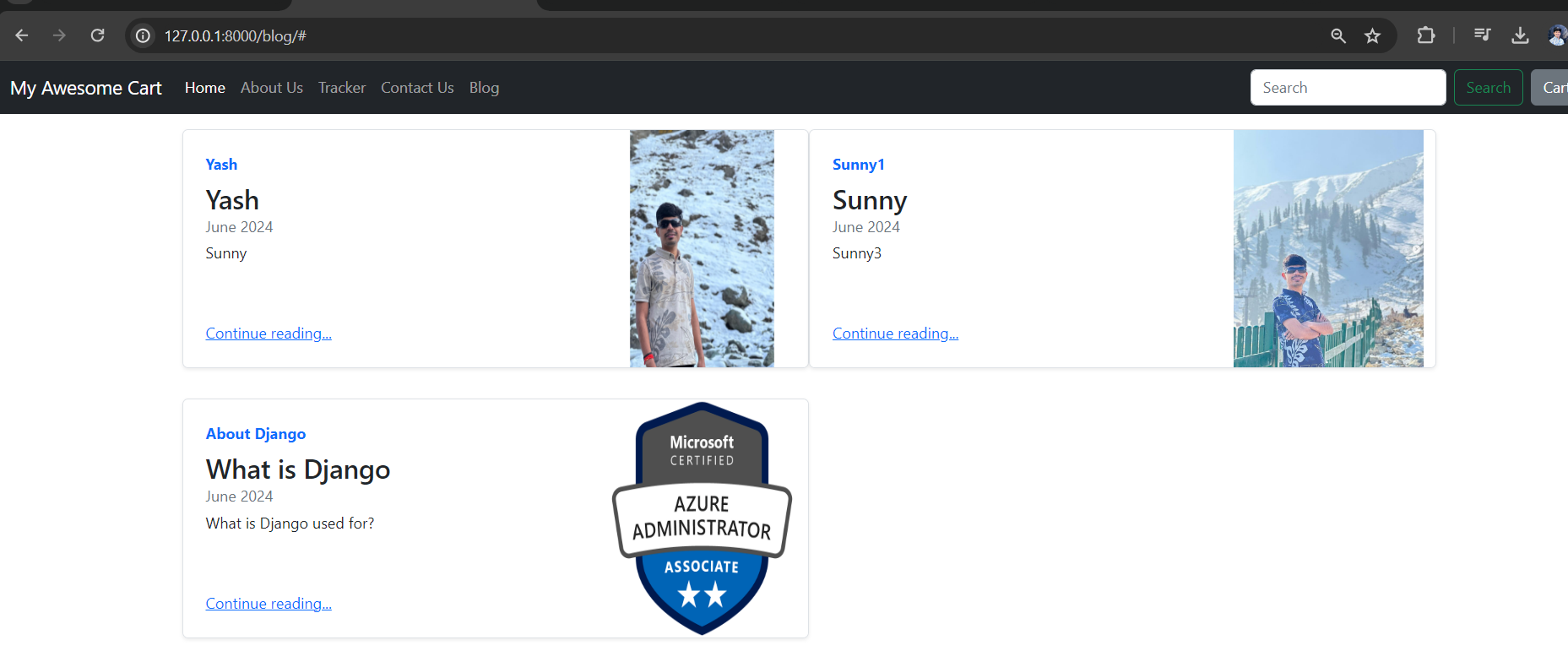
****

**Index.html**

{%extends 'blog/basic.html'%}  
{%block body%}  
<div class="container mt-3">  
 <div class="row my-2">  
 {% for item in myposts %}  
 <div class="col-md-6">  
 <div class="row no-gutters border rounded overflow-hidden flex-md-row mb-4 shadow-sm h-md-250 position-relative">  
 <div class="col p-4 d-flex flex-column position-static">  
 <strong class="d-inline-block mb-2 text-primary">  
 {{item.head0}}  
 </strong>  
 <h3 class="mb-0">{{item.title}}</h3>  
 <div class="mb-1 text-muted">June 2024</div>  
 <p class="card-text mb-auto">{{item.head1}}</p>  
 <a href="#" class="stretched-link">Continue reading...</a>  
 </div>  
 <div class="col-auto d-none d-lg-block">  
 <img src="/media/{{item.thumbnail}}" class="bd-placeholder-img" width="200" height="250" aria-label="Placeholder: Thumbnail">  
 <title>Placeholder</title></img>  
 </div>  
 </div>  
 </div>  
 {% if forloop.counter|divisibleby:2%}  
 </div>  
 <div class="row my-2">  
 {%endif%}  
 {%endfor%}</div>  
</div>  
{%endblock%}

