Santiago Salamanca Diaz

Análisis y Desarrollo de Software (ADSO)

2558104 G1-G2

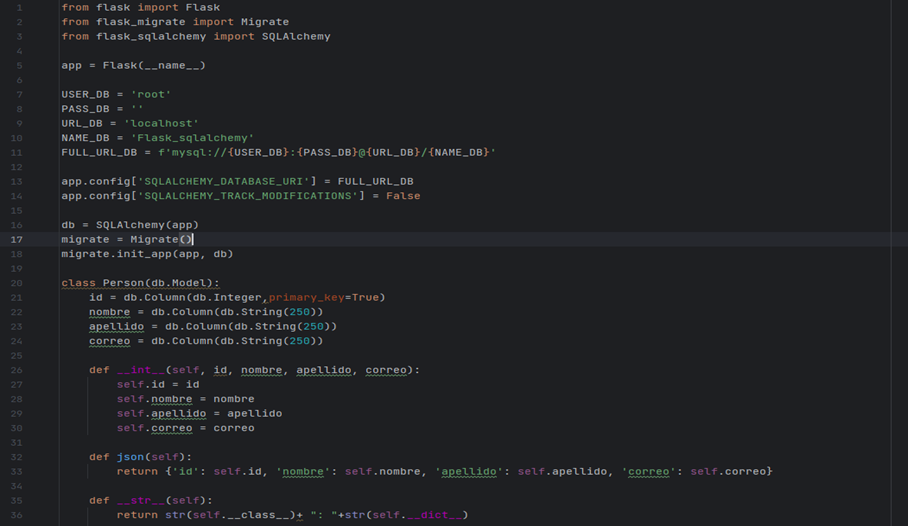
Instructora Isaura María Suarez Novoa

Servicio Nacional de Aprendizaje (SENA), CEET

Bogotá D.C.

18/08/2023

Taller

En el código anterior, lo configuré para postgