***Karim Rivera Litardo***

*Curso: Desarrollo de software*

*3H*

*MATERIA: PROGRAMACION VISUAL*

*TEMA: GIT-GITHUB*

*Profesor: CARLOS LUIS PAZMIÑO PALMA*

*TALLER # 1*

Los temas a desarrollar son los siguientes:

1. **¿Qué es Django?**

Es un framework de desarrollo web de código abierto, escrito en Python, que respeta el patrón de diseño conocido como modelo–vista–controlador.

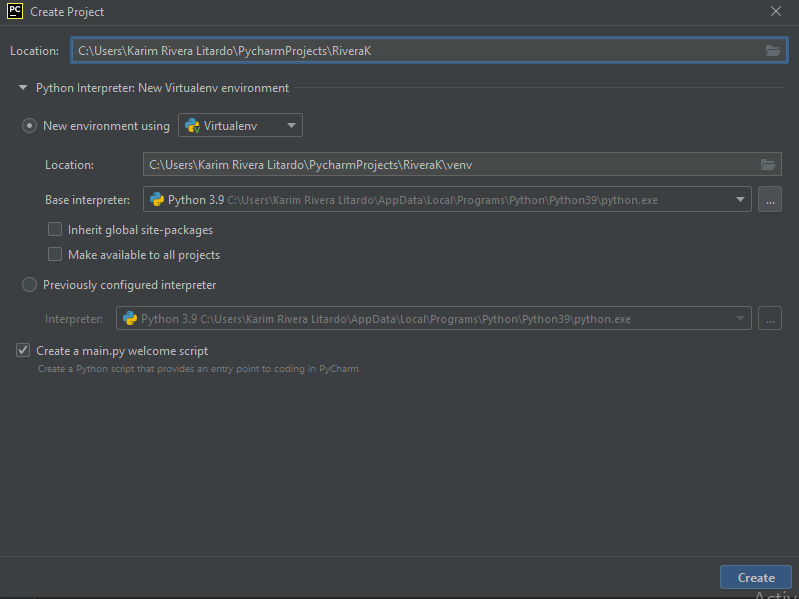
1. **¿Qué es la máquina virtual en Django?**

Es la que permite trabajar en el proyecto con Django

1. **¿Qué es MVT en Django?**

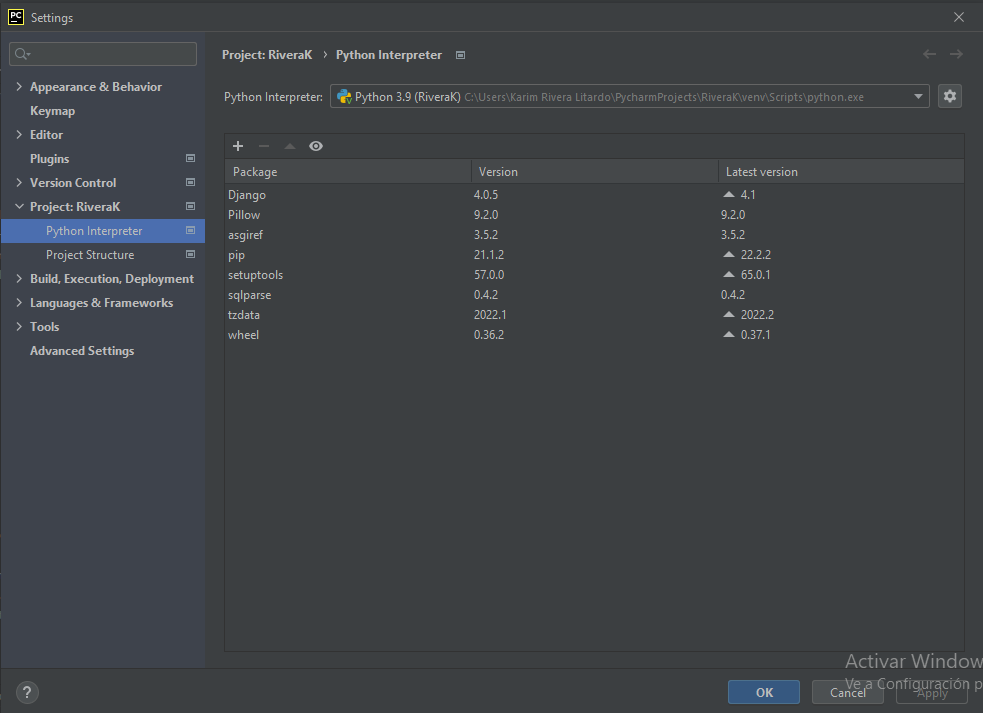
Es una forma de que este conectado el directorio templates al views y al url del Core

1. **Crear un proyecto con la máquina virtual.**

****

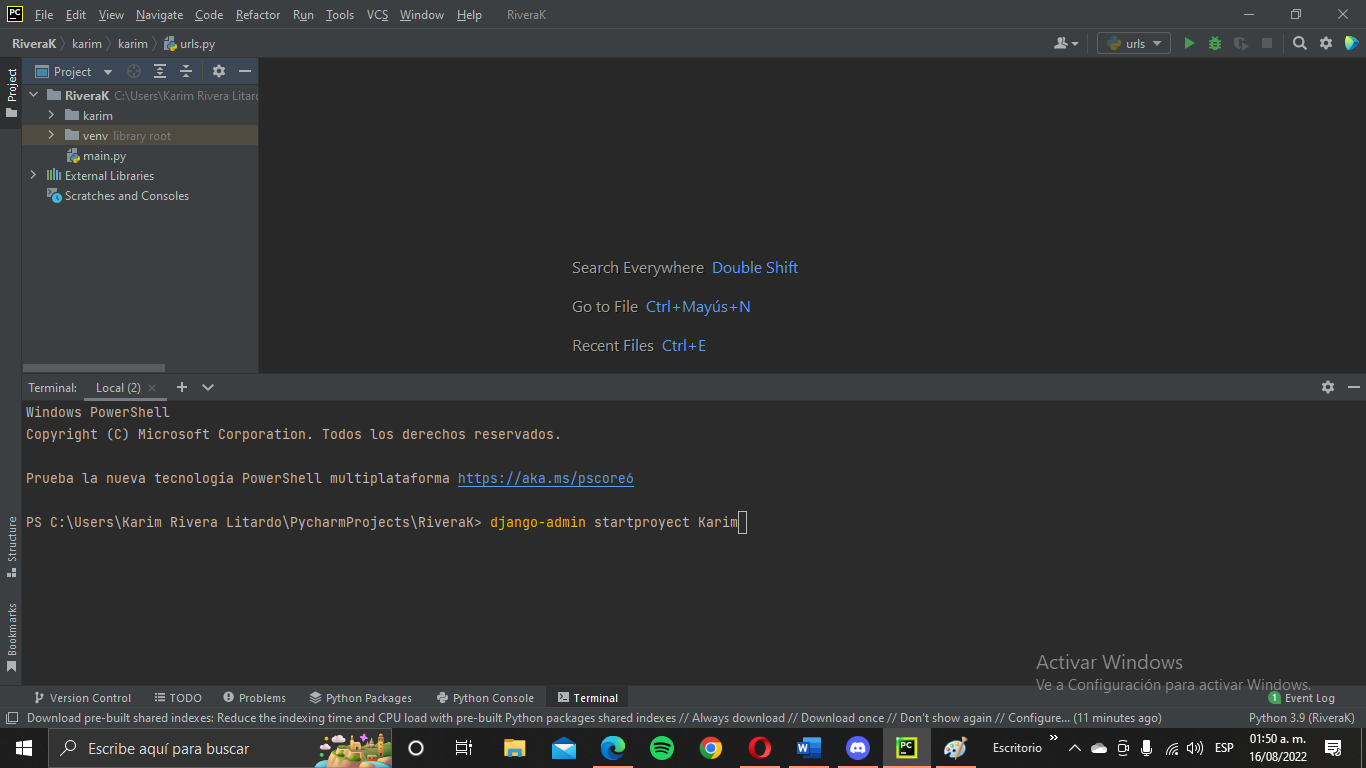
En esta imagen se ve como se realizo el proceso de crea un proyecto con la maquina virtual

1. **Descargar los instaladores de Django al proyecto**

****

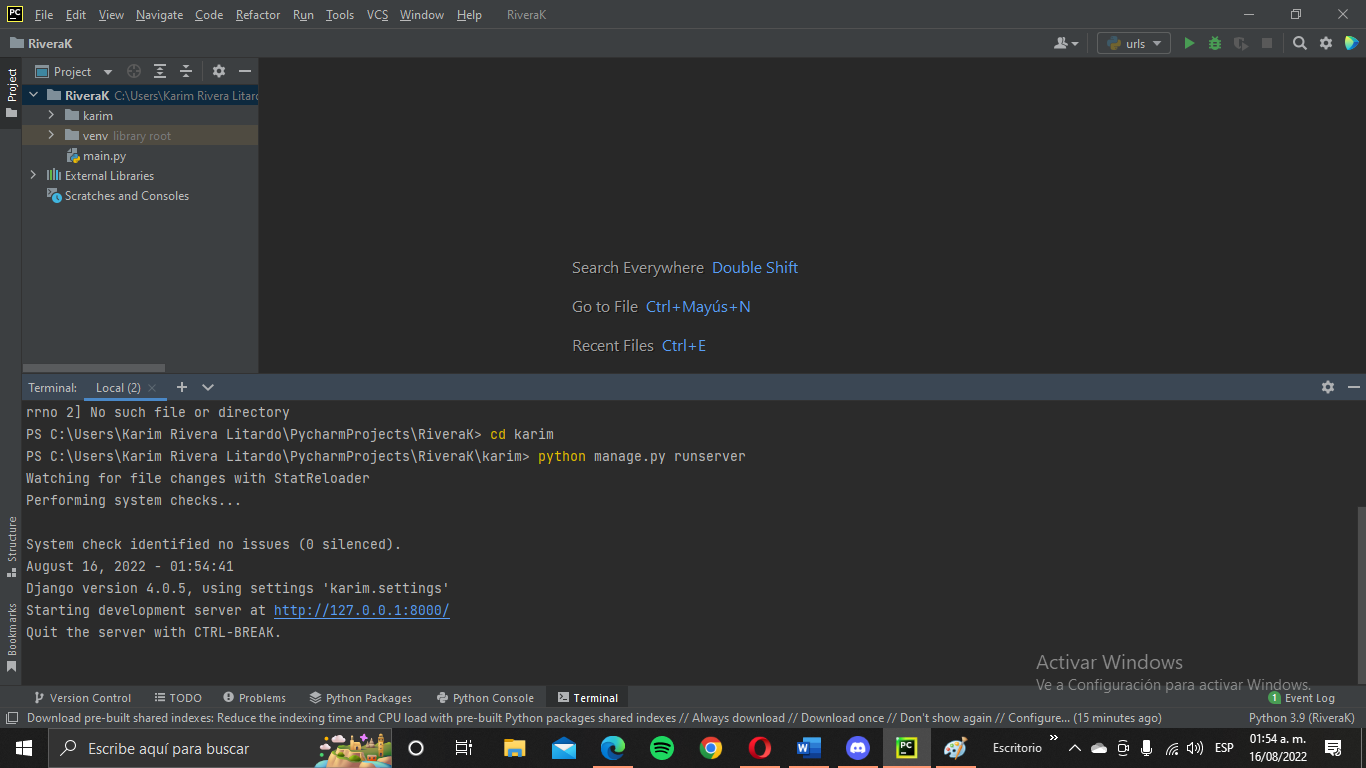
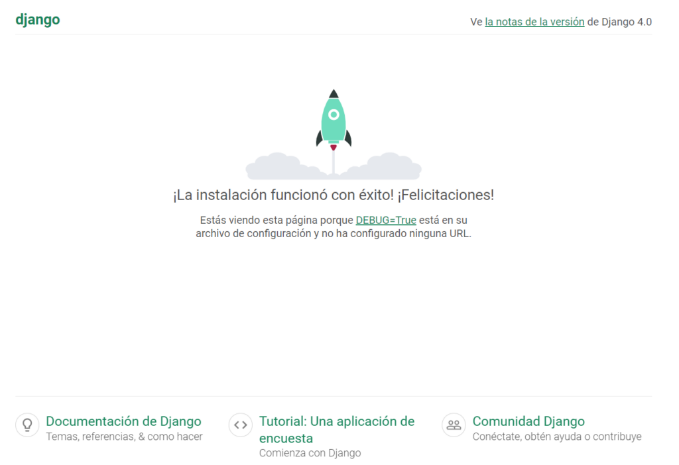
Se procedió a descargar el paquete de Django y de paso también instalar el paquete pillow el cual servirá para mas tarde

1. **Crear un proyecto para programar en Django**



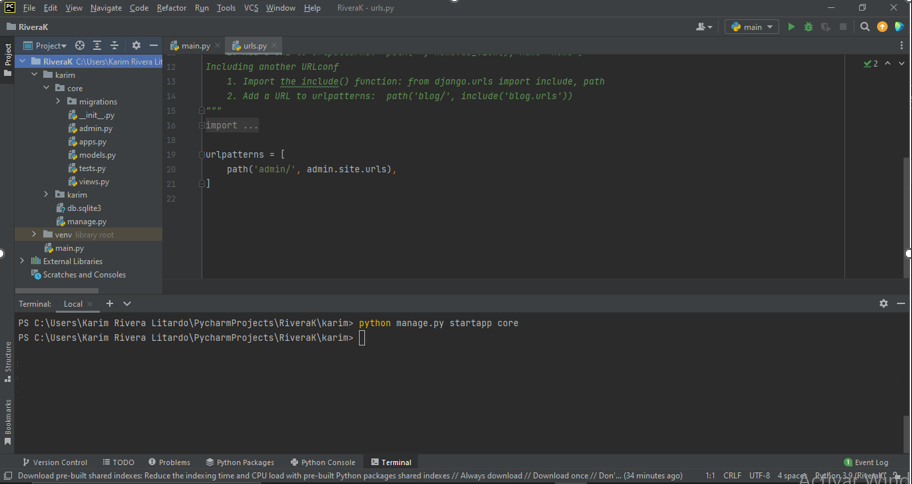
Para crear el proyecto en Django se procedió a crear otro proyecto dentro del original, pero con diferente nombre esto utilizando el comando django-admin startproyect (El nombre que se le quiera poner)

1. **Ejecutar el proyecto y el mensaje de felicitaciones.**

Para poder ejecutar el programa se necesitó un comando para llamar al proyecto que posee el archivo manage, el cual es cd (el nombre del proyecto) y luego se pone el comando Python manage.py runserver para que se pueda ejecutar

1. **Crear una Apps core.**

****

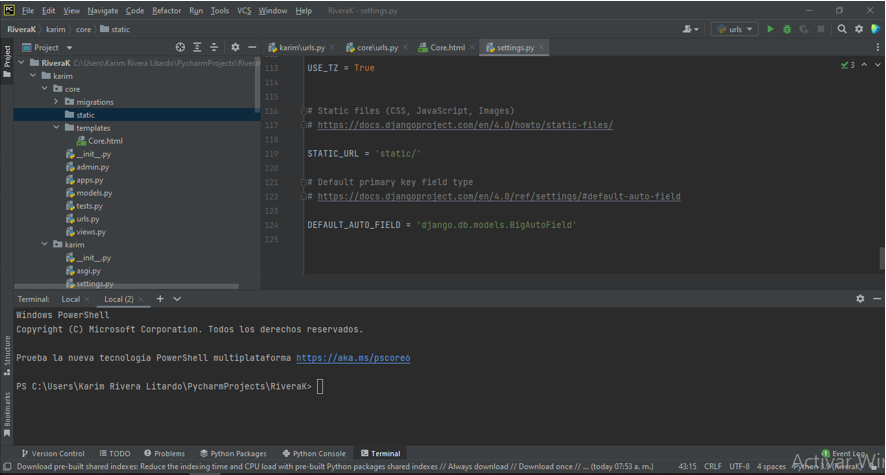
Para crear la Core se utilizo un comando el cual es Python manage.py startapp core lo cual hace que se cree el directorio con su migrations y archivos Python correspondientes

1. **¿Qué es la Carpeta Templates?**

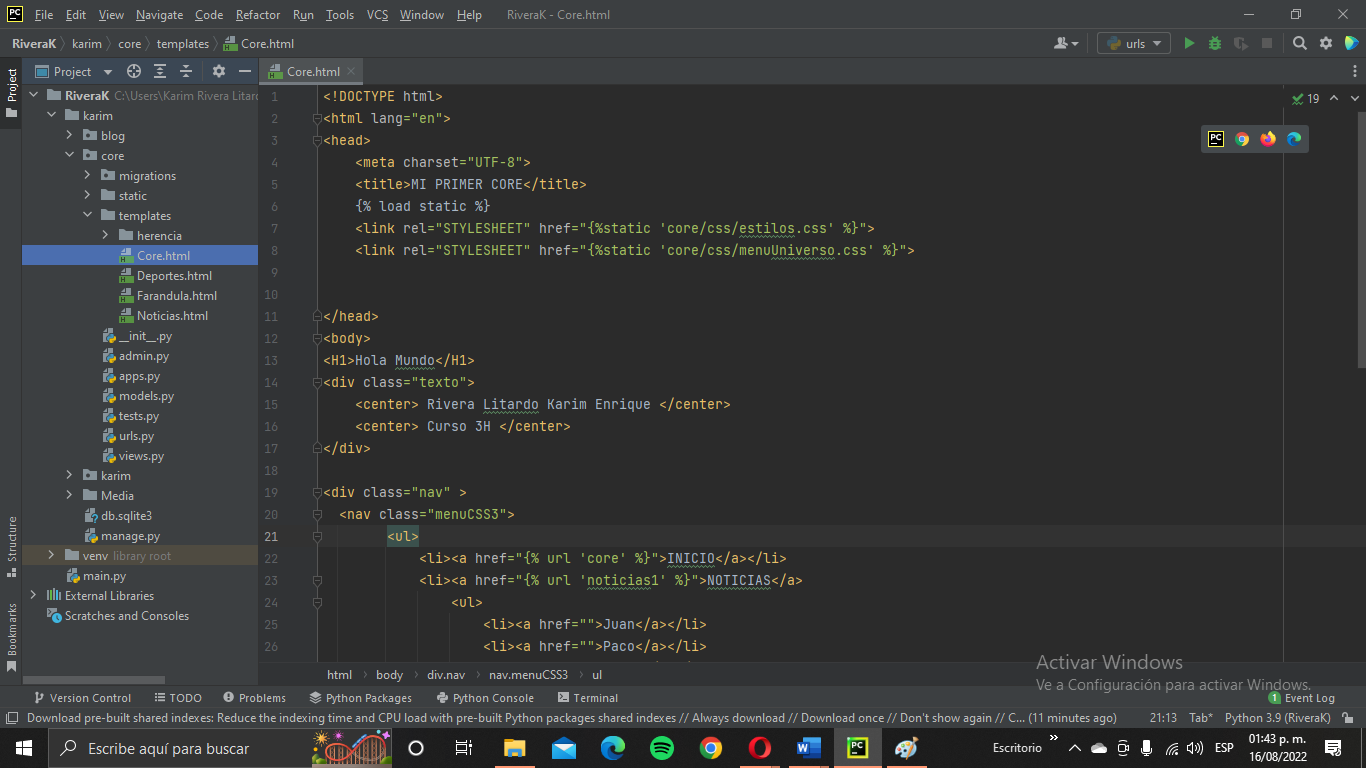
Es una carpeta en la cual se pondrán todos los html que se vayan creando

1. **¿Qué es la Carpeta stactic?**

Es una carpeta en la cual se guardan los archivos css y img que se van a utilizados en el proyecto



1. **Crear un archivo base html en la APPS core**

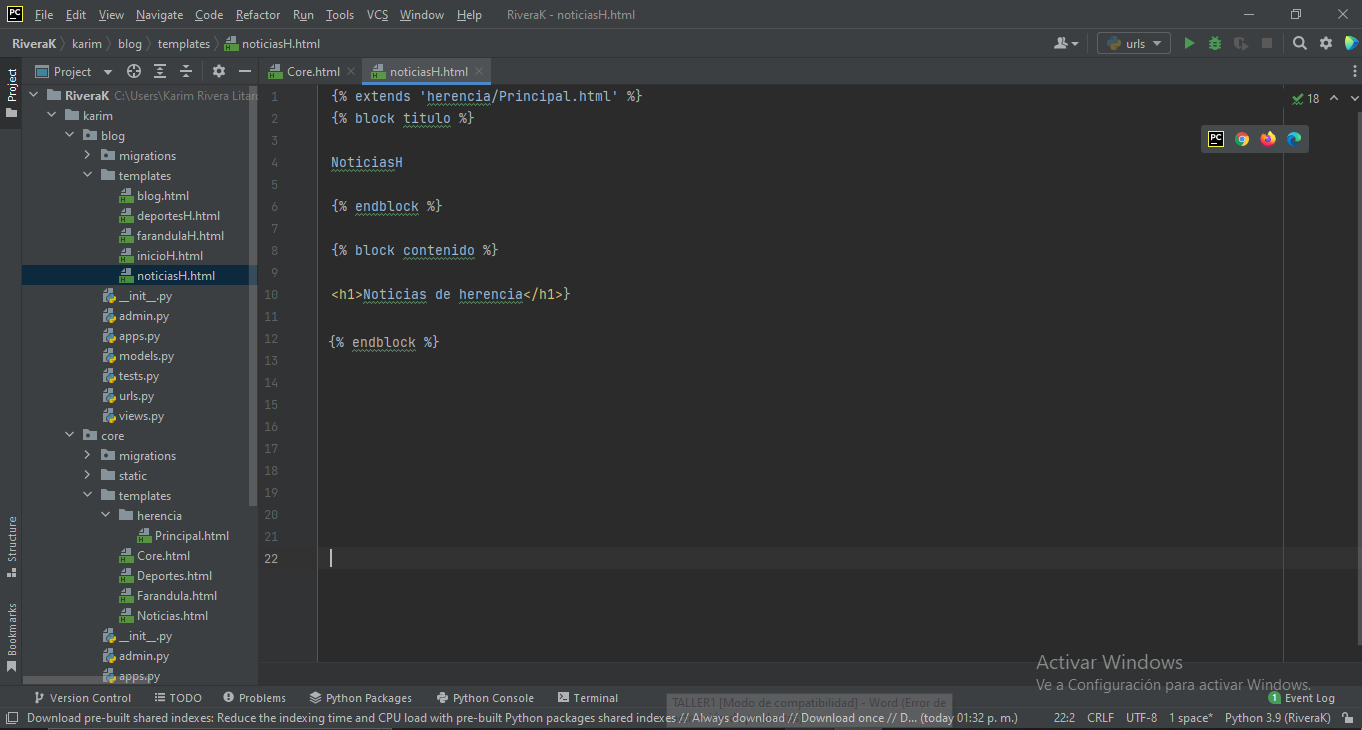


En esta parte se puede ver como se realizo una parte del html que se encuentra en el core y a este html se lo llamo core.html

1. **Como se llaman a los CSS desde el archivo base html.**

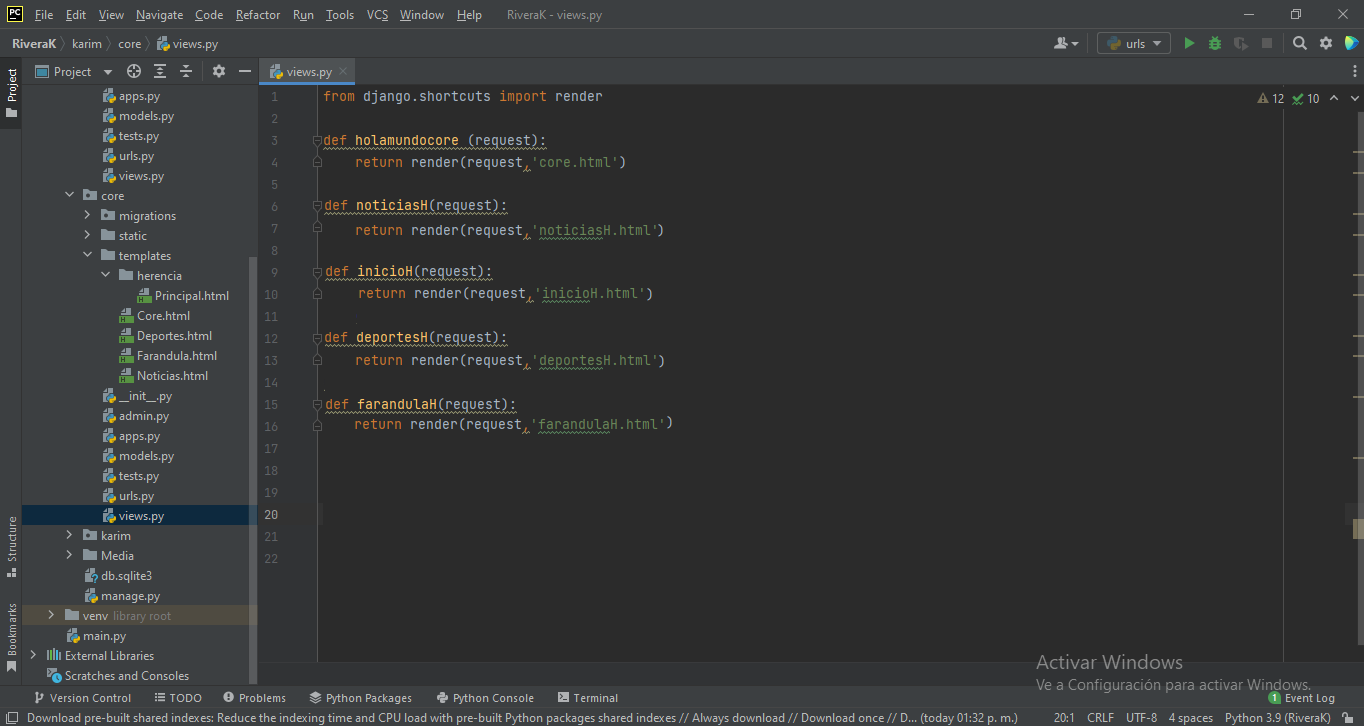
La forma de llamar al css es con este código (<link rel="STYLESHEET" href="{%static 'core/css/estilos.css' %}">), lo que tiene que ponerse es la carpeta static y también a la subcarpeta llamada core y dentro esa se encontraría el css y por último poner el nombre correspondiente del archivo

1. **Como consume un archivo hijo html al utilizar la herencia del archivo base html.**

