CARRERA TECNOLOGÍA SUPERIOR EN DESARROLLO DE SOFTWARE

**ASIGNATURA:** PROGRAMACION VISUAL

**TEMA:** GIT-GITHUB

**DOCENTE:** ING. CARLOS PAZMIÑO

**NOMBRE ESTUDIANTE:** DIEGO ANDRÉS VILLAO GODOY

**PARALELO:** 3 “G”

****

**INDICES**

Contenido

[1. ¿Qué es Django? 4](#_Toc111544994)

[2. ¿Qué es la máquina virtual en Django? 4](#_Toc111544995)

[3. ¿Qué es MVT en Django? 4](#_Toc111544996)

[4. Crear un proyecto con la máquina virtual. 5](#_Toc111544997)

[5. Descargar los instaladores de Django al proyecto 5](#_Toc111544998)

[6. Crear un proyecto para programar en Django 5](#_Toc111544999)

[7. Ejecutar el proyecto y el mensaje de felicitaciones. 6](#_Toc111545000)

[8. Crear apps core 6](#_Toc111545001)

[9. ¿Qué es la carpeta “Templates”? 7](#_Toc111545002)

[10. ¿Qué es la Carpeta stactic? 7](#_Toc111545003)

[11. Crear un archivo base html en la APPS core 7](#_Toc111545004)

[12. Como se llaman a los CSS desde el archivo base html. 7](#_Toc111545005)

[13. Como consume un archivo hijo html al utilizar la herencia del archivo base html 8](#_Toc111545006)

[14. Crear un view que llame al html hijo 8](#_Toc111545007)

[15. Crear la urls que llame al views. 9](#_Toc111545008)

[16. Integrar la aplicación APPS core al proyecto principal 9](#_Toc111545009)

[17. Crear las tablas del sistema de usuarios para utilizar el panel de administración. 10](#_Toc111545010)

[18. Crear un usuario para poder ingresar al Panel de Administración. 10](#_Toc111545011)

[19. Que es un modelo en Django 10](#_Toc111545012)

[20. Crear un modelo en Django. 11](#_Toc111545013)

[21. Migrar el Modelo a la base del Panel de Administración. 11](#_Toc111545014)

[22. Integrar el Modelo al Panel de Administración. 11](#_Toc111545015)

[23. Ingresar información al modelo por el Panel de Administración. 12](#_Toc111545017)

[24. Realizar la consulta de todo lo ingresado en el modelo desde el views. 12](#_Toc111545018)

[25. Mostrar los datos guardados en el modelo al html hijo. 13](#_Toc111545019)

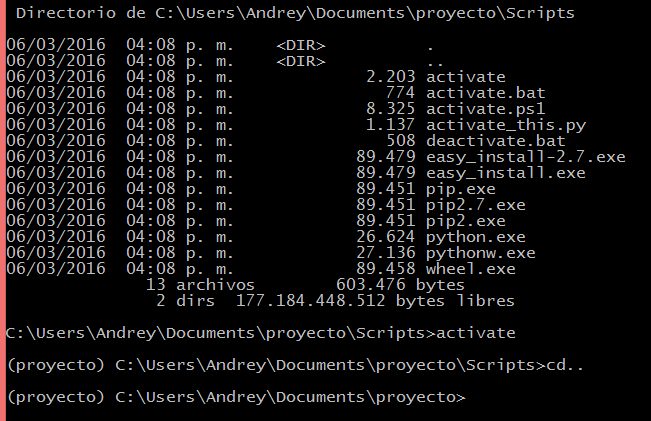
# ¿Qué es Django?

**Django es un framework web diseñado para realizar aplicaciones de cualquier complejidad**en unos tiempos muy razonables.

Está escrito en Python y tiene una comunidad muy amplia, que está en continuo crecimiento.

# ¿Qué es la máquina virtual en Django?

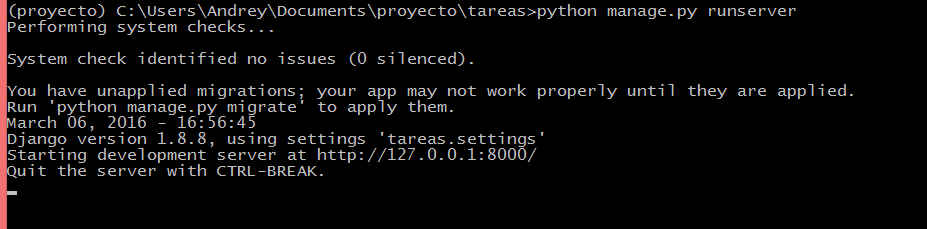
Se la utiliza para crear el entorno virtual Lo primero que haremos será crea un directorio para nuestro proyecto, en este caso “tareas”. De esta forma creamos el directorio tareas y accedemos a él. Aquí utilizamos el módulo “venv” para crear un entorno virtual llamada “venv” en el directorio actual.



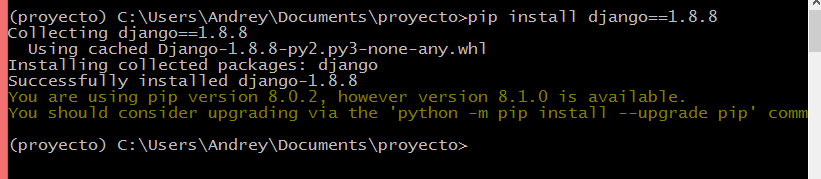
# ¿Qué es MVT en Django?

 Django redefine este modelo como **MVT**: **Modelo-Vista-Template.** Hasta ahora lo que hemos hecho no requería de interactuar con la base de datos.

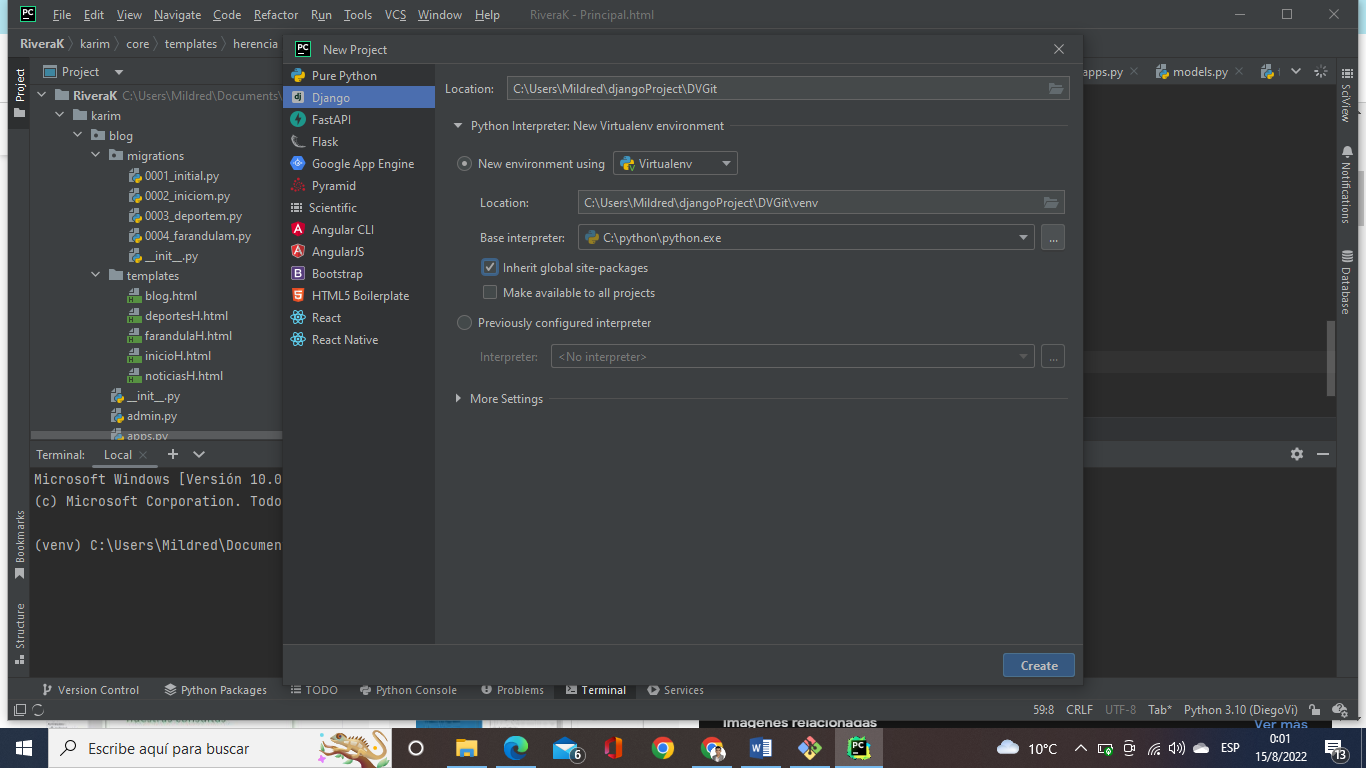
# Crear un proyecto con la máquina virtual.



# Descargar los instaladores de Django al proyecto

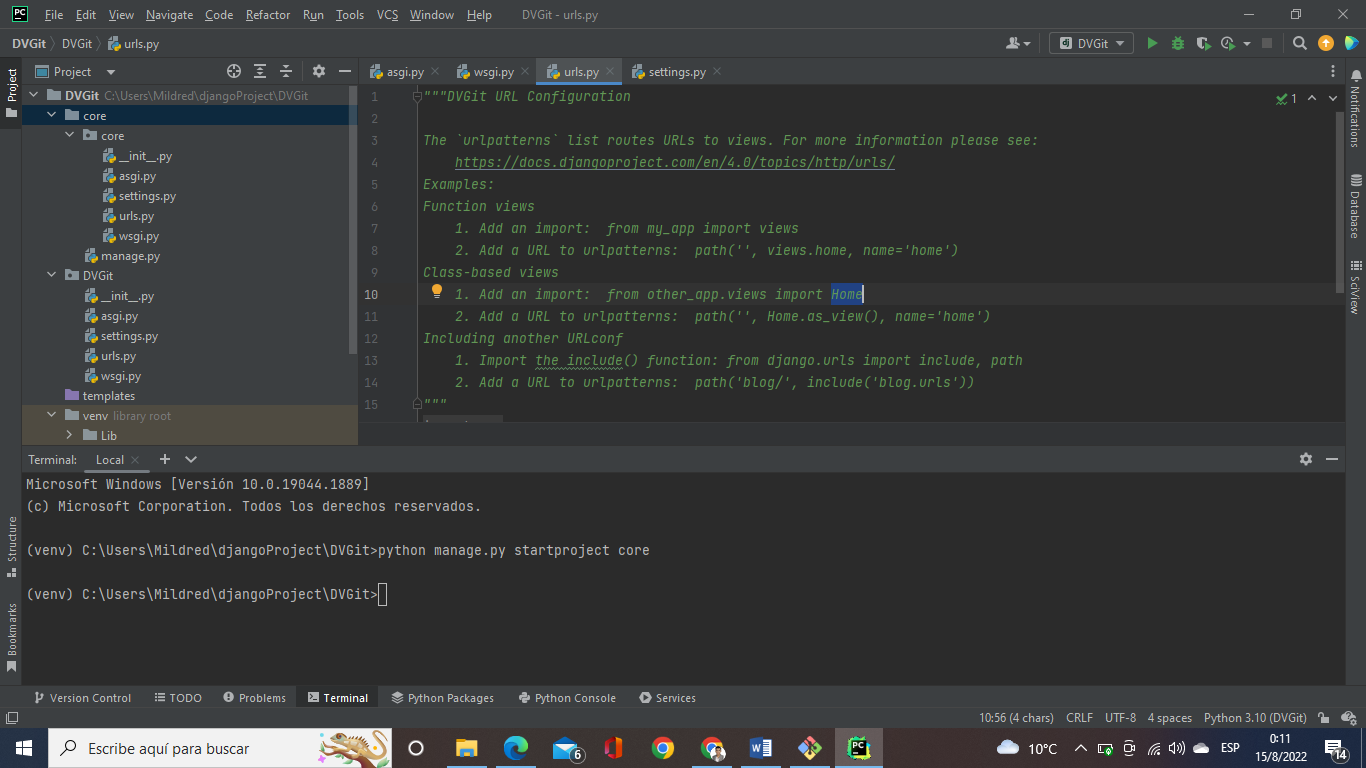


# Crear un proyecto para programar en Django



# Ejecutar el proyecto y el mensaje de felicitaciones.

# Crear apps core

Utilizamos el código en la terminal para crear aplicación: python manage.py startproject core.

