**POLLSTER PROJECT**

G:\ >mkdir POLLSTER\_PROJECT

G:\ >cd POLLSTER\_PROJECT

G:\POLLSTER\_PROJECT>pipenv shell

(POLLSTER\_PROJECT-IcR9QnOD) G:\POLLSTER\_PROJECT>pipenv install django

(POLLSTER\_PROJECT-IcR9QnOD) G:\POLLSTER\_PROJECT>django-admin startproject pollster

(POLLSTER\_PROJECT-IcR9QnOD) G:\POLLSTER\_PROJECT>cd pollster

(POLLSTER\_PROJECT-IcR9QnOD) G:\POLLSTER\_PROJECT\pollster>python manage.py runserver

(POLLSTER\_PROJECT-IcR9QnOD) G:\POLLSTER\_PROJECT\pollster>python manage.py migrate

(POLLSTER\_PROJECT-IcR9QnOD) G:\POLLSTER\_PROJECT\pollster>python manage.py startapp polls

**Polls/models.py**

from django.db import models

class Question(models.Model):

    question\_text = models.CharField(max\_length=200)

    pub\_date = models.DateTimeField('date published')

    def \_\_str\_\_(self):

        return self.question\_text

class Choice(models.Model):

    question = models.ForeignKey(Question, on\_delete=models.CASCADE)

    choice\_text = models.CharField(max\_length=200)

    votes = models.IntegerField(default=0)

    def \_\_str\_\_(self):

        return self.choice\_text

**question = models.ForeignKey(Question, on\_delete=models.CASCADE)**

**This will create foreignkey for Question Table**

**on\_delete=models.CASCADE => If question is deleted then choice is also automatically deleted**

* If we create onle class and fields in admin we will get object so

def \_\_str\_\_(self):

        return self.choice\_text

This will give us Choice Text In admin dashboard

**pollster/settings.py**

INSTALLED\_APPS = [

    'polls.apps.PollsConfig',

]

(POLLSTER\_PROJECT-IcR9QnOD) G:\POLLSTER\_PROJECT\pollster>python manage.py makemigrations polls

(POLLSTER\_PROJECT-IcR9QnOD) G:\POLLSTER\_PROJECT\pollster>python manage.py migrate

(POLLSTER\_PROJECT-IcR9QnOD) G:\POLLSTER\_PROJECT\pollster>python manage.py shell (InteractiveConsole)

>>> from polls.models import Question, Choice

>>> Question.objects.all()

<QuerySet []>

>>> from django.utils import timezone

>>> q=Question(question\_text='What is your favorite Python Framework?', pub\_date=timezone.now())

>>> q.save()

>>> q.question\_text

'What is your favorite Python Framework?'

>>> Question.objects.all()

<QuerySet [<Question: What is your favorite Python Framework?>]>

>>> Question.objects.filter(id=1)

<QuerySet [<Question: What is your favorite Python Framework?>]>

=>This gives list in an array

>>> Question.objects.get(pk=1)

<Question: What is your favorite Python Framework?>

=>This gives us a single question

>>> q = Question.objects.get(pk=1)

>>> q.choice\_set.all()

<QuerySet []>

>>> q.choice\_set.create(choice\_text="Django", votes=0)

<Choice: Django>

>>> q.choice\_set.create(choice\_text="Flask", votes=0)

<Choice: Flask>

>>> q.choice\_set.create(choice\_text="Web2py", votes=0)

<Choice: Web2py>

>>> q.choice\_set.all()

<QuerySet [<Choice: Django>, <Choice: Flask>, <Choice: Web2py>]>

>>> quit()

(POLLSTER\_PROJECT-IcR9QnOD) G:\POLLSTER\_PROJECT\pollster>python manage.py createsuperuser