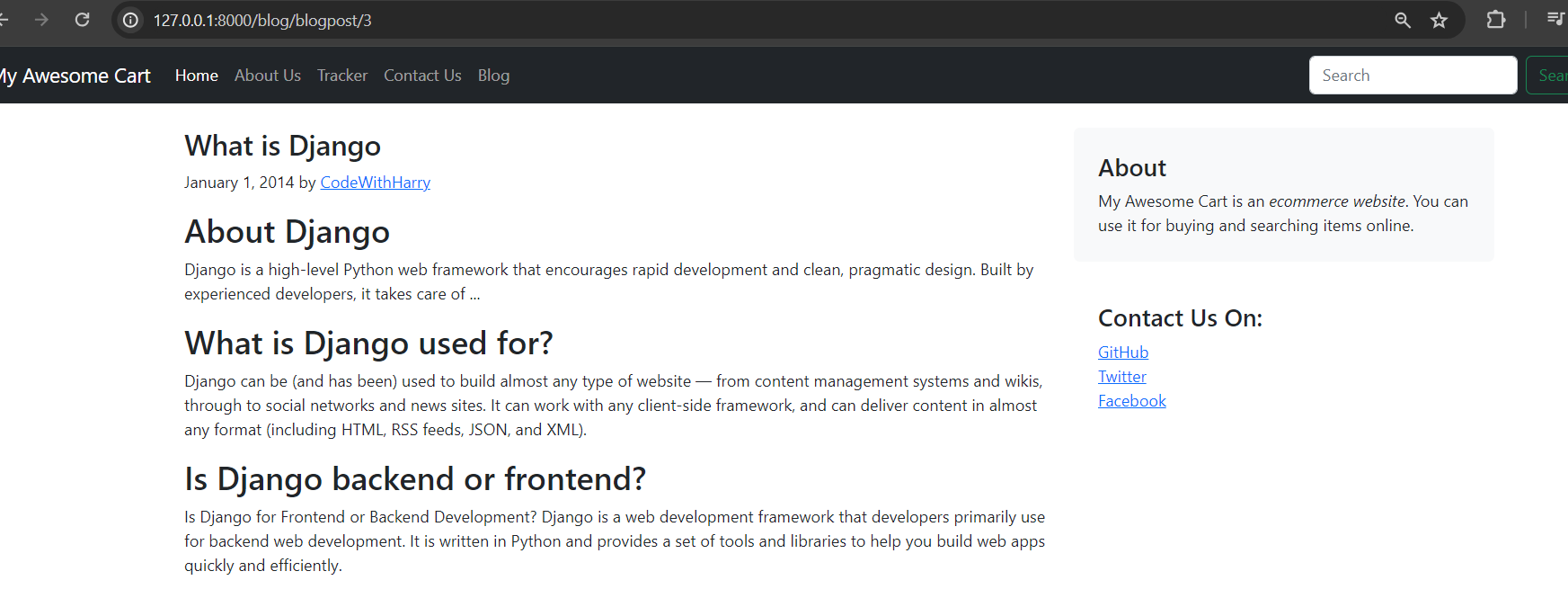
****

**//Linking Django Posts to blog home**

****

**Here when we click on continue reading its not working**

<a href="blogpost/{{item.post\_id}}" class="stretched-link">Continue reading...</a>

****

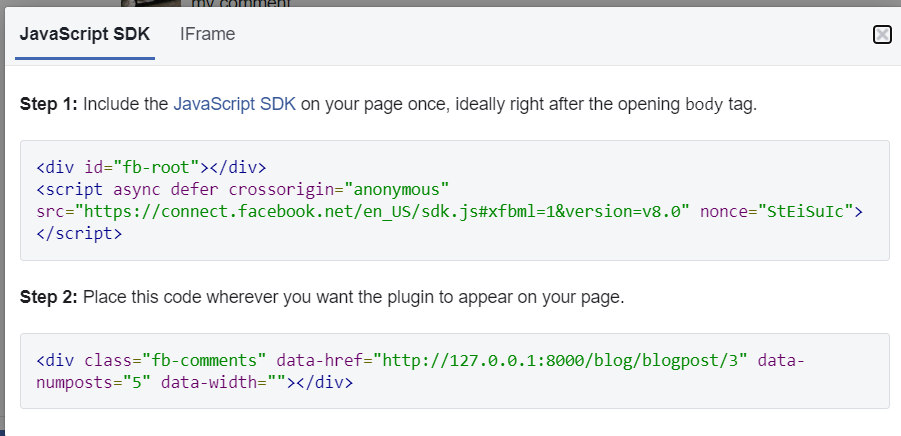
**Its working now**

**Integrating Facebook Like, Comments & Shares In Django Blog**

**Goto google**

**Imagine how amazing it will be if we provide the functionality to share the blog posts directly to the user's Facebook account. We will use the Facebook plugins to integrate Facebook like, comment, and share functionality on our E-commerce website's blog section. Open the base.html file and type the below just after the </style> block :**