****

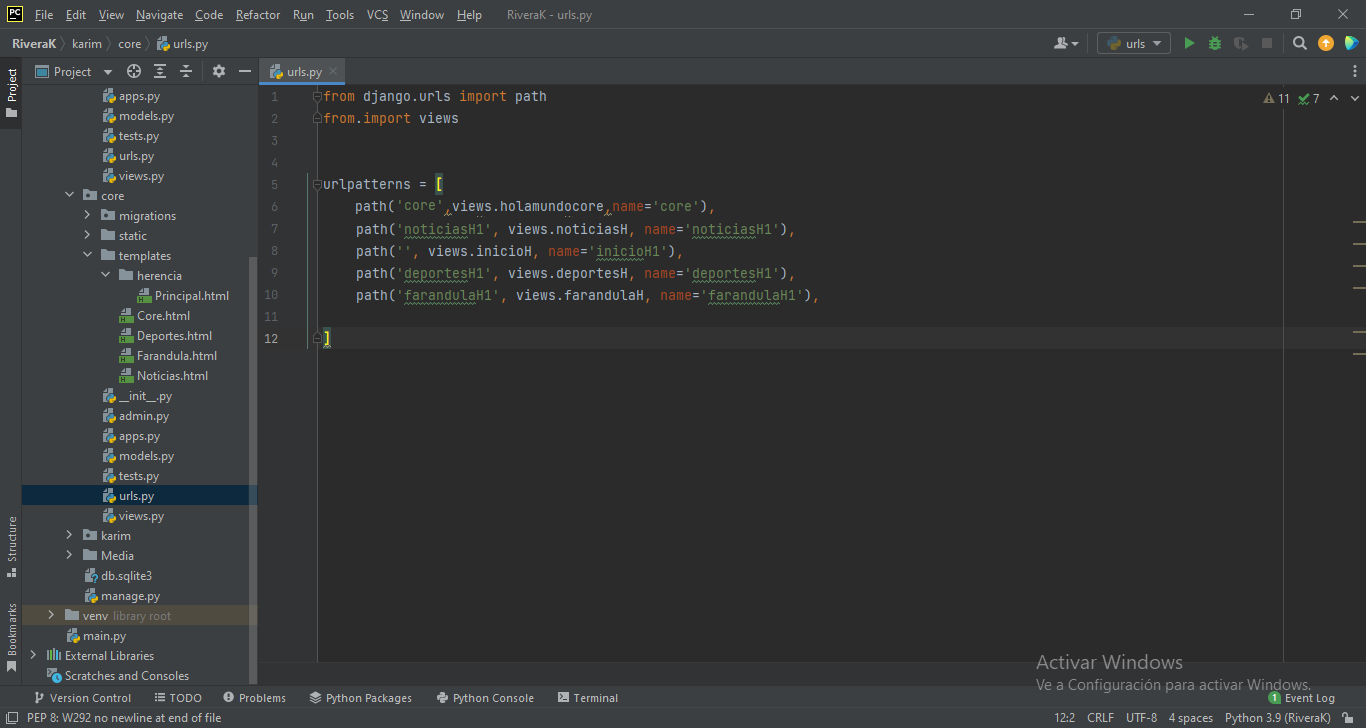
Para que el archivo hijo utilice el archivo base se necesita la extensión que lo llame utilizando {% extends 'herencia/Principal.html' %} con su respectivo block titulo y block contenido, el cual hacen que lo que se coloque dentro del bloque le pertenezca al html

1. **Crear un view que llame al html hijo**

****

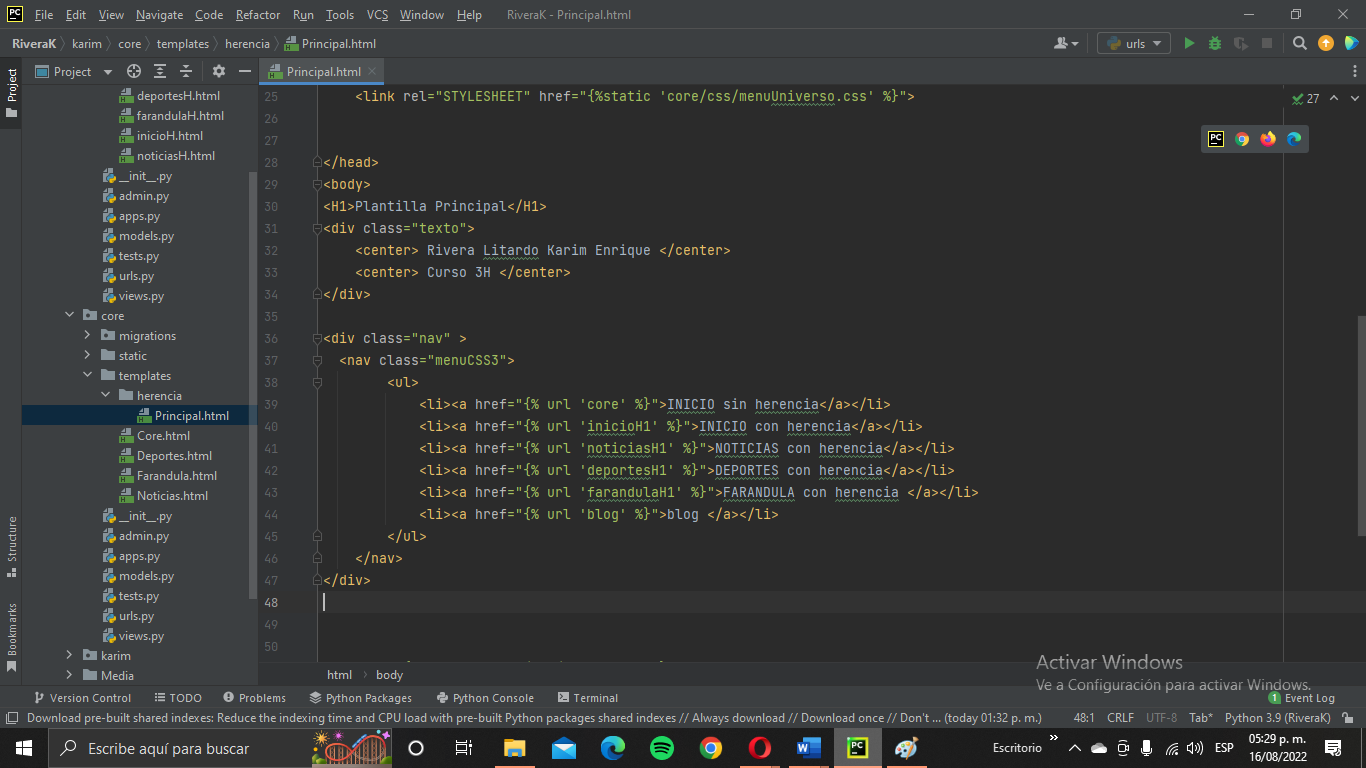
En este apartado se creo los views de los diferentes html que tienen herencia

1. **Crear la urls que llame al views.**

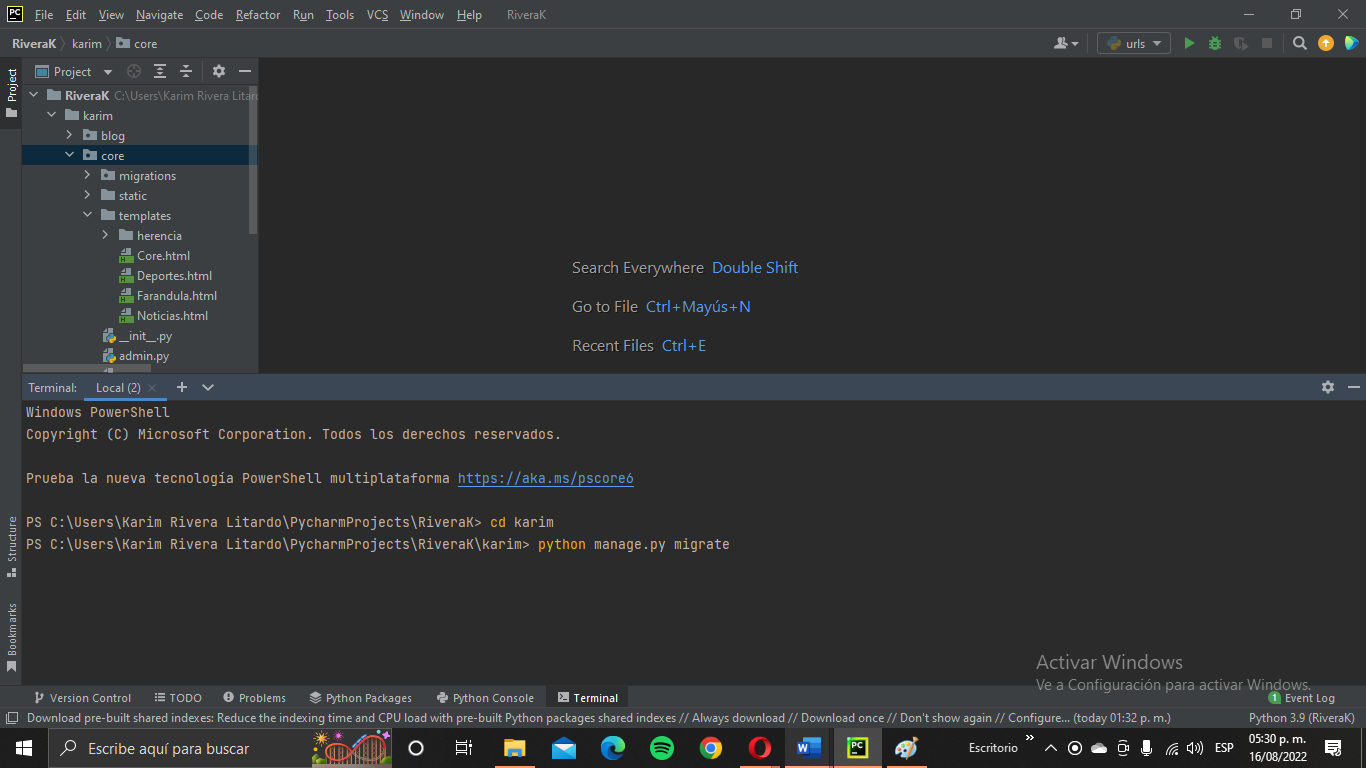
****

Al igual que el views en el urls es el mismo procedimiento, pero estos estáran conectados con el views