Username (leave blank to use 'smile'): oxus

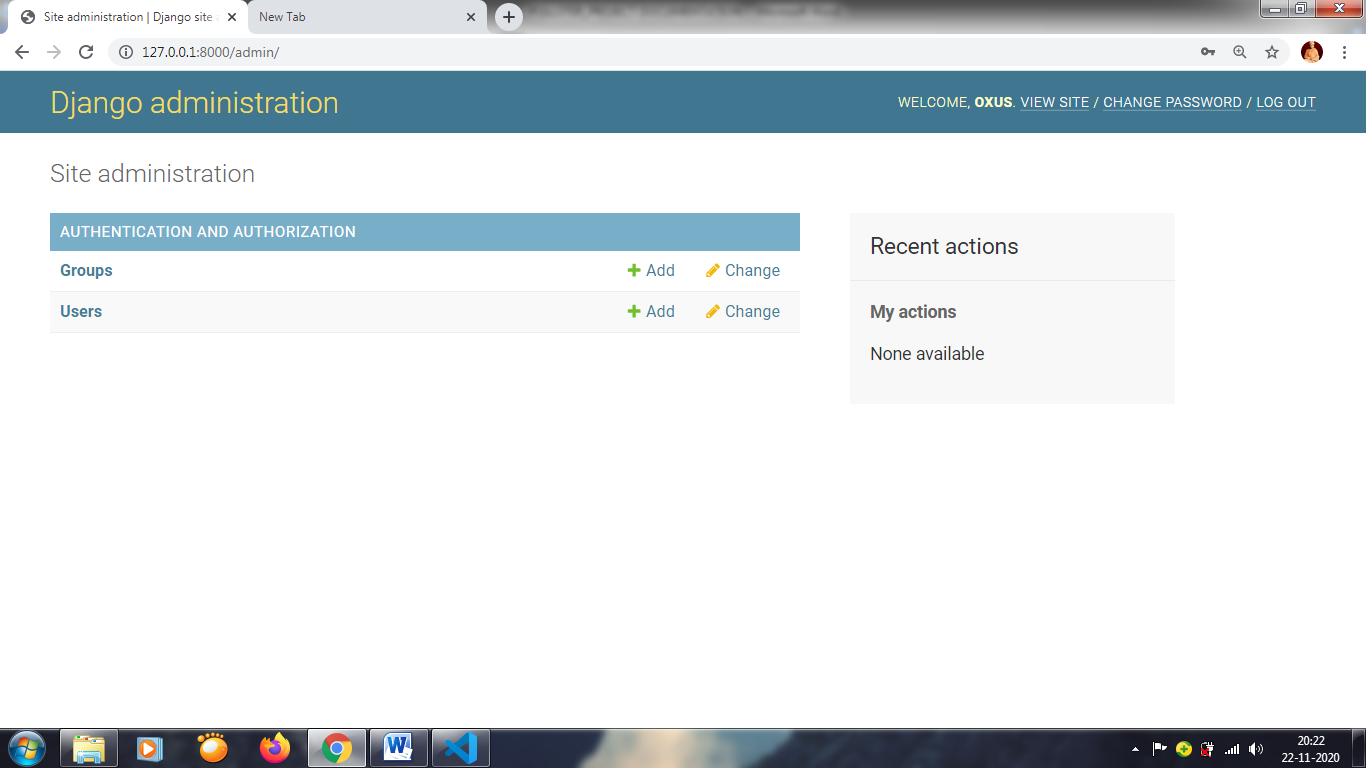
Email address: oxus@gmail.com

Password:oxus

Password (again): oxus

Superuser created successfully.