**<script async defer crossorigin="anonymous" src="https://connect.facebook.net/en\_GB/sdk.js#xfbml=1&version=v3.2"></script>**

**The above javascript SDK is generated from the developer page of Facebook. You can generate the Facebook js SDK for any website from here. To generate SDK, you need to paste the URL of the website where you want to use the comments plugin in the code configurator provided by Facebook. After this, click on the "Get code" button, you will see a screen like this :** 

Blogpost.html

{%extends 'blog/basic.html'%}  
{%block body%}  
<div class="container">  
 <div class="row mt-4">  
 <div class="col-md-8 blog-main">  
 <h3 class="font-italic">  
 {{post.title}}  
 </h3>  
 <div class="blog-post">  
 <h2 class="blog-post-title">{{post.heading0}}</h2>  
 <p class="blog-post-meta">January 1, 2014 by <a href="#">CodeWithHarry</a></p>  
 <div class="fb-like" data-href="http://127.0.0.1:8000/blog/blogpost/{{post.post\_id}}" data-width="" data-layout="" data-action="" data-size="" data-share="true"></div>  
 <h2>{{post.head0}}</h2>  
 <p>{{post.chead0}}</p>  
 <h2>{{post.head1}}</h2>  
 <p>{{post.chead1}}</p>  
 <h2>{{post.head2}}</h2>  
 <p>{{post.chead2}}</p>  
 <div class="fb-comments" data-href="http://127.0.0.1:8000/blog/blogpost/{{post.post\_id}}" data-width="" data-numposts="5"></div>  
  
 </div><!-- /.blog-post -->  
 </div><!-- /.blog-main -->  
 <aside class="col-md-4 blog-sidebar">  
 <div class="p-4 mb-3 bg-light rounded">  
 <h4 class="font-italic">About</h4>  
 <p class="mb-0">My Awesome Cart is an <em>ecommerce website</em>. You can use it for buying and searching items online. </p>  
 </div>  
 <div class="my-3 fb-share-button" data-href="http://127.0.0.1:8000/blog/blogpost/{{post.post\_id}}" data-layout="" data-size=""><a target="\_blank" href="https://www.facebook.com/sharer/sharer.php?u=http%3A%2F%2F127.0.0.1%3A8000%2Fblog%2Fblogpost%2F3&amp;src=sdkpreparse" class="fb-xfbml-parse-ignore">Share</a></div> <div class="p-4">  
 <h4 class="font-italic">Contact Us On: </h4>  
 <ol class="list-unstyled">  
 <li><a href="#">GitHub</a></li>  
 <li><a href="#">Twitter</a></li>  
 <li><a href="#">Facebook</a></li>  
 </ol>  
 </div>  
 </aside><!-- /.blog-sidebar -->  
 </div>  
</div>  
{%endblock%}

**Basic.html**

<!doctype html>  
<html lang="en">  
<head>  
 <meta charset="utf-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1">  
 <title>{%block title%} {% endblock %}</title>  
 <style>  
 {%block css%} {% endblock %}  
 </style>  
<!-- <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.1/jquery.min.js"></script></script>-->  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet"  
 integrity="sha384-rbsA2VBKQhggwzxH7pPCaAqO46MgnOM80zW1RWuH61DGLwZJEdK2Kadq2F9CUG65" crossorigin="anonymous">  
 <script async defer crossorigin="anonymous" src="https://connect.facebook.net/en\_US/sdk.js#xfbml=1&version=v20.0" nonce="TiygyUwu"></script>  
</head>  
<body>  
{% load static %}  
<nav class="navbar navbar-expand-lg navbar-dark bg-dark">  
 <div class="container-fluid">  
 <a class="navbar-brand" href="/shop">My Awesome Cart</a>  
 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"  
 aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">  
 <span class="navbar-toggler-icon"></span>  
 </button>  
 <div class="collapse navbar-collapse" id="navbarSupportedContent">  
 <ul class="navbar-nav me-auto mb-2 mb-lg-0">  
 <li class="nav-item">  
 <a class="nav-link active" aria-current="page" href="/shop">Home</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="/shop/about">About Us</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="/shop/tracker">Tracker</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="/shop/contact">Contact Us</a>  
 </li>  
 <li class="nav-item">  
 <a class="nav-link" href="/blog">Blog</a>  
 </li>  
  
