

Contenido dinamico SQLite3 a una aplicación web con Django

Aprende cómo integrar bases de datos SQLite3 con un sitio web creado en Django, para visualizar de manera dinámica la información almacenada.

Grupo 5 Alvaro Manterola - Guido Saavedra - Cynthia Abarca - Esteban Santibañez

¿Qué es SQLite3?

SQLite3 es una base de datos que no requiere de un servidor ni de una configuración demasiado compleja. Su principal ventaja es la portabilidad, y es ideal para proyectos de pequeña o mediana escala.

¿Qué es Django?

Django es un framework web

Desarrollado en Python, Django es un framework para la creación de aplicaciones web. Permite implementar la arquitectura modelo-vista-controlador (MVC), lo que le brinda una alta flexibilidad y modularidad.

Características destacadas de Django


- Administración de bases de datos eficiente.
- Cobertura de seguridad del sitio.
- Gran calidad en el manejo de errores.
- Escalable y adaptable.

Preparación del ambiente de desarrollo

Crea un ambiente virtual de Python para trabajar, ya que te permitirá mantener instaladas las versiones de las librerías y paquetes que necesitas sin interferir con el resto de los proyectos.

Sigue las siguientes instrucciones para crear un ambiente de desarrollo:

1. Abre una terminal en la raíz del proyecto.
2. Crea un ambiente virtual con el siguiente comando: `python3 -m venv env_nombre` / En windows `python -m venv env_nombre`
3. Activa el ambiente virtual: `source env_nombre/bin/activate.bat`
4. Instala la librería de Python para comunicarse con la estructura de datos SQLite3: `python3 -m pip install sqlite3`



Creación de un nuevo proyecto en Django

Para crear un nuevo proyecto en Django, se debe utilizar el comando `django-admin startproject project_name`, seguido del nombre del proyecto. Este comando generará la estructura de archivos necesaria para comenzar a trabajar en el proyecto. Una vez creado, se puede comenzar a definir las aplicaciones y modelos que conformarán la aplicación web encontraremos nuestro archivo `db.sqlite3`.

Definición del modelo de datos

Tablas en SQLite3

SQLite3 funciona mediante tablas, que son estructuras de datos organizadas en filas y columnas, al igual que en Excel. Para conectar una aplicación de Django con una base de datos SQLite3, es importante definir cuáles serán las tablas y campos necesarios.

Modelos de Django

En Django, los modelos son clases que se usan para definir las tablas de la base de datos. Cada atributo de la clase corresponde a una columna en la tabla, y se pueden definir restricciones de datos como valor default, tipo de entrada, entre otros. Para establecer una relación entre varias tablas, es necesario definir el tipo de clave ajena según corresponda.

Creación de vistas

Las vistas de Django manejan las solicitudes HTTP y devuelven una respuesta. Esta respuesta puede ser una página HTML, un archivo JSON, una imagen, entre otros.

Creamos la vista en el archivo `views.py` de la siguiente manera:

```
from .models import NombreClaseDeTuModelo
```

```
def nombre_de_tu_funcion(request):
```

```
    # ...
```

```
    datos = NombreClaseDeTuModelo.objects.all()
```

1. Importa las funciones que acabas de crear desde el módulo `models`:
2. Crea una función que maneje el HTTP request:
3. Realiza la lógica necesaria para conectarte con la base de datos:

Representación de los datos en la página web

Atributos de Django para HTML

Django incluye algunos atributos que facilitan la inserción dinámica de información en una página, como `{{ variable }}`, que permite mostrar el valor de una variable que se calculó en la vista.

For loop en Django

Permite recorrer una lista de elementos y mostrarlos en pantalla. Se usa la siguiente sintaxis:

```
<ul>
  {% for elemento in lista_de_elementos %}
    <li>{{ elemento }}</li>
  {% endfor %}
</ul>
```


Conclusión

Integrar Django con SQLite3 permite tener páginas web más dinámicas y llenas de contenido. Además, permite mostrar datos desde distintas perspectivas en vez de tener páginas estáticas. Esperamos que esta guía te haya sido útil y que puedas aplicar los conocimientos adquiridos en tus propios proyectos.