1. **Integrar la aplicación APPS core al proyecto principal**

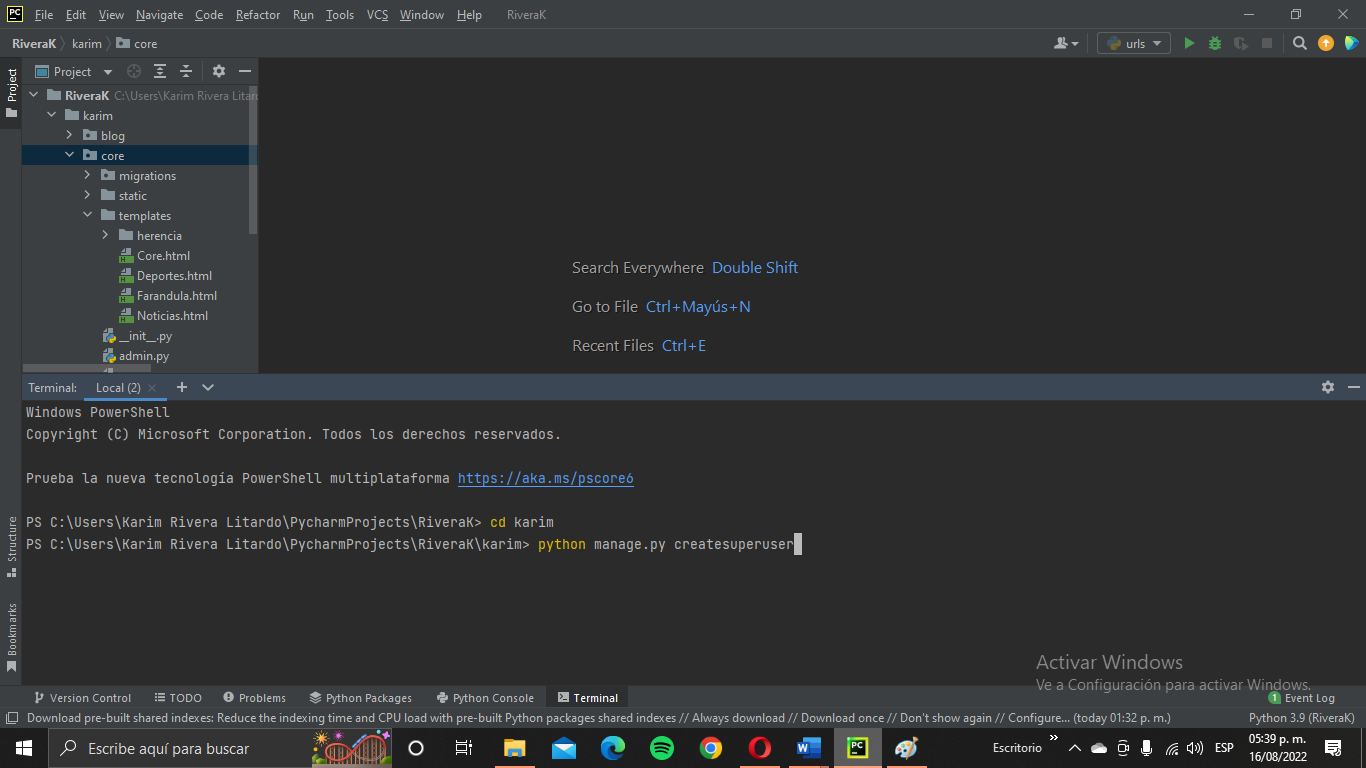


1. **Crear las tablas del sistema de usuarios para utilizar el panel de administración.**

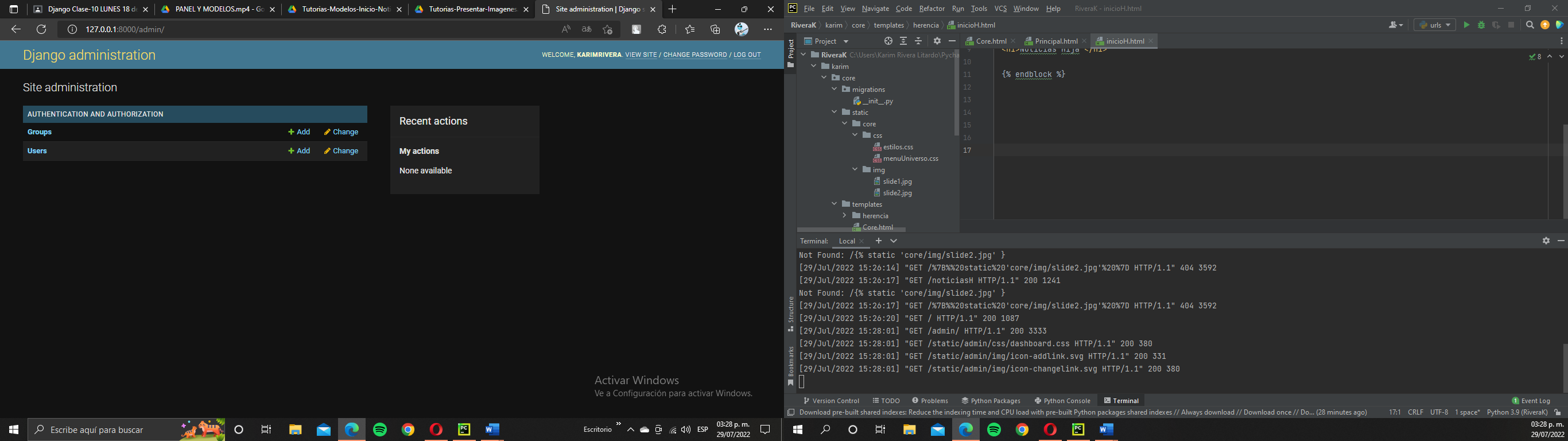


Para crear y migrar la base de datos se necesita el código Python manage.py migrate una vez ejecuto saldrá que se migro correctamente

1. **Crear un usuario para poder ingresar al Panel de Administración**



Para poder crear un usuario para ingresar en el panel de admin se necesita poner en el terminal el código Python manage.py createsuperuser al ejecutarlo pidara que se ingrese un correo, nombre de usuario y una contraseña, luego de ejecutar el programa, se ingresan los datos y se visualiza el panel de admin