 </ul>  
 <form class="d-flex" role="search">  
 <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search">  
 <button class="btn btn-outline-success" type="submit">Search</button>  
 </form>  
 <button type="button" class="btn btn-secondary mx-2" data-bs-toggle="popover" title="Cart for your items in my shopping cart"  
 data-html='true' id ='popcart' data-bs-content="Cart for your items in shopping cart">Cart(<span  
 id="cart">0</span>)  
 </button>  
 </div>  
 </div>  
</nav>  
  
{%block body%} {% endblock %}  
  
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js"  
 integrity="sha384-oBqDVmMz9ATKxIep9tiCxS/Z9fNfEXiDAYTujMAeBAsjFuCZSmKbSSUnQlmh/jp3"  
 crossorigin="anonymous"></script>  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.min.js"  
 integrity="sha384-cuYeSxntonz0PPNlHhBs68uyIAVpIIOZZ5JqeqvYYIcEL727kskC66kF92t6Xl2V"  
 crossorigin="anonymous"></script>  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/js/bootstrap.bundle.min.js"  
 integrity="sha384-kenU1KFdBIe4zVF0s0G1M5b4hcpxyD9F7jL+jjXkk+Q2h455rYXK/7HAuoJl+0I4"  
 crossorigin="anonymous"></script>  
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js"></script>  
<script src="//ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js"></script>  
 <script src="https://code.jquery.com/jquery-3.3.1.js"  
 integrity="sha256-2Kok7MbOyxpgUVvAk/HJ2jigOSYS2auK4Pfzbm7uH60="  
 crossorigin="anonymous"></script>  
  
{%block js%} {% endblock %}  
  
</body>  
</html>

**Integrating Payment Gateway: PayTM, UPI, Cards & NetBanking**

**Adding payment option**

**Add amount field in models of shop app**

class Orders(models.Model):  
 order\_id= models.AutoField(primary\_key=True)  
 items\_json= models.CharField(max\_length=5000)  
 amount = models.IntegerField(default=0)  
 name=models.CharField(max\_length=90)  
 email=models.CharField(max\_length=111)  
 phone=models.CharField(max\_length=111, default="")  
 address=models.CharField(max\_length=111)  
 city=models.CharField(max\_length=111)  
 state=models.CharField(max\_length=111)  
 zip\_code=models.CharField(max\_length=111)  
  
 def \_\_str\_\_(self):  
 return self.name

**make migrations now**

**add hidden field amount in checkout.html**

<div class="col my-4">  
 <h2>Step 2 - Enter Address & Other Details:</h2>  
 <form method="post" action='/shop/checkout/' class="row g-3">{% csrf\_token %}  
 <input type="hidden" name="itemsJson" id="itemsJson">  
 <input type="hidden" name="amount" id="amount">  
 <div class="col-md-6">  
 <label for="inputname" class="form-label">Name</label>

**Checkout.html**

**Add**

$('#amount').val($('#totalPrice').html())

</script>  
{% endblock %}

**Views.html**

**Add amount field**

def checkout(request):  
 if request.method == 'POST':  
 items\_json = request.POST.get('itemsJson', '')  
 name = request.POST.get('name', '')  
 amount = request.POST.get('amount', '')  
 email = request.POST.get('email', '')  
 address = request.POST.get('address1', '') + request.POST.get('address2', '')  
 city = request.POST.get('city', '')  
 state = request.POST.get('state', '')  
 zip\_code = request.POST.get('zip\_code', '')  
 phone = request.POST.get('phone', '')  
  
 order = Orders(name=name, email=email, address=address, city=city, state=state, zip\_code=zip\_code, phone=phone, items\_json=items\_json,  
 amount=amount)  
 order.save()  
 update = OrderUpdate(order\_id = order.order\_id, update\_desc="The order has been placed")  
 update.save()  
 thank = True  
 id = order.order\_id  
 return render(request, "shop/checkout.html", {'thank':thank, 'id':id})  
 return render(request, "shop/checkout.html")

**now we can see amount in admin page also in orders**

**Now will send post request to paytm from sumit checkout page, then paytm will take details validates and sends post request to us with status**

**pip install pycryptodome**

**paymentstatus.html**

{% extends 'shop/basic.html' %}  
{% block title%} My Awesome Cart Tracker{% endblock %}  
{% block body %}  
<div class="container">  
 {{response}}  
 <div class="col my-4">  
  
 <h2>Payment status regarding your order Id {{response.ORDERID}}</h2>  
 {% if response.RESPCODE == '01' %}  
 ORDER SUCCESS  
 {% else %}  
 ORDER FAILURE  
 {% endif%}  
  
 </div>  
  
</div>  
{% endblock %}  
{% block js %}  
<script>  
</script>  
{% endblock %}

**Paytm.html**

<!DOCTYPE html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <title>Paytm merchant payment page</title>  
</head>  
<body>  
<h1>Redirecting you to the merchant....</h1>  
<h1>Please do not refresh your page....</h1>  
<form action="https://securegw-stage.paytm.in/theia/processTransaction" method="post" name="paytm">  
 {% for key, value in param\_dict.items %}  
 <input type="hidden" name="{{key}}" value="{{value}}">  
 {% endfor %}  
  