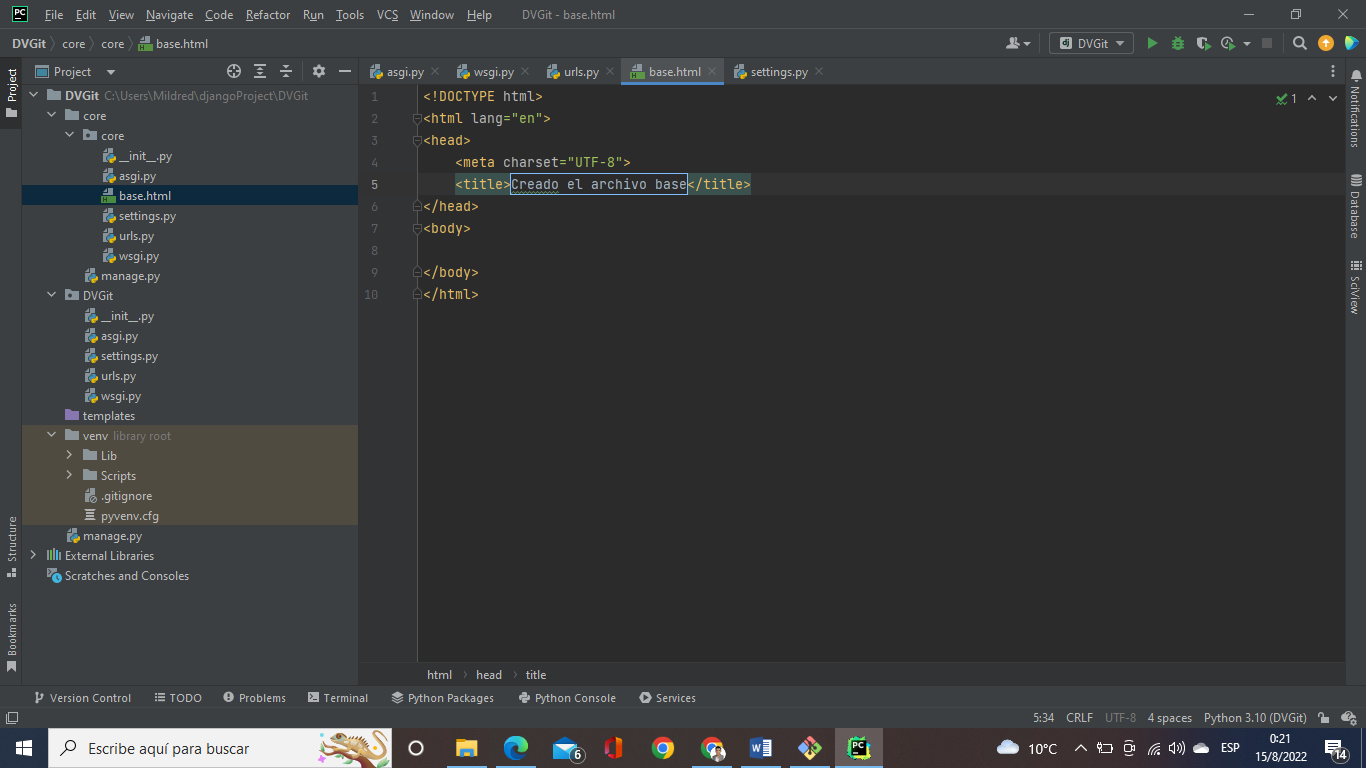
# ¿Qué es la carpeta “Templates”?

Una plantilla es un archivo de texto que determina la estructura o diseño de un archivo (como una página HTML), con marcadores usados para representar el contenido real. Django automáticamente buscará plantillas en un directorio llamado 'templates' de su aplicación.

# ¿Qué es la Carpeta stactic?

django. contrib. staticfiles proporciona un comando de administración conveniente para recopilar archivos estáticos en un solo directorio para que pueda servirlos fácilmente. Esto copiará todos los archivos de sus carpetas estáticas en el directorio STATIC\_ROOT.

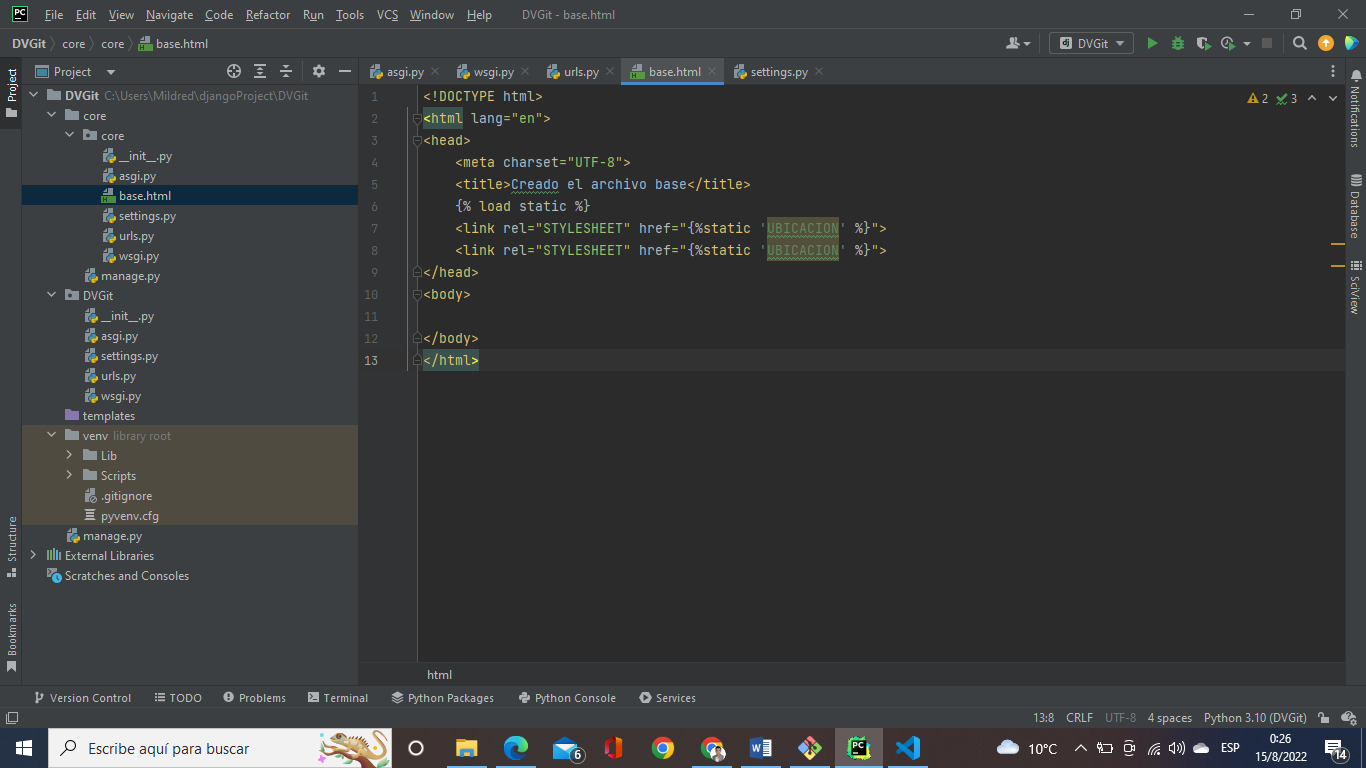
# Crear un archivo base html en la APPS core



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

<link rel="STYLESHEET" href="{%static 'UBICACION' %}">

<link rel="STYLESHEET" href="{%static 'UBICACION' %}">

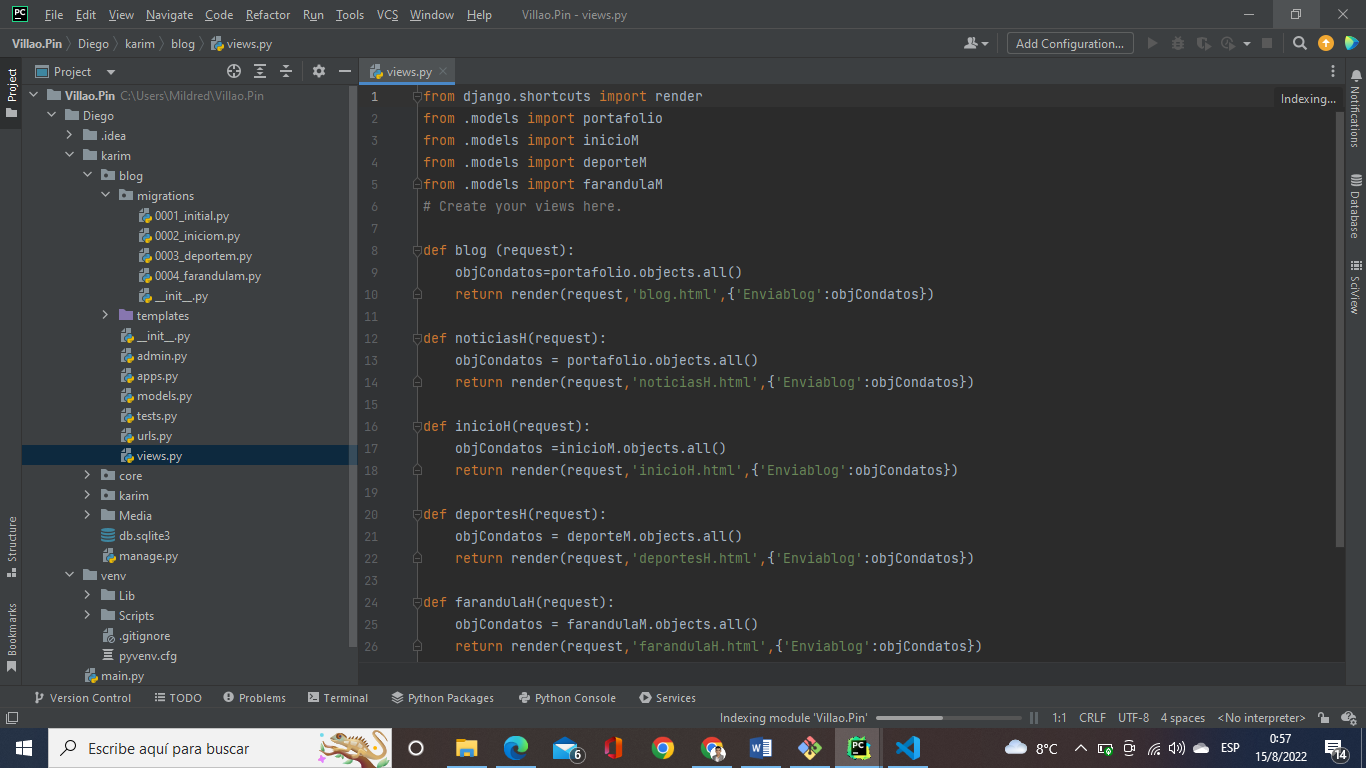


# Como consume un archivo hijo html al utilizar la herencia del archivo base html

En este caso creamos dentro de la core una subcarpeta con título herencia, para consumir el archivo hijo al heredar tenemos que utilizar estas sentencias al principio del archivo hijo, adjuntando la ubicacion donde se encuentra la herencia.

