Python Signals

**Post/models.py**

from django.db import models

from django.db.models.signals import post\_save, pre\_save, post\_delete

# sender

class Post(models.Model):

title = models.CharField(max\_length=200)

def \_\_str\_\_(self):

return self.title

# receiver

def after\_post\_saved(sender, instance, \*\*kwargs):

print("Post is saved!")

# to receive a signal, you have to register a receiver function that gets # called when a signal is sent using Signal.connect() method

def after\_post\_deleted(sender, instance, \*\*kwargs):

print("Post is deleted!")

post\_save.connect(after\_post\_saved, sender=Post)

post\_delete.connect(after\_post\_deleted, sender=Post)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**terminal**

path > python manage.py makemigrations

path > python manage.py migrate

path > python manage.py shell

>>> from post.models import Post

>>> post1 = Post.objects.create(title='First title')

Post is saved

>>> post2 = Post()

>>> post2.title = 'Second title'

>>> post2.save()

Post is saved

>>> post1.delete()

Post is deleted

**Main/urls.py**

from django.contrib import admin

from django.urls import path

from post import views

urlpatterns = [

path('admin/', admin.site.urls),

path("", views.home, name="home-page")

]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**post/views.py**

from django.shortcuts import render

from django.http import HttpResponse

from django.dispatch import receiver

from django.core.signals import request\_finished

# request\_finished

# Create your views here.

def home(request):

return HttpResponse("Page is loaded!")

@receiver(request\_finished)

def home\_request\_finished(sender, \*\*kwargs):

print("Request finished!")