</form>  
</body>  
<script>  
document.paytm.submit()  
</script>  
</html>

**Checksum.html**

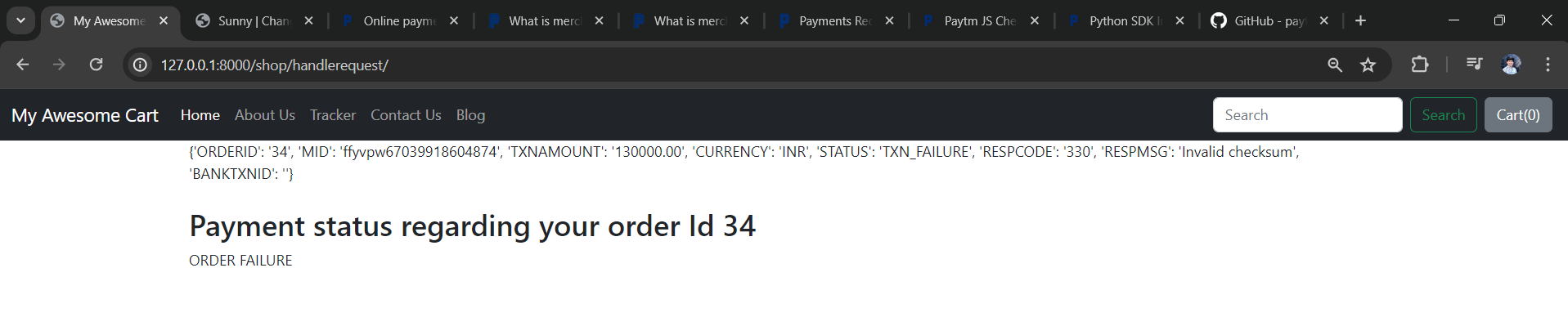
# pip install pycryptodome  
import base64  
import string  
import random  
import hashlib  
  
from Crypto.Cipher import AES  
  
  
IV = "@@@@&&&&####$$$$"  
BLOCK\_SIZE = 16  
  
  
def generate\_checksum(param\_dict, merchant\_key, salt=None):  
 params\_string = \_\_get\_param\_string\_\_(param\_dict)  
 salt = salt if salt else \_\_id\_generator\_\_(4)  
 final\_string = '%s|%s' % (params\_string, salt)  
  
 hasher = hashlib.sha256(final\_string.encode())  
 hash\_string = hasher.hexdigest()  
  
 hash\_string += salt  
  
 return \_\_encode\_\_(hash\_string, IV, merchant\_key)  
  
def generate\_refund\_checksum(param\_dict, merchant\_key, salt=None):  
 for i in param\_dict:  
 if("|" in param\_dict[i]):  
 param\_dict = {}  
 exit()  
 params\_string = \_\_get\_param\_string\_\_(param\_dict)  
 salt = salt if salt else \_\_id\_generator\_\_(4)  
 final\_string = '%s|%s' % (params\_string, salt)  
  
 hasher = hashlib.sha256(final\_string.encode())  
 hash\_string = hasher.hexdigest()  
  
 hash\_string += salt  
  
 return \_\_encode\_\_(hash\_string, IV, merchant\_key)  
  
  
def generate\_checksum\_by\_str(param\_str, merchant\_key, salt=None):  
 params\_string = param\_str  
 salt = salt if salt else \_\_id\_generator\_\_(4)  
 final\_string = '%s|%s' % (params\_string, salt)  
  
 hasher = hashlib.sha256(final\_string.encode())  
 hash\_string = hasher.hexdigest()  
  
 hash\_string += salt  
  
 return \_\_encode\_\_(hash\_string, IV, merchant\_key)  
  
  
def verify\_checksum(param\_dict, merchant\_key, checksum):  
 # Remove checksum  
 if 'CHECKSUMHASH' in param\_dict:  
 param\_dict.pop('CHECKSUMHASH')  
  
 # Get salt  
 paytm\_hash = \_\_decode\_\_(checksum, IV, merchant\_key)  
 salt = paytm\_hash[-4:]  
 calculated\_checksum = generate\_checksum(param\_dict, merchant\_key, salt=salt)  
 return calculated\_checksum == checksum  
  
def verify\_checksum\_by\_str(param\_str, merchant\_key, checksum):  
 # Remove checksum  
 #if 'CHECKSUMHASH' in param\_dict:  
 #param\_dict.pop('CHECKSUMHASH')  
  