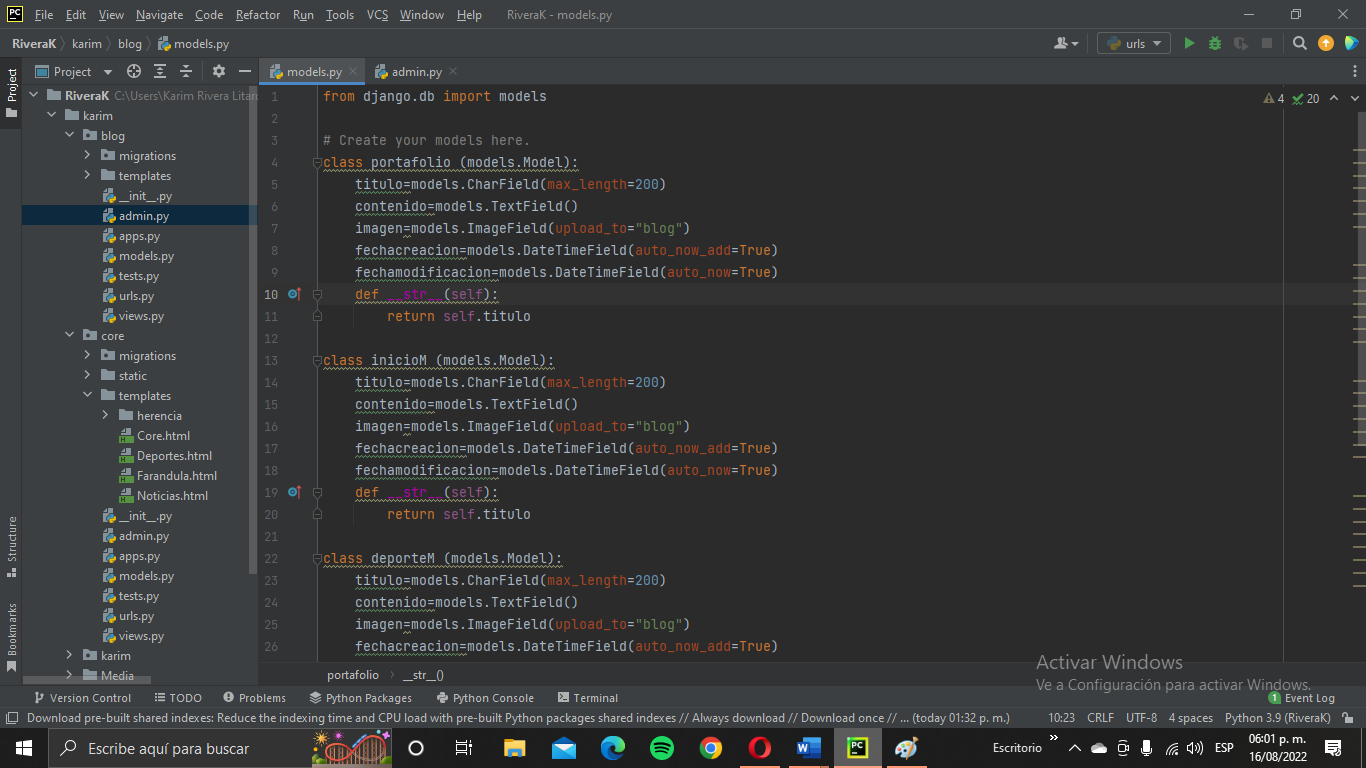


1. **Que es un modelo en Django**

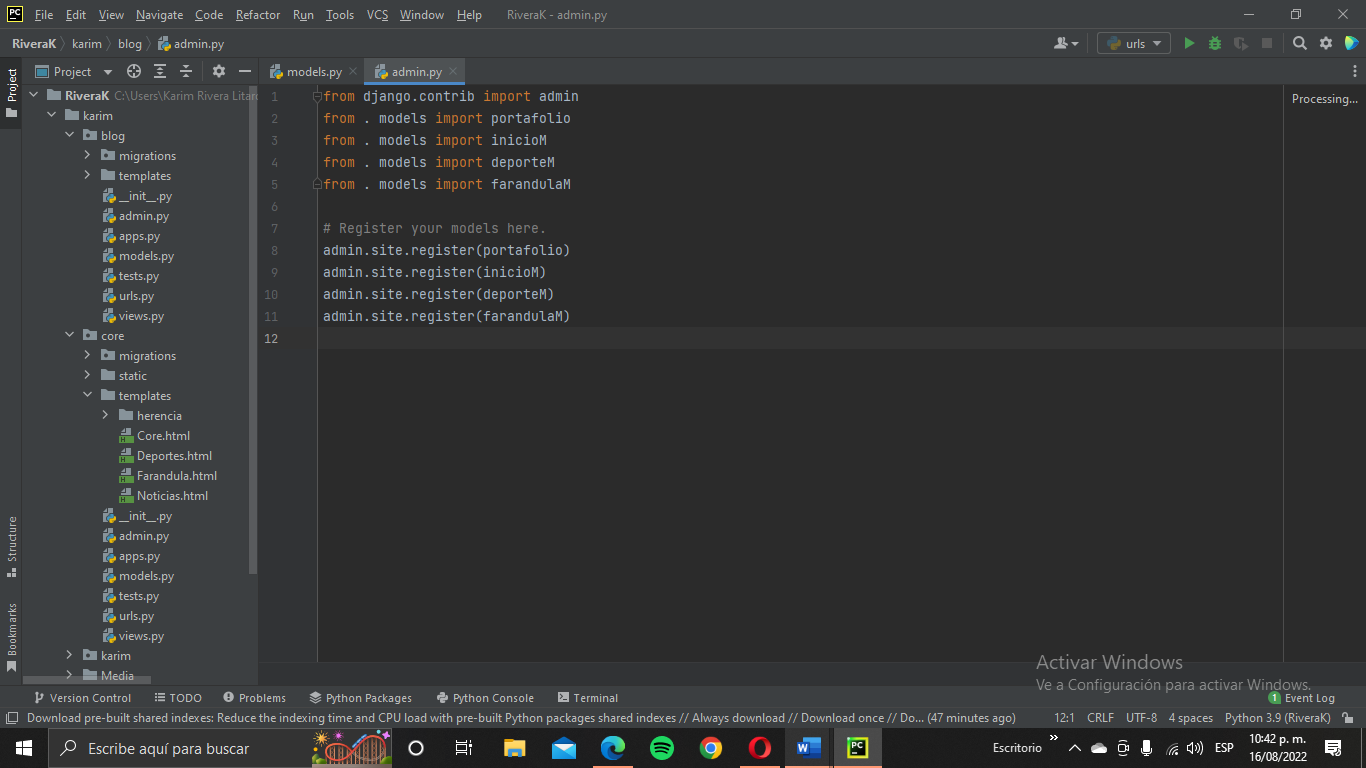
Es un tipo especial de objeto que se guarda en la base de datos

1. **Crear un modelo en Django.**

Antes de hacer el modelo hay que hacer un directorio llamado blog utilizando el código Python manage.py startapp blog



Para poder crear un modelo se coloca un class y el nombre que tendrá el modelo junto a los contenidos que quiero que tenga como título, contenido, imagen, fecha de creación y fecha de modificación



Esto va unido con el apartado admin el cual hace que se registre con su respectiva importación.

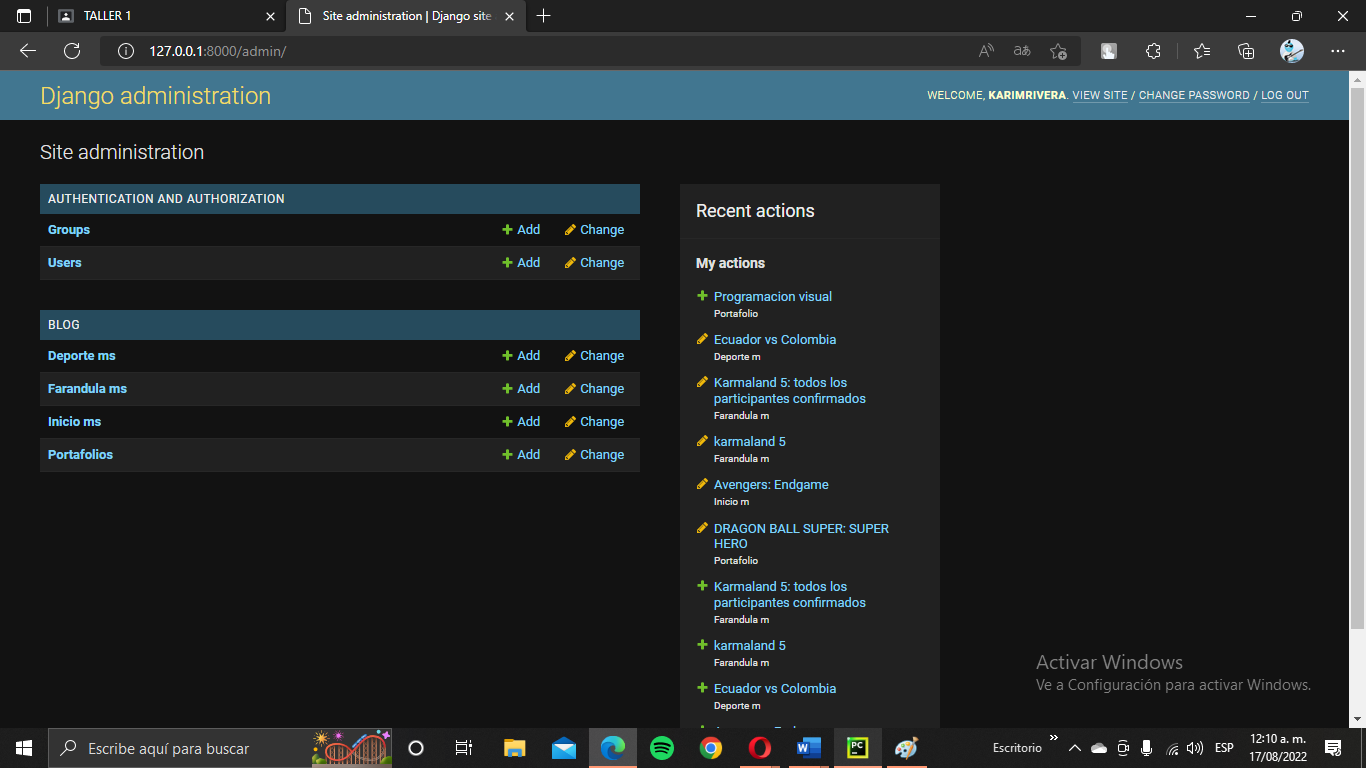
1. **Migrar el Modelo a la base del Panel de Administración.**

