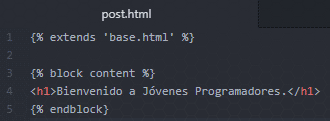
**Curso Python 2: Introducción a Django – Resumen sesión 5**

Ejecutar proyectos con pasos señalados en la sesión anterior.

Para seguir con la App, se deben crear 2 carpetas:

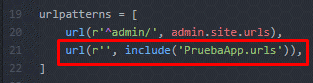
* ***static***: Tendrá todo lo relacionado con código JS, CSS e imágenes de la página web que se creará.
* ***templates***: Aloja los templates y también el archivo base, donde están todos los típicos meta tags de HTML. Es el lugar donde se visualiza cada template que solicita el usuario.

En la carpeta templates, se crea un nuevo archivo “post.html” que de momento sólo tendrá un título h1:

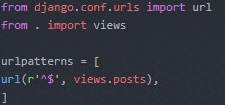


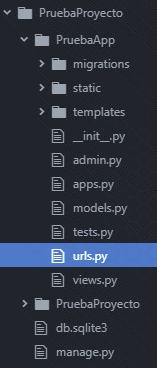
Enlace del URL con la vista

Se debe ir al archivo urls.py de la carpeta “PruebaProyecto” y añadir la siguiente línea:



Luego, en la carpeta “PruebaApp” se crea el archivo “urls.py” (NO SE REALIZA EN NINGUNA DE SUS SUB-CARPETAS) con el código





Al guardar e ir a la URL en el navegador, se verá:

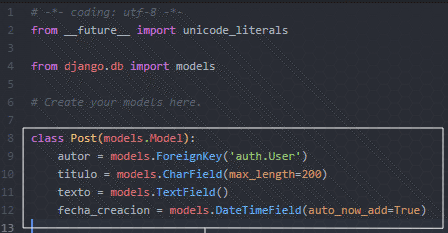


Los textos fueron señalados en el archivo base.html de los templates, con sus respectivas etiquetas HTML.

El modelo permitirá crear posts, para lo que necesitará los siguientes datos:

* Autor
* Título
* Texto
* Fecha de creación

Estos campos se añaden en el archivo models.py en la carpeta templates:



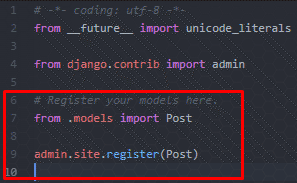
Luego, se crea una migración del modelo (en Git Bash) usando:

***python manage.py makemigrations***

Y posteriormente lo migramos a la BD usando

***python manage.py migrate***

Finalmente, en admin.py de la carpeta templates, se añade lo siguiente:



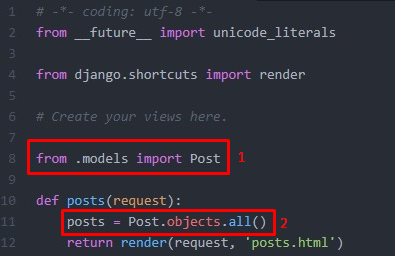
En el panel de administración veremos:



También tendremos vistas para agregar y modificar los posts.

Enlace del modelo con las vistas

1. En templates, archivo views.py, se debe importar el modelo Post
2. Obtener toda la información del modelo “Post”, antes del return de la función



Se modifica el método render() añadiendo un parámetro para enviarle la variable que contiene la información de las publicaciones al template:



Posteriormente, se modifica el HTML de los posts en la carpeta “templates” quedando



Las sentencias de Django se encierran con llave y símbolos de porcentaje {% % } (Tags de Django), y otras con dos llaves {{ }} (variables) para utilizarse directamente en el HTML.