<http://127.0.0.1:8000/admin/>



**Polls/admin.py**

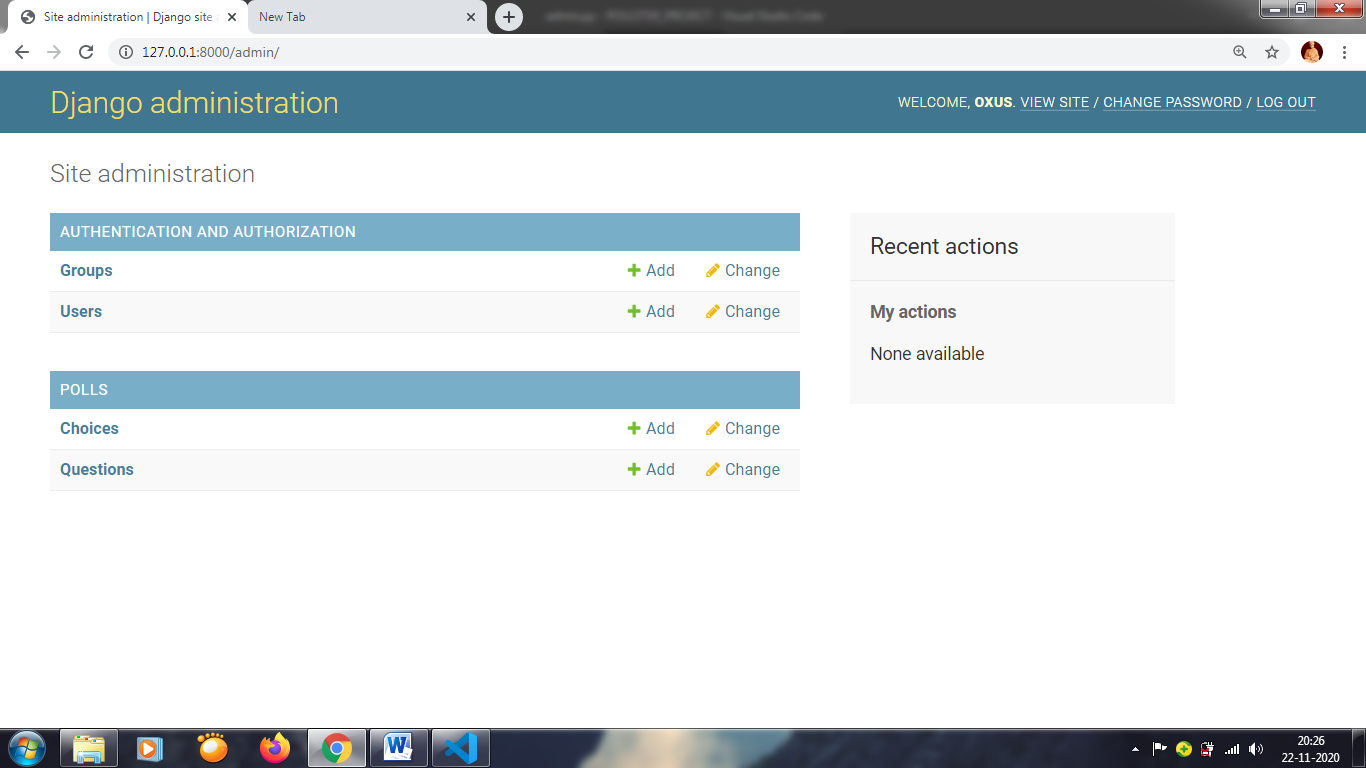