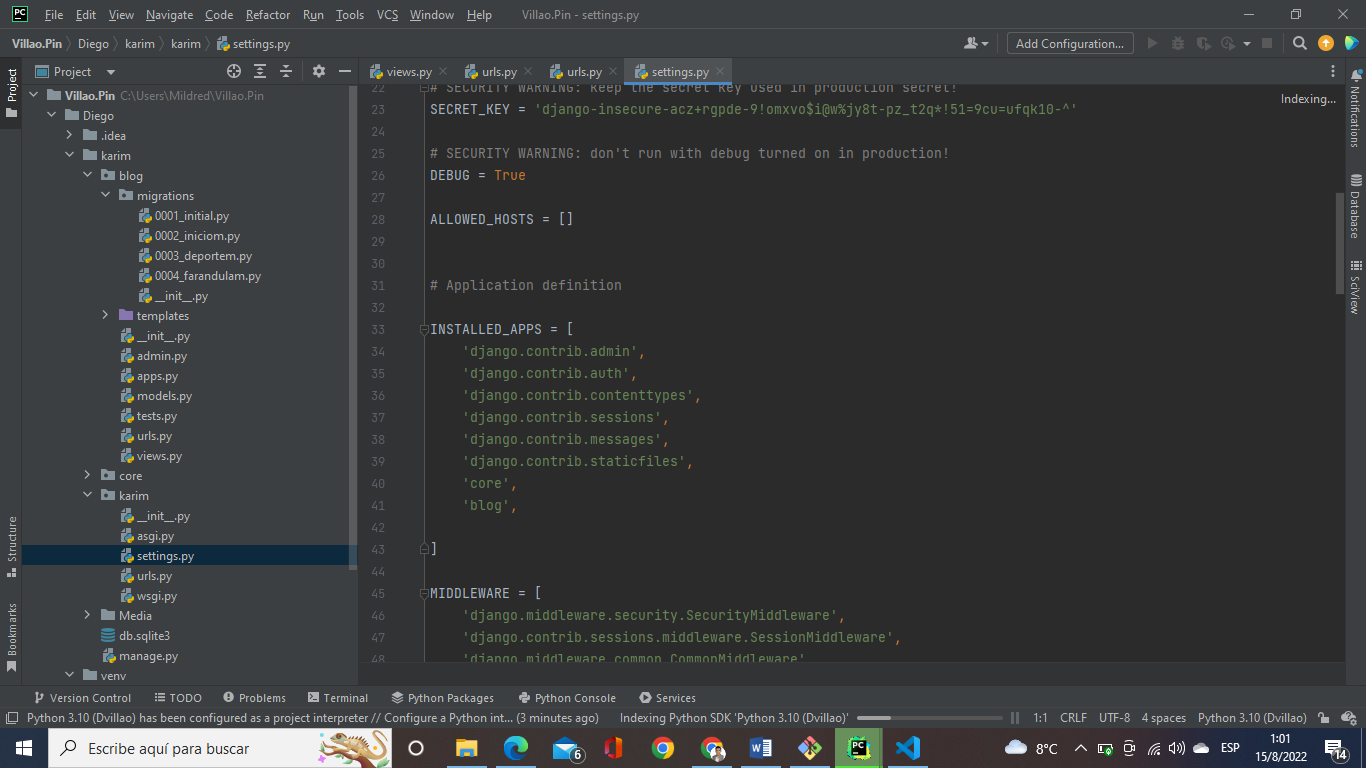
 {% extends 'herencia/base.html' %}

# Crear un view que llame al html hijo

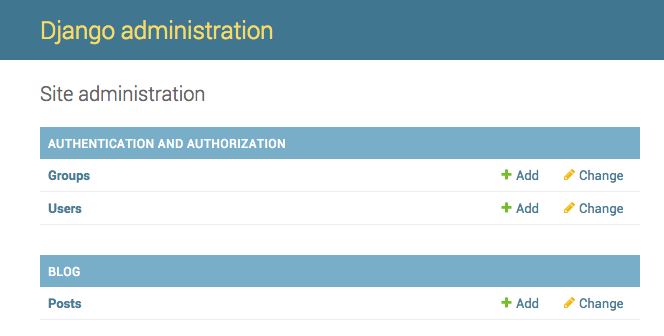


# Crear la urls que llame al views.

# Integrar la aplicación APPS core al proyecto principal



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

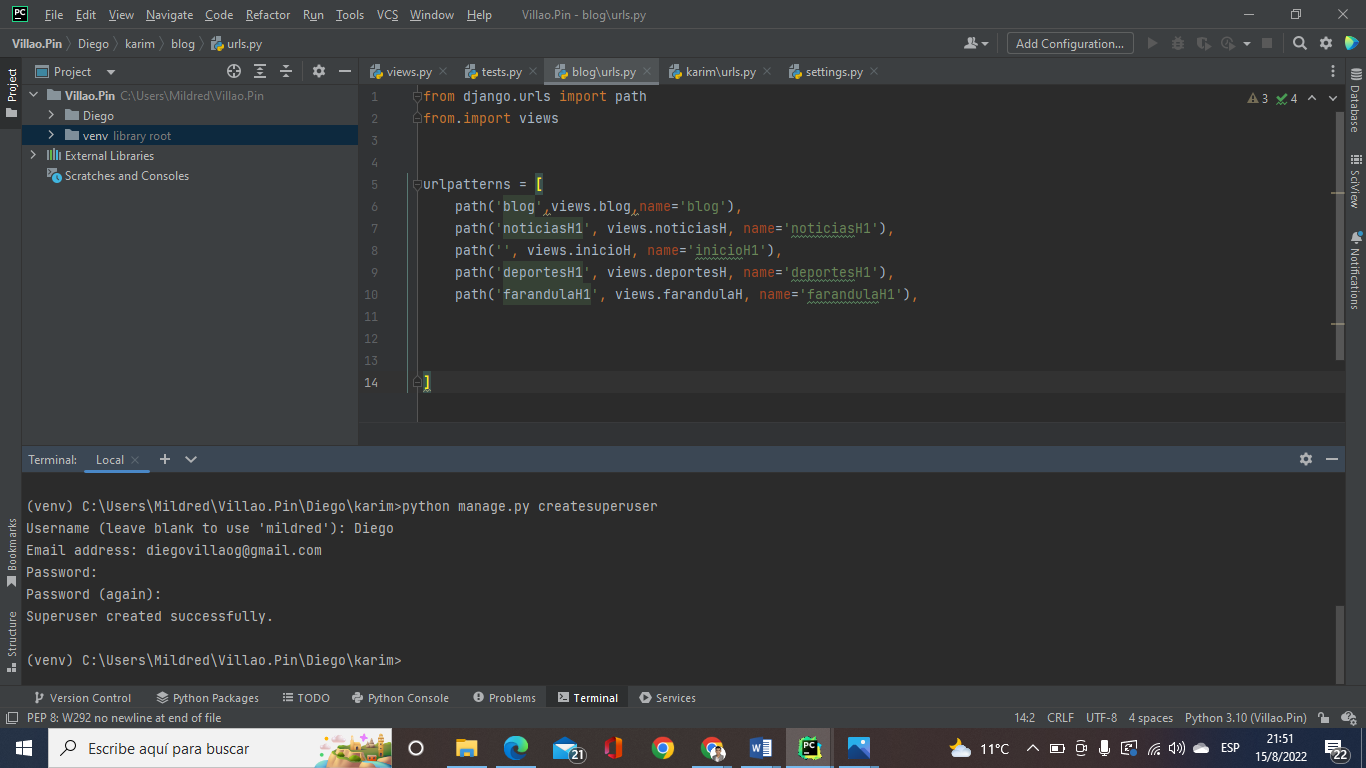


# Crear un usuario para poder ingresar al Panel de Administración.

Creamos un usuario en el panel de administración con el siguiente código:  
**python manage.py createsuperuser**

Luego de eso que nos pida los datos del usuario lo corremos de nuevo el proyecto con el código:

**Python manage,py runserver**

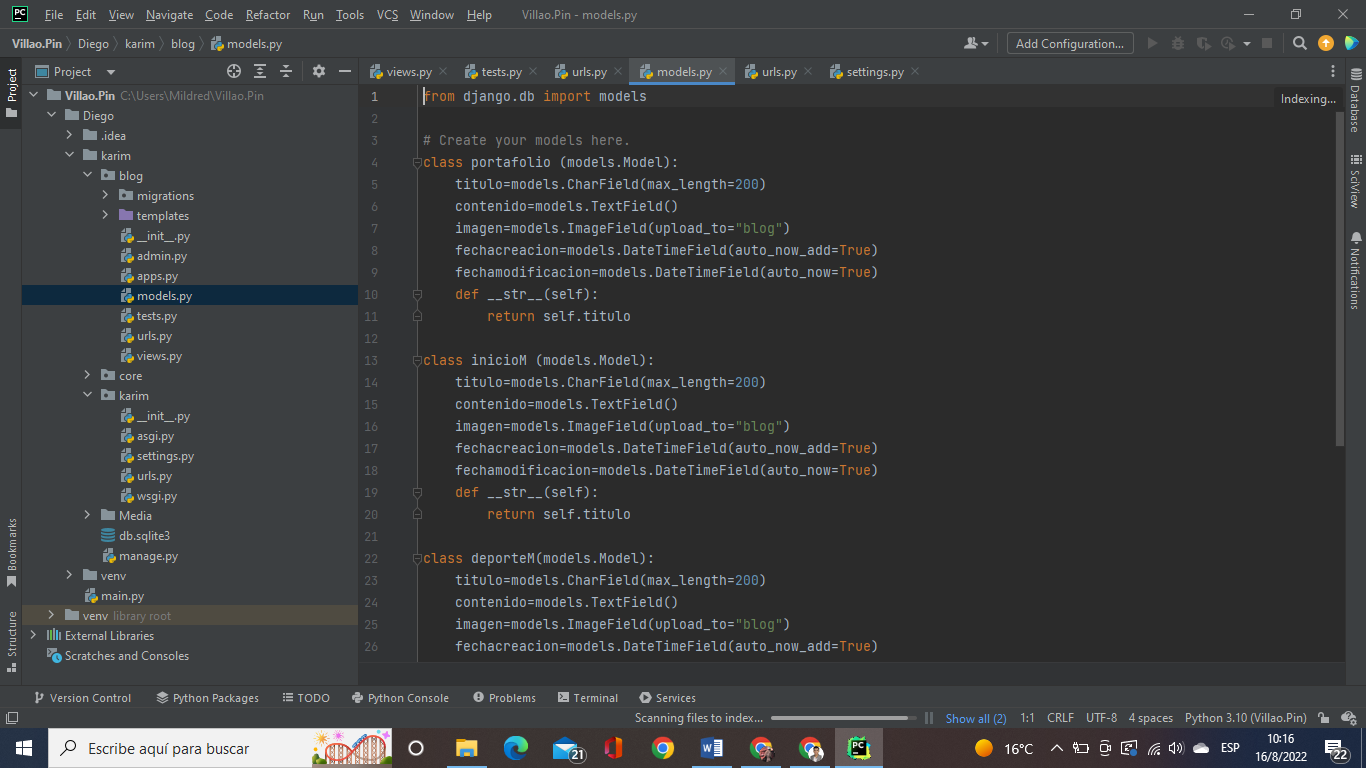


# Que es un modelo en Django

Un modelo en Django **es un tipo especial de objeto que se guarda en la base de datos** . Una base de datos es una colección de datos. Es un lugar en el cual almacenarás la información sobre usuarios, tus entradas de blog, etc. Utilizaremos una base de datos SQLite para almacenar nuestros datos.



# Crear un modelo en Django.

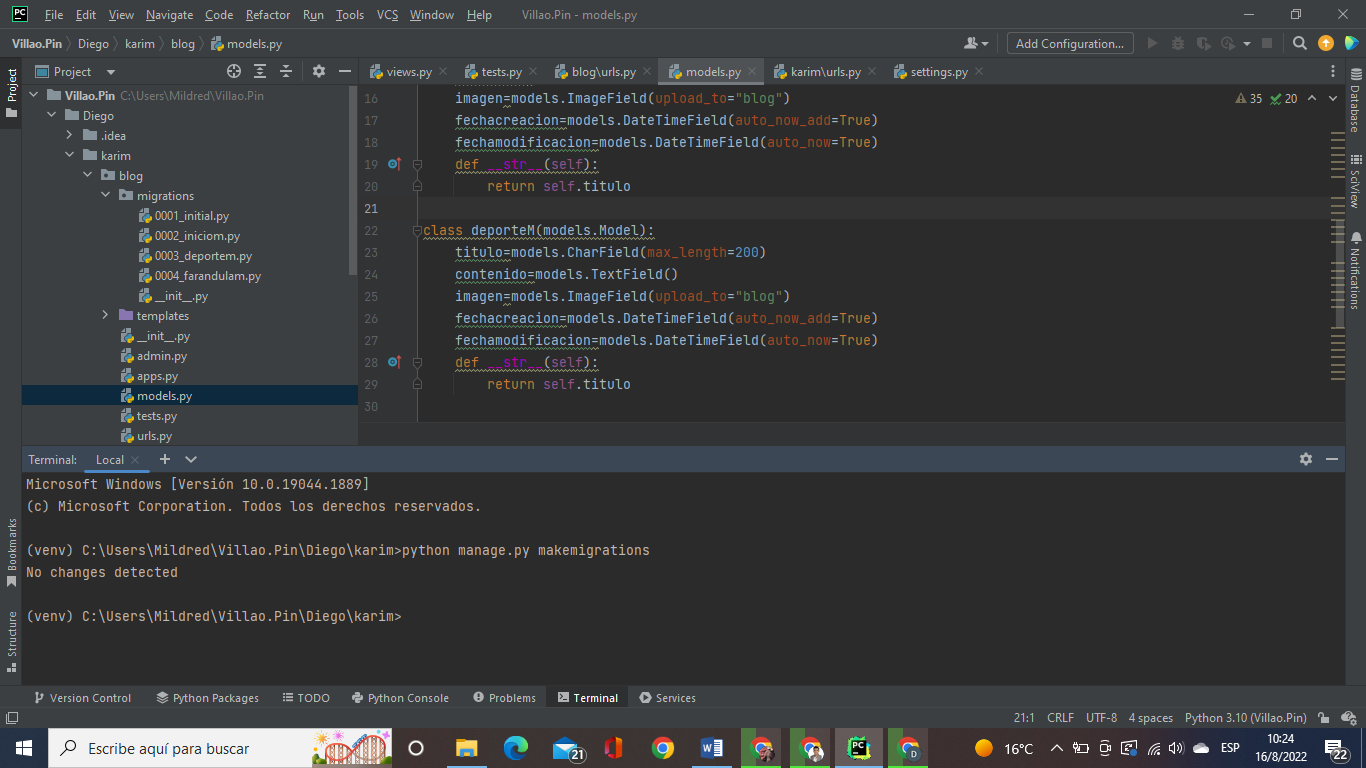


# Migrar el Modelo a la base del Panel de Administración.

Lo hacemos desde la terminal y con estos código:  
**python manage.py makemigrations**