Para poder migrar el modelo a la base de datos se necesitan de 2 códigos

1. python manage.py makemigrations blog
2. python manage.py migrate blog

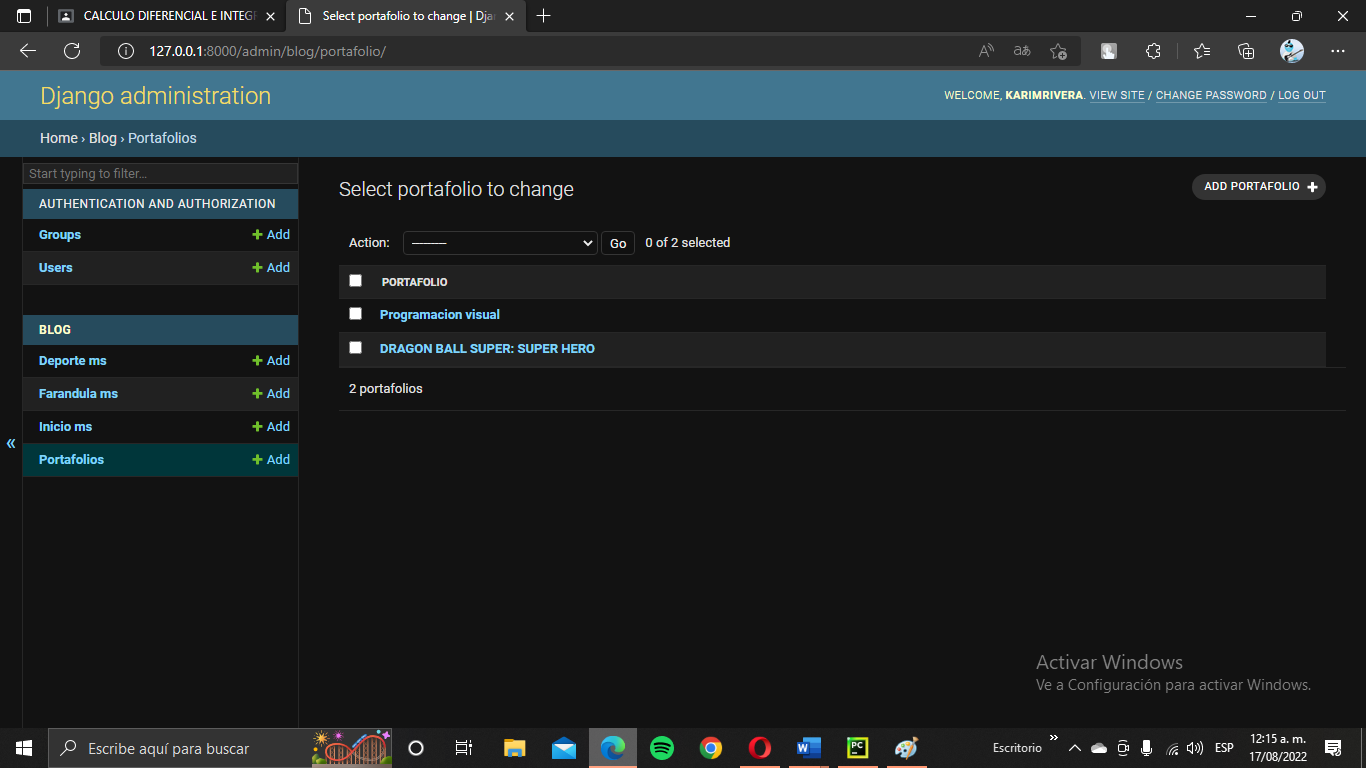
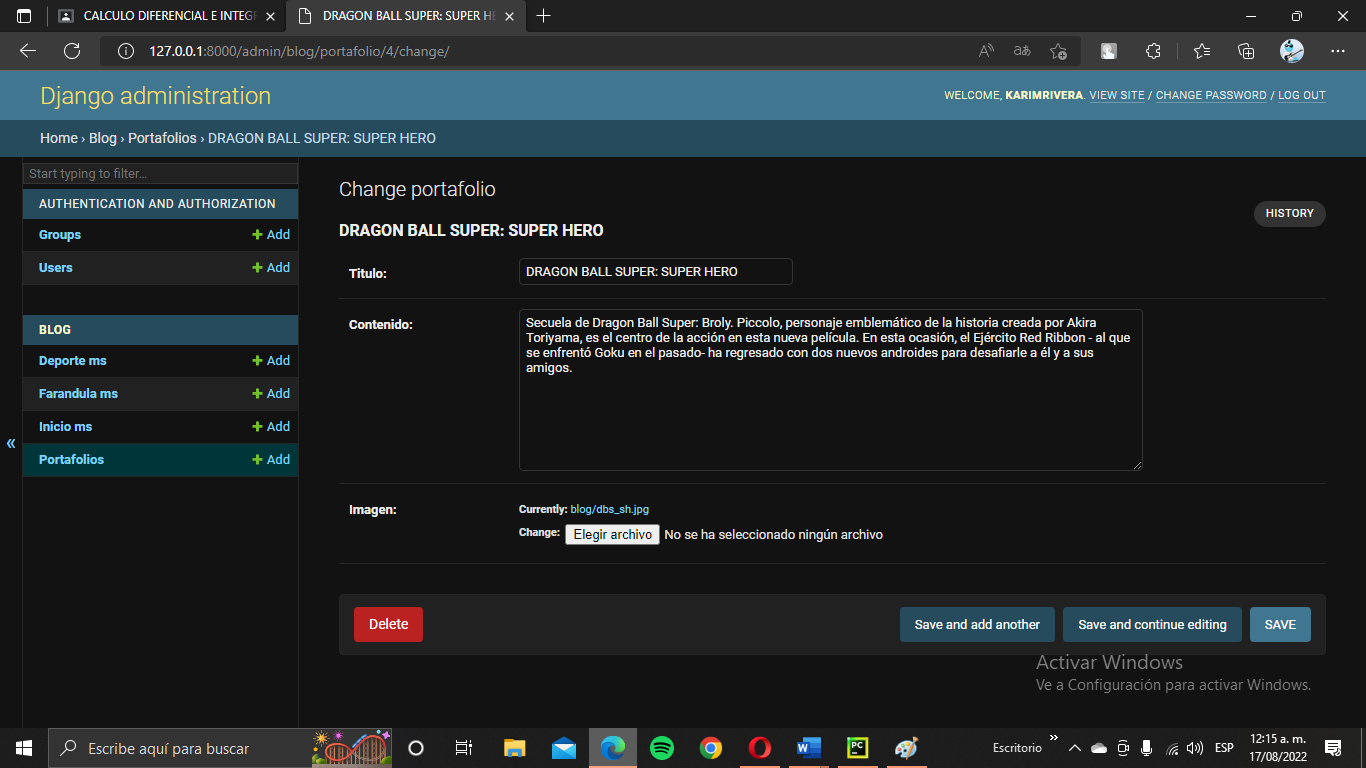
estos dos harán que se muestre los modelos creados en el panel de administración

1. **Integrar el Modelo al Panel de Administración.**



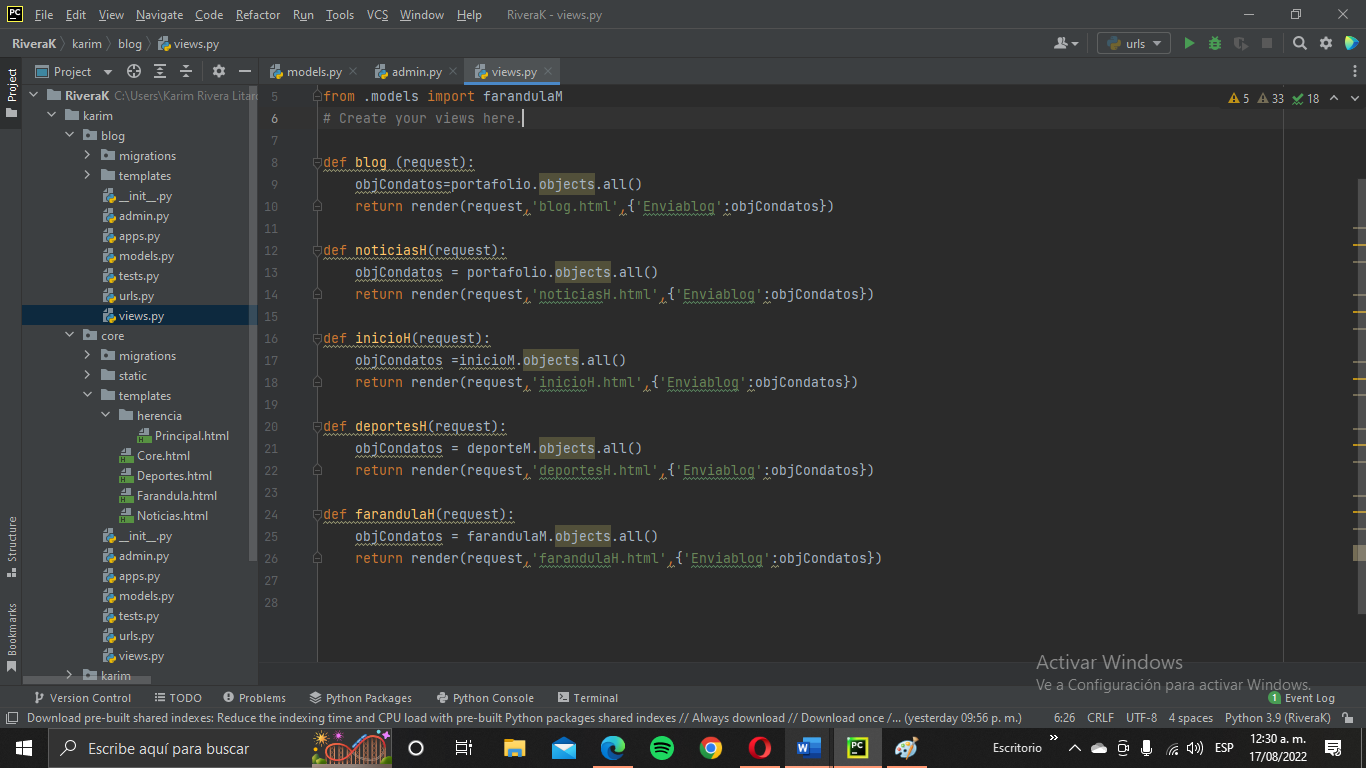
**Aquí se puede visualizar como todos los modelos que se crearon ya se ven en panel de administración**