from django.contrib import admin

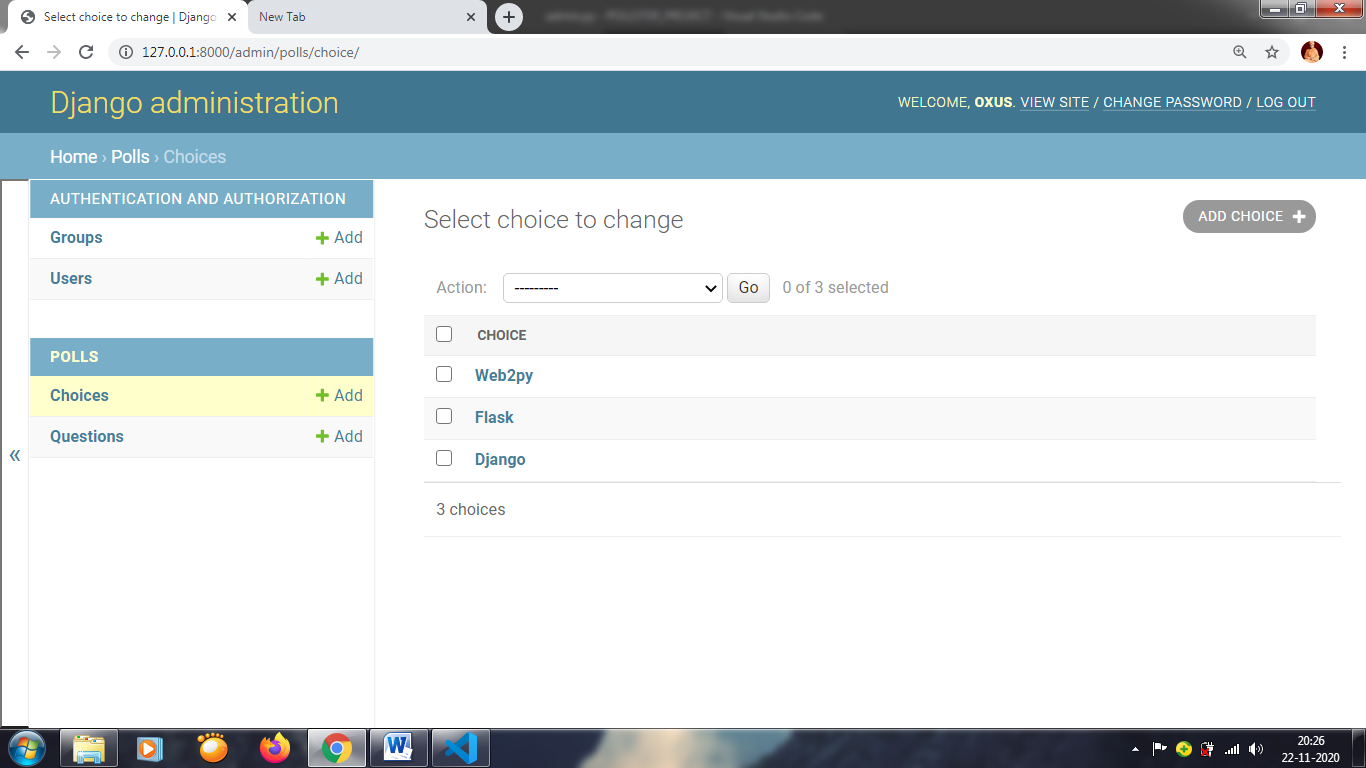
from .models import Question, Choice