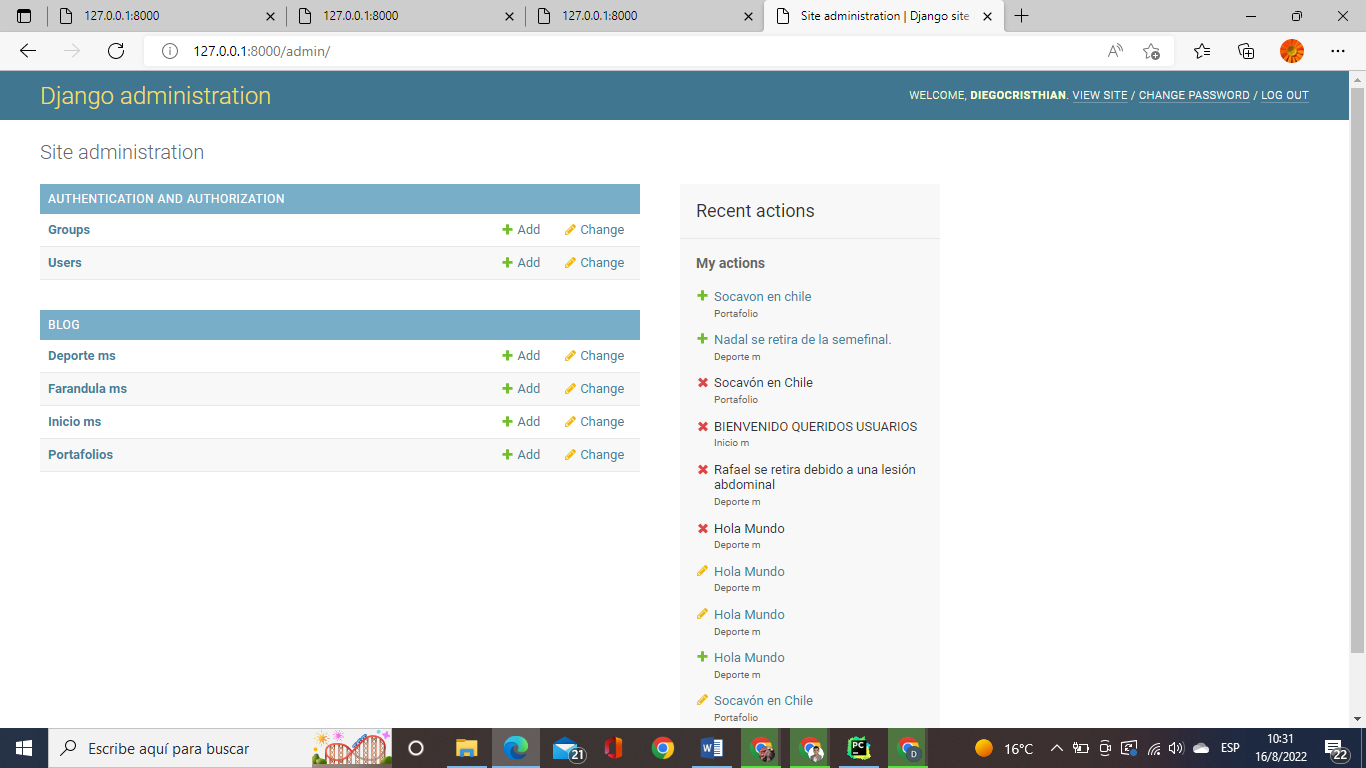
**python manage.py migrate**

Luego de ejecutar esa acción en la terminal, migraremos las tablas como podemos observar en la siguiente imagen.



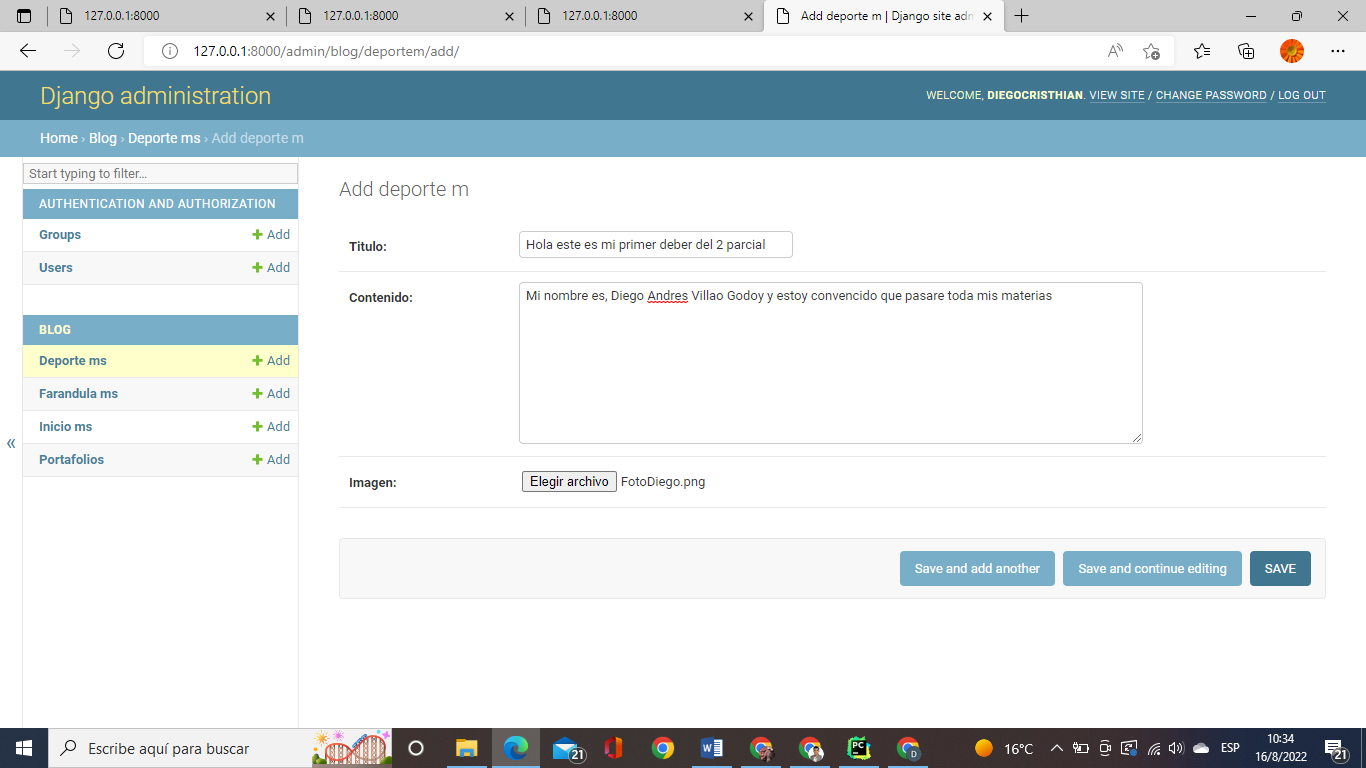
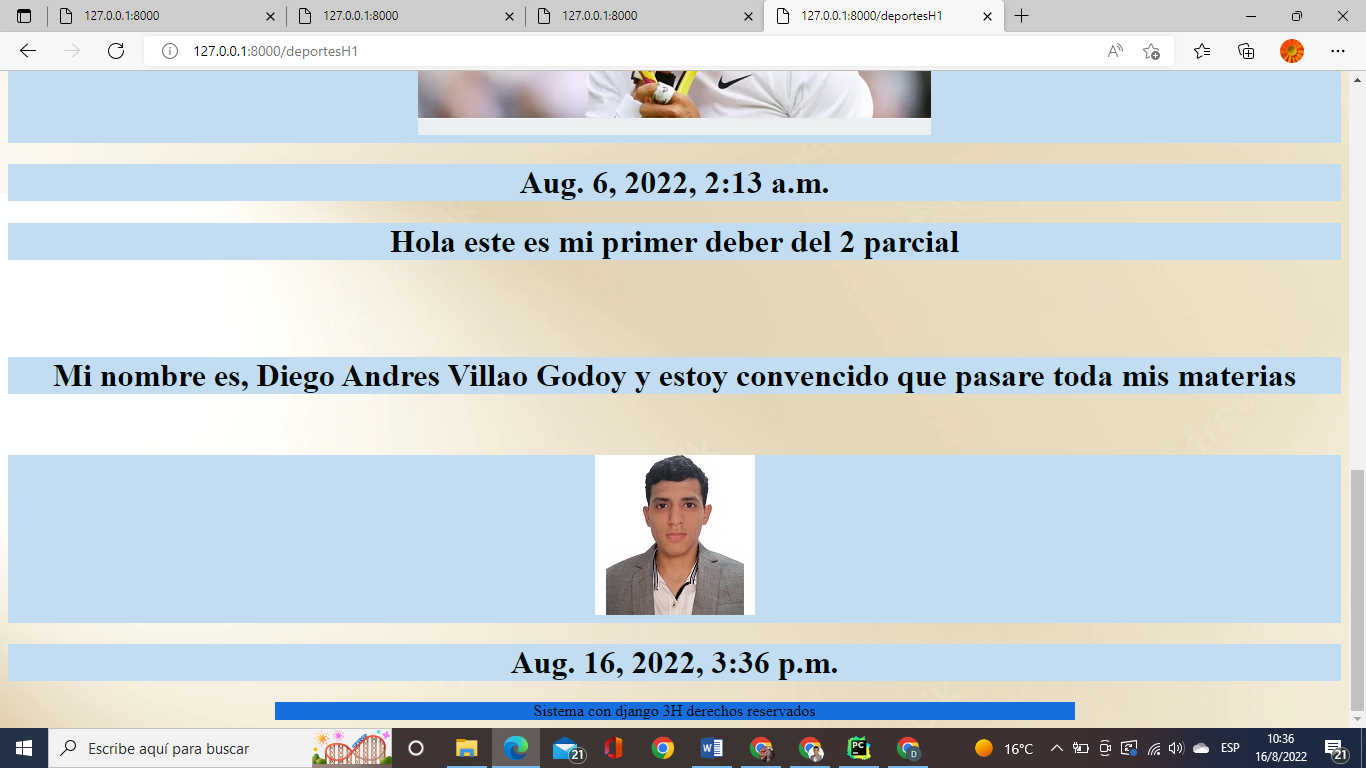
# Integrar el Modelo al Panel de Administración.