1. **Ingresar información al modelo por el Panel de Administración.**

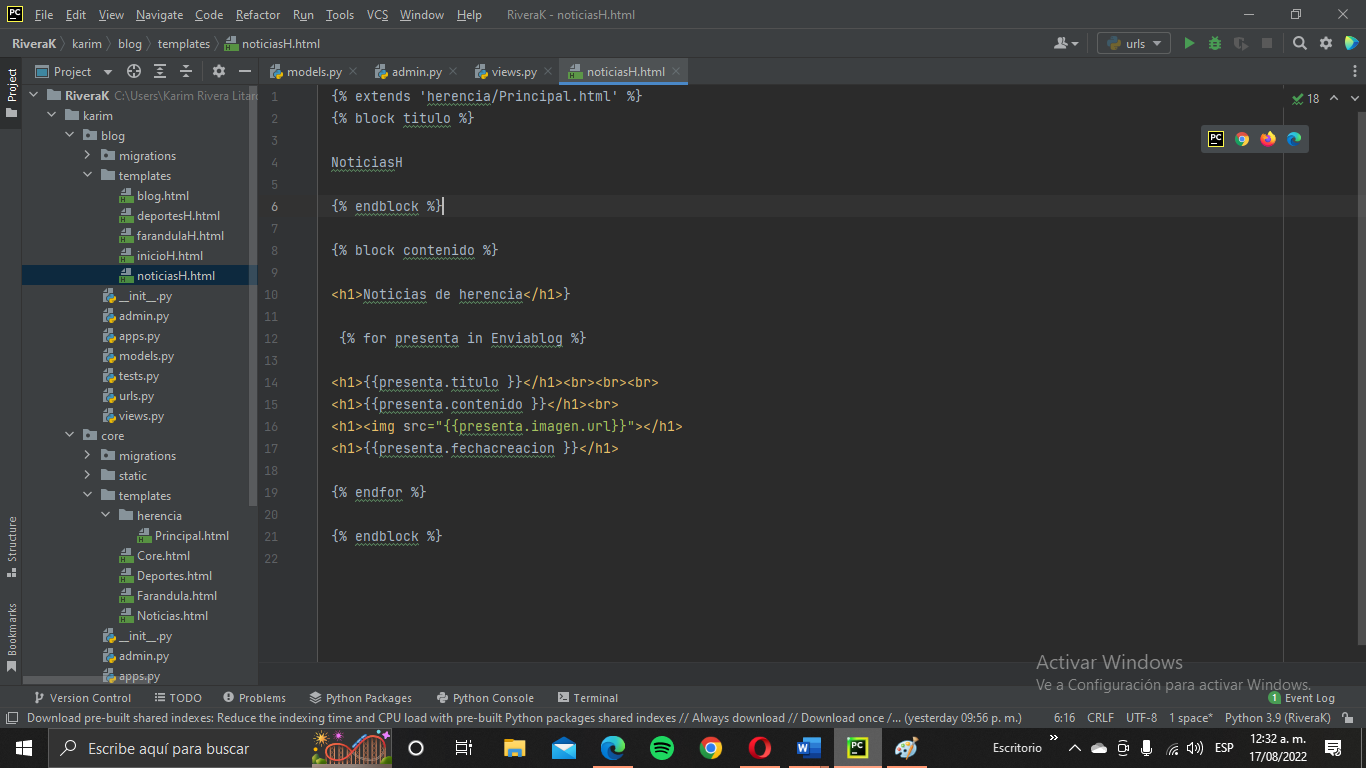
En este apartado se añade una nueva ventana de información en la cual se le agregará información la cual se proyectará en el proyecto principal, en este se puede ver como tiene lo que se coloco en el modelo como el título, contenido, imagen y fecha

1. **Realizar la consulta de todo lo ingresado en el modelo desde el views.**

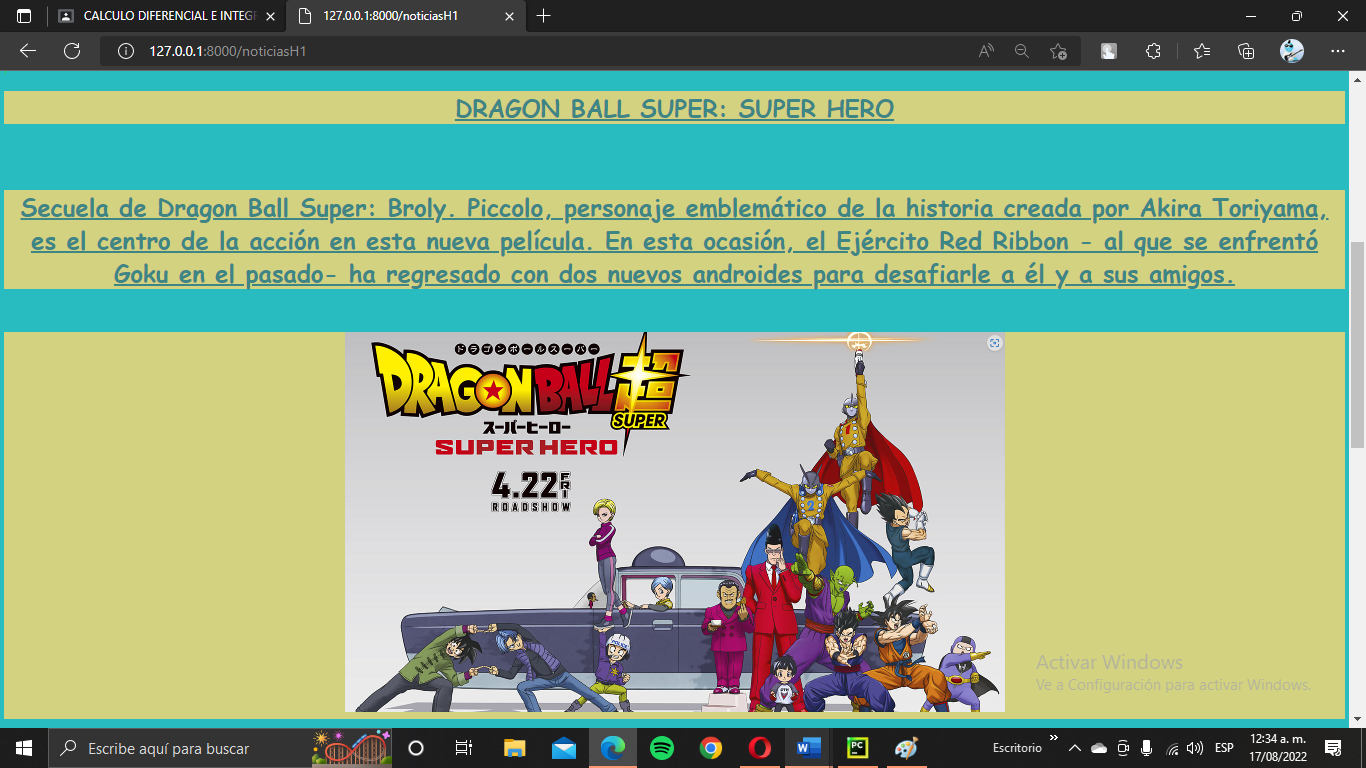


En esta parte se puede visualizar como esta todo estructurado en el views para que este conectado y no haya errores

Y adicional a eso cada html tiene que tener un for para que llame del panel de administración y lo proyecte en el proyecto



1. **Mostrar los datos guardados en el modelo al html hijo**



**Y por ultimo se ve como todo lo que se puso en el panel de administración se ve en el proyecto final, se puede observar cómo se ve el css, la imagen los títulos.**