 # Get salt  
 paytm\_hash = \_\_decode\_\_(checksum, IV, merchant\_key)  
 salt = paytm\_hash[-4:]  
 calculated\_checksum = generate\_checksum\_by\_str(param\_str, merchant\_key, salt=salt)  
 return calculated\_checksum == checksum  
  
  
  
def \_\_id\_generator\_\_(size=6, chars=string.ascii\_uppercase + string.digits + string.ascii\_lowercase):  
 return ''.join(random.choice(chars) for \_ in range(size))  
  
  
def \_\_get\_param\_string\_\_(params):  
 params\_string = []  
 for key in sorted(params.keys()):  
 if("REFUND" in params[key] or "|" in params[key]):  
 respons\_dict = {}  
 exit()  
 value = params[key]  
 params\_string.append('' if value == 'null' else str(value))  
 return '|'.join(params\_string)  
  
  
\_\_pad\_\_ = lambda s: s + (BLOCK\_SIZE - len(s) % BLOCK\_SIZE) \* chr(BLOCK\_SIZE - len(s) % BLOCK\_SIZE)  
\_\_unpad\_\_ = lambda s: s[0:-ord(s[-1])]  
  
  
def \_\_encode\_\_(to\_encode, iv, key):  
 # Pad  
 to\_encode = \_\_pad\_\_(to\_encode)  
 # Encrypt  
 c = AES.new(key.encode('utf-8'), AES.MODE\_CBC, iv.encode('utf-8'))  
 to\_encode = c.encrypt(to\_encode.encode('utf-8'))  
 # Encode  
 to\_encode = base64.b64encode(to\_encode)  
 return to\_encode.decode("UTF-8")  
  
  
def \_\_decode\_\_(to\_decode, iv, key):  
 # Decode  
 to\_decode = base64.b64decode(to\_decode)  
 # Decrypt  
 c = AES.new(key.encode('utf-8'), AES.MODE\_CBC, iv.encode('utf-8'))  
 to\_decode = c.decrypt(to\_decode)  
 if type(to\_decode) == bytes:  
 # convert bytes array to str.  
 to\_decode = to\_decode.decode()  
 # remove pad  
 return \_\_unpad\_\_(to\_decode)  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 params = {  
 "MID": "WorldP64425807474247",  
 "ORDER\_ID": "order\_id",  
 "CUST\_ID": "cust\_id",  
 "TXN\_AMOUNT": "1",  
 "CHANNEL\_ID": "WEB",  
 "INDUSTRY\_TYPE\_ID": "Retail",  
 "WEBSITE": "xxxxxxxxxxx"  
 }  
  
 print(verify\_checksum(  
 params, 'xxxxxxxxxxxxxxxx',  
 "CD5ndX8VVjlzjWbbYoAtKQIlvtXPypQYOg0Fi2AUYKXZA5XSHiRF0FDj7vQu66S8MHx9NaDZ/uYm3WBOWHf+sDQAmTyxqUipA7i1nILlxrk="))  
  
 # print(generate\_checksum(params, "xxxxxxxxxxxxxxxx"))

**Urls.py**

from django.urls import path  
from . import views  
  
urlpatterns = [  
 path('', views.index, name="ShopHome"),  
 path('about/', views.about, name="AboutUs"),  
 path('contact/', views.contact, name="ContactUs"),  
 path('tracker/', views.tracker, name="TrackingStatus"),  
 path('search/', views.search, name="Search"),  
 path('products/<int:myid>', views.prodView, name="ProductView"),  
 path('checkout/', views.checkout, name="Checkout"),  
 path("handlerequest/", views.handlerequest, name="HandleRequest"),  
]

**views.py**

from django.shortcuts import render  
from django.http import HttpResponse  
from .models import Product, Contact, Orders, OrderUpdate  
from django.views.decorators.csrf import csrf\_exempt  
from math import ceil  
import json  
from django.http import HttpResponse  
from PayTm import Checksum  
  
# MERCHANT\_KEY = 'kbzk1DSbJiV\_03p5';  
MERCHANT\_KEY = 'yWtPp&P4Gqh!lPTr';  
  
# Create your views here.  
def index(request):  
 # fetches products from database  
 products = Product.objects.all()  
 print(products)  
 n=len(products)  
 nSlides = n//4 + ceil((n/4) - (n//4))  
 # params = {'product' : products, 'no\_of\_slides':nSlides, 'range' : range(1, nSlides)}  
 #  
 # allProds = [[products, range(1, nSlides), nSlides],  
 # [products, range(1, nSlides), nSlides]]  
  