# Como podemos visualizar nos aparece los modelos que migramos de la tabla:

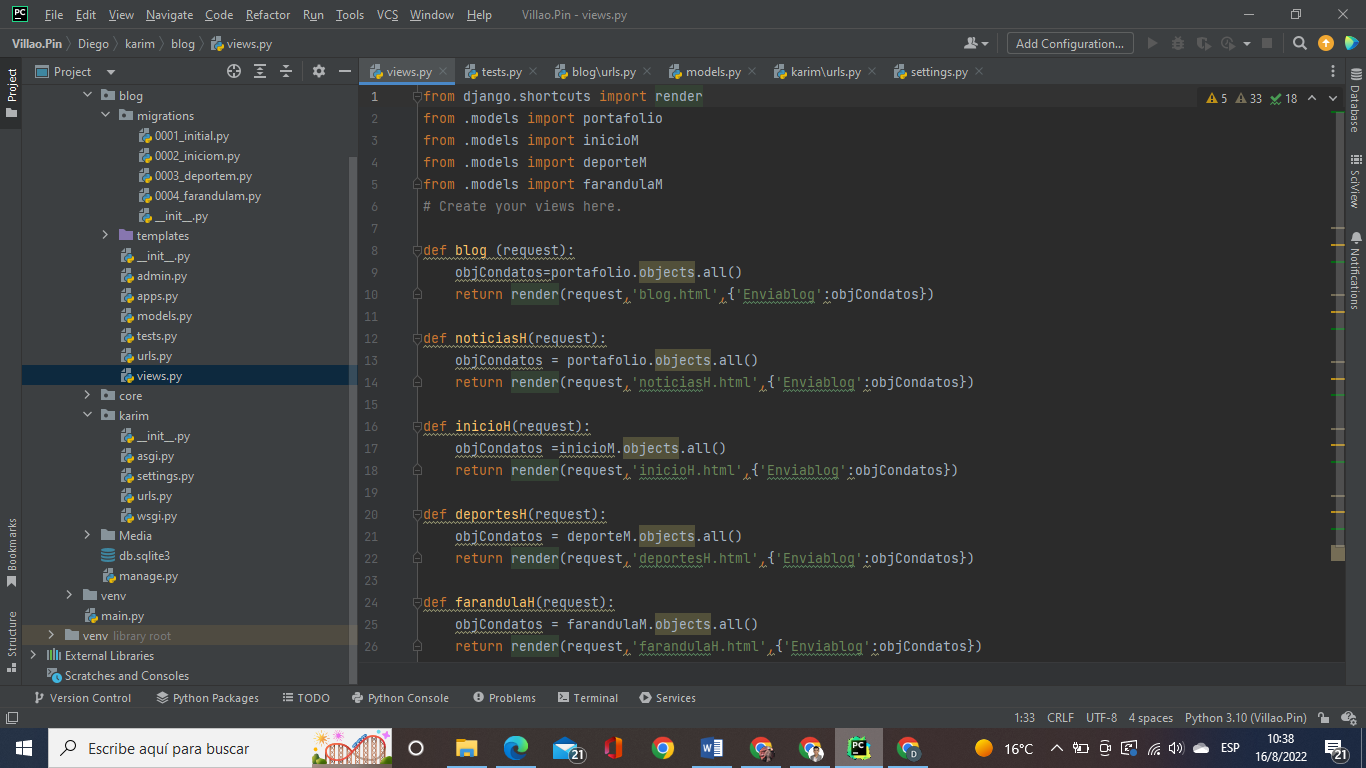


# Ingresar información al modelo por el Panel de Administración.

Añadimos un texto:



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



# Mostrar los datos guardados en el modelo al html hijo.