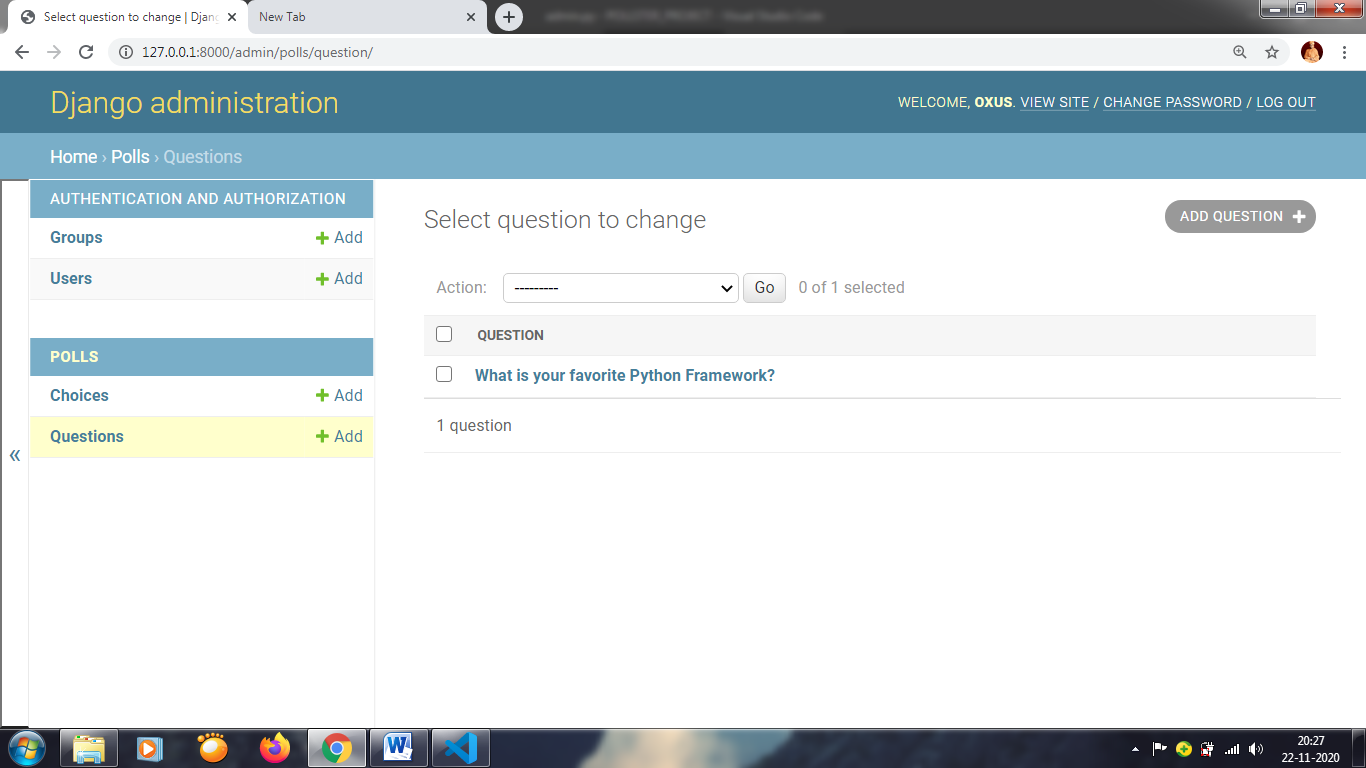
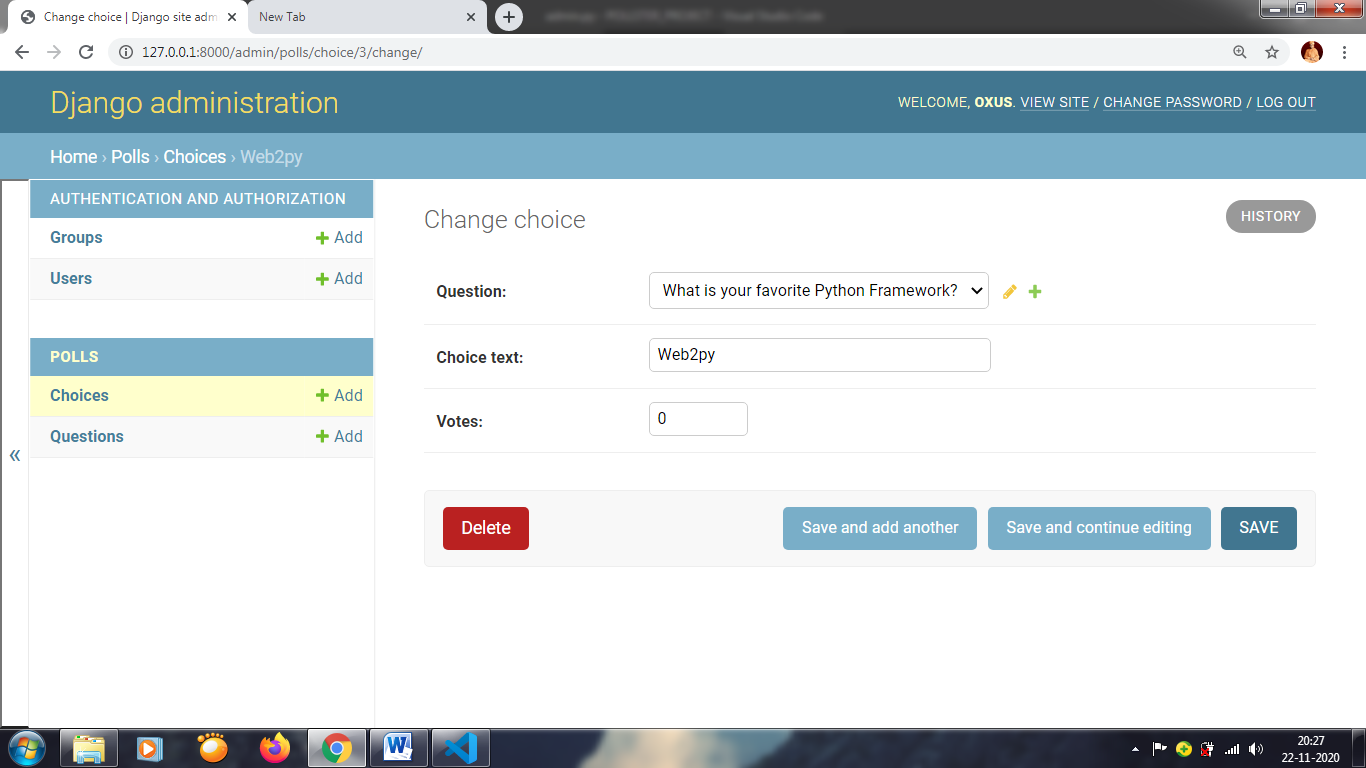
admin.site.register(Question)