 allProds = []  
 catprods = Product.objects.values('category','id')  
 cats = {item['category'] for item in catprods}  
 for cat in cats:  
 prod = Product.objects.filter(category = cat)  
 n = len(prod)  
 nSlides = n//4 + ceil((n/4) - (n//4))  
 allProds.append([prod, range(1, nSlides), nSlides])  
  
 params = {'allProds' : allProds}  
 return render(request, "shop/index.html", params)  
  
def about(request):  
 return render(request, "shop/about.html")  
  
def contact(request):  
 thank = False  
 if request.method == 'POST':  
 name = request.POST.get('name', '')  
 email = request.POST.get('email', '')  
 phone = request.POST.get('phone', '')  
 desc = request.POST.get('desc', '')  
 contact = Contact(name=name, email=email, phone=phone, desc=desc)  
 contact.save()  
 thank=True  
 return render(request, "shop/contact.html", {'thank': thank})  
  
def tracker(request):  
 if request.method=="POST":  
 orderId = request.POST.get('orderId', '')  
 email = request.POST.get('email', '')  
 try:  
 order = Orders.objects.filter(order\_id=orderId, email=email)  
 if len(order)>0:  
 update = OrderUpdate.objects.filter(order\_id=orderId)  
 updates = []  
 for item in update:  
 updates.append({'text': item.update\_desc, 'time': item.timestamp})  
 response = json.dumps([updates, order[0].items\_json], default=str)  
 return HttpResponse(response)  
 else:  
 return HttpResponse('{}')  
 except Exception as e:  
 return HttpResponse('{}')  
  
 return render(request, 'shop/tracker.html')  
  
def search(request):  
 return render(request, "shop/search.html")  
  
def prodView(request, myid):  
 # fetch the product using id  
 product = Product.objects.filter(id=myid)  
 print(product)  
  
 return render(request, "shop/prodView.html", {'product': product[0]})  
  
def checkout(request):  
 if request.method == 'POST':  
 items\_json = request.POST.get('itemsJson', '')  
 name = request.POST.get('name', '')  
 amount = request.POST.get('amount', '')  
 email = request.POST.get('email', '')  
 address = request.POST.get('address1', '') + request.POST.get('address2', '')  
 city = request.POST.get('city', '')  
 state = request.POST.get('state', '')  
 zip\_code = request.POST.get('zip\_code', '')  
 phone = request.POST.get('phone', '')  
  
 order = Orders(name=name, email=email, address=address, city=city, state=state, zip\_code=zip\_code, phone=phone, items\_json=items\_json,  
 amount=amount)  
 order.save()  
 update = OrderUpdate(order\_id = order.order\_id, update\_desc="The order has been placed")  
 update.save()  
 thank = True  
 id = order.order\_id  
 # return render(request, "shop/checkout.html", {'thank':thank, 'id':id})  
 # Request paytm to transfer the amount to your account after payment by user  
 param\_dict = {  
  
 # 'MID': 'WorldP64425807474247',  
 'MID': 'ffyvpw67039918604874',  
 'ORDER\_ID': str(order.order\_id),  
 'TXN\_AMOUNT': str(amount),  
 'CUST\_ID': email,  
 'INDUSTRY\_TYPE\_ID': 'Retail',  
 'WEBSITE': 'WEBSTAGING',  
 'CHANNEL\_ID': 'WEB',  
 'CALLBACK\_URL': 'http://127.0.0.1:8000/shop/handlerequest/',  
  
 }  
 param\_dict['CHECKSUMHASH'] = Checksum.generate\_checksum(param\_dict, MERCHANT\_KEY)  
 return render(request, 'shop/paytm.html', {'param\_dict': param\_dict})  
 return render(request, "shop/checkout.html")  
  
#  
@csrf\_exempt  
# def handlerequest(request):  
# # paytm will send you post request here  
# form = request.POST  
# response\_dict = {}  
# for i in form.keys():  
# response\_dict[i] = form[i]  
# if i == 'CHECKSUMHASH':  
# checksum = form[i]  
#  
# verify = Checksum.verify\_checksum(response\_dict, MERCHANT\_KEY, checksum)  
# if verify:  
# if response\_dict['RESPCODE'] == '01':  
# print('order successful')  
# else:  
# print('order was not successful because' + response\_dict['RESPMSG'])  
# # return HttpResponse("Done")  
# # pass  
# return render(request, 'shop/paymentstatus.html', {'response': response\_dict})  
  
@csrf\_exempt  
def handlerequest(request):  
 if request.method == "POST":  
 try:  
 form = request.POST  
 response\_dict = {}  
 checksum = None # Initialize checksum variable  
 for key in form.keys():  
 response\_dict[key] = form[key]  
 if key == 'CHECKSUMHASH':  
 checksum = form[key] # Assign value to checksum  
  
 if checksum is not None: # Check if checksum is assigned a value  
 verify = Checksum.verify\_checksum(response\_dict, MERCHANT\_KEY, checksum)  
 if verify:  
 if response\_dict['RESPCODE'] == '01':  
 print('Order successful')  
 else:  
 print('Order was not successful because: ' + response\_dict['RESPMSG'])  
 else:  
 print('Checksum verification failed')  
 else:  
 print('Checksum not found in the response')  
  
 return render(request, 'shop/paymentstatus.html', {'response': response\_dict})  
  
 except Exception as e:  
 return HttpResponse("Error in payment processing: " + str(e))  
  
 return HttpResponse("Invalid request method")

****

**//handle request from chagpt not working so copied**

# Creating Product Search Feature In Our E Commerce Website

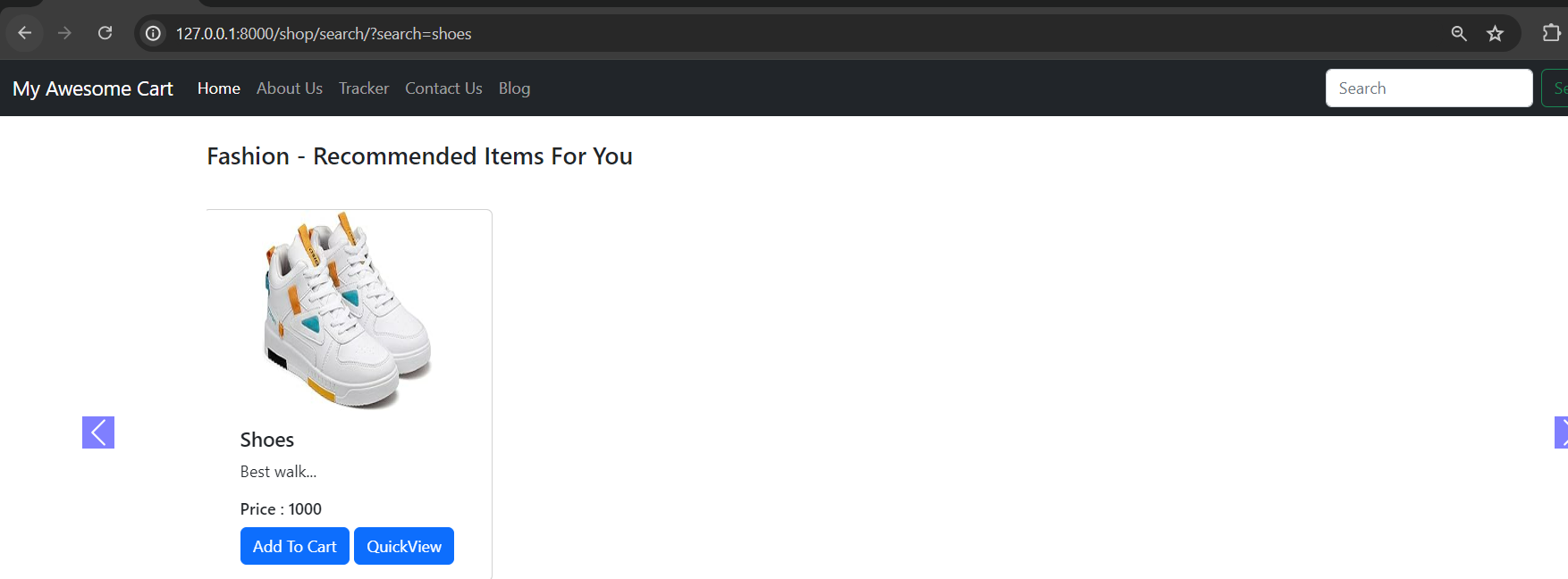
**To make search button to work**

**In basic.html change form tag to**

<form method="get" action="/shop/search/" class="d-flex" role="search">  
 <input class="form-control me-2" type="search" placeholder="Search" aria-label="Search" name="search" id="search">

**In views.py in shop app**

def searchMatch(query, item):  
 if query in item.desc.lower() or query in item.product\_name.lower() or query in item.category.lower():  
 return True  
 else:  
 return False  
def search(request):  
 query=request.GET.get('search')  
 allProds = []  
 catprods = Product.objects.values('category','id')  
 cats = {item['category'] for item in catprods}  
 for cat in cats:  
 prodtemp = Product.objects.filter(category = cat)  
 prod=[item for item in prodtemp if searchMatch(query, item)]  
 n = len(prod)  
 nSlides = n//4 + ceil((n/4) - (n//4))  
 if len(prod)!= 0:  
 allProds.append([prod, range(1, nSlides), nSlides])  
  
 params = {'allProds' : allProds}  
 return render(request, "shop/index.html", params)

****