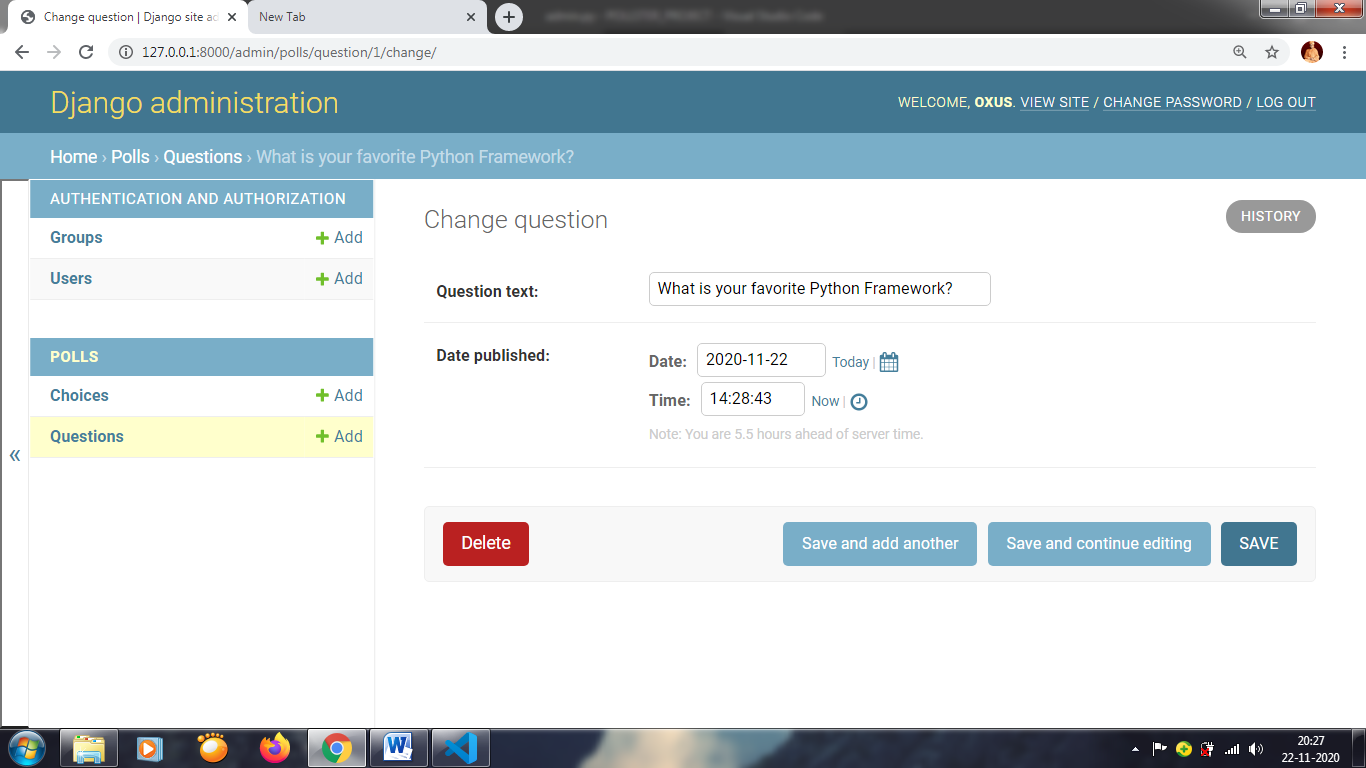
admin.site.register(Choice)

<http://127.0.0.1:8000/admin/>







polls/admin.py

**We want have those choices within Question admin screen. We are going to use tabular inline**

from django.contrib import admin

from .models import Question, Choice

class ChoiceInline(admin.TabularInline):

    model = Choice

    extra = 3

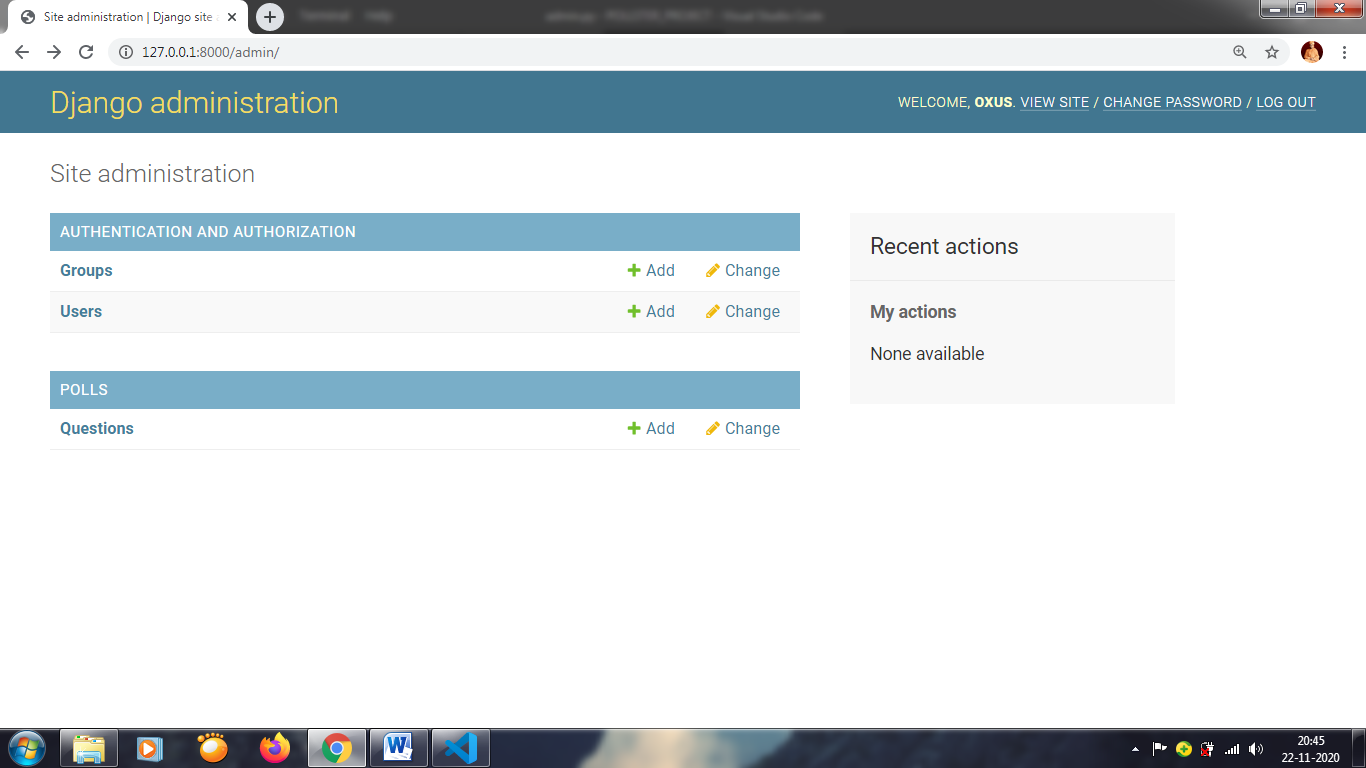
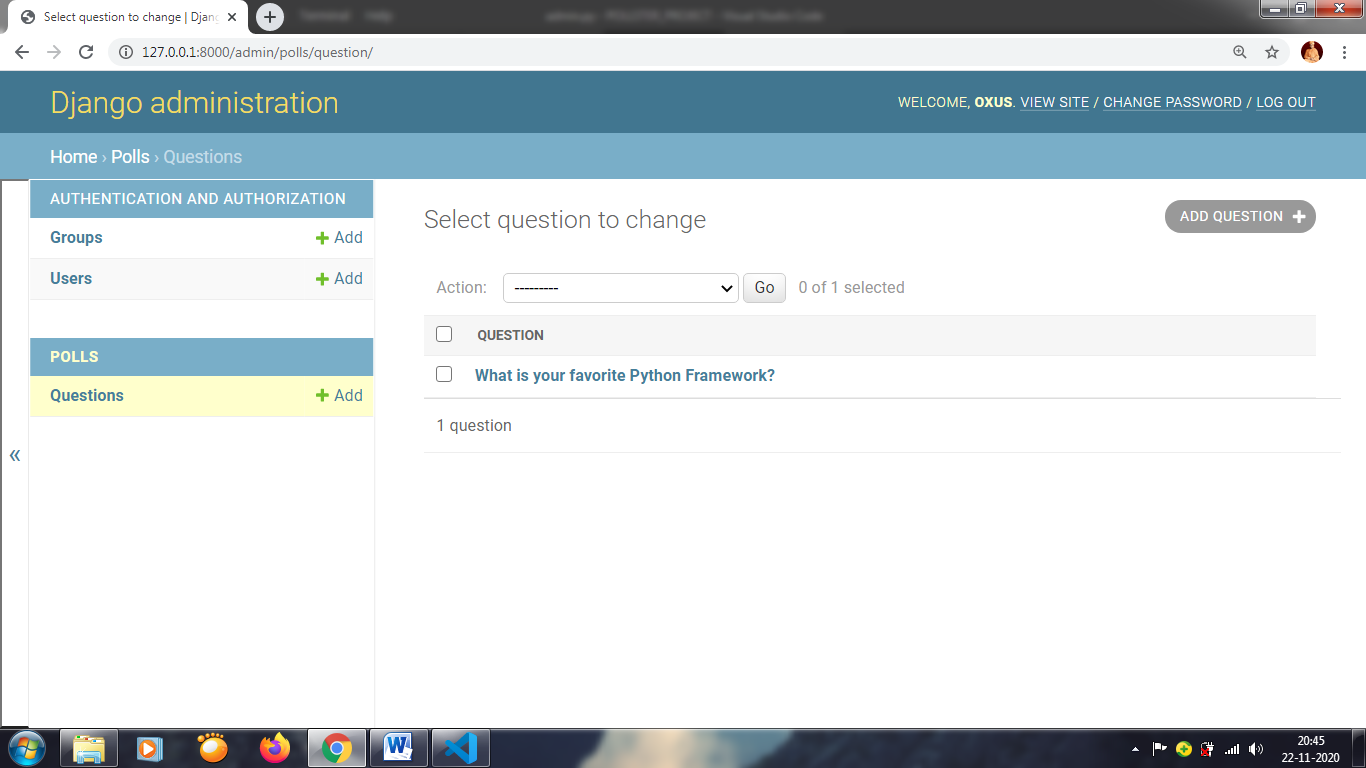
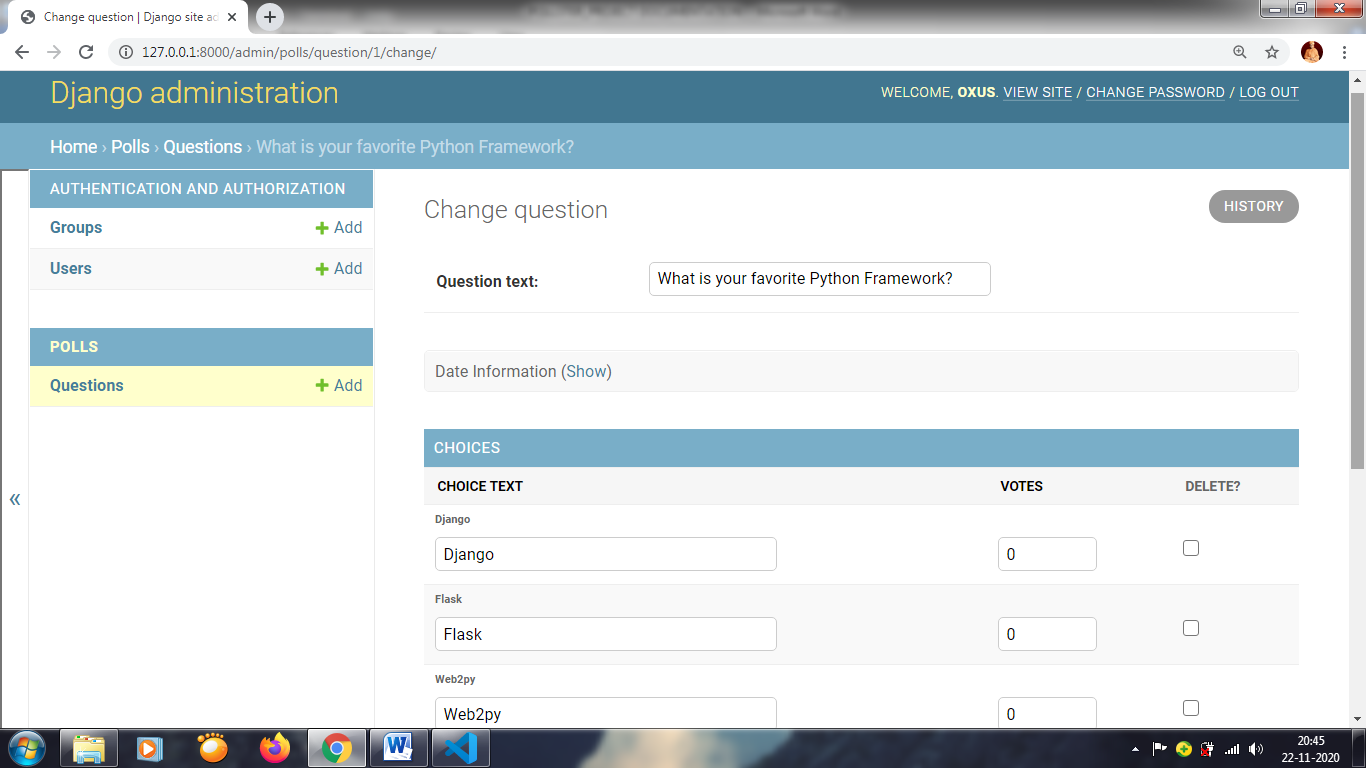
class QuestionAdmin(admin.ModelAdmin):

    fieldsets = [(None, {'fields': ['question\_text']}), ('Date Information', {'fields': ['pub\_date'], 'classes': ['collapse']}), ]

    inlines = [ChoiceInline]

admin.site.register(Question, QuestionAdmin)

[**http://127.0.0.1:8000/admin/**](http://127.0.0.1:8000/admin/)

**pollster/urls.py**

from django.contrib import admin

from django.urls import path, include

urlpatterns = [

    path('admin/', admin.site.urls),

    path('polls/', include('polls.urls'))

]

**polls/urls.py**

from django.urls import path

from . import views

app\_name = 'polls'

urlpatterns = [

    path('',views.index, name='index')

]

**polls/views.py**

from django.shortcuts import render

from .models import Question, Choice

# Get qusetions and display them

def index(request):

    return render(request, 'polls/index.html')

**G:\POLLSTER\_PROJECT\pollster>mkdir templates**

**Create index.html file inside templates**

**templates/index.html**

**pollster/settings.py**

import os

TEMPLATES = [

    {

        'DIRS': [os.path.join(BASE\_DIR, 'templates')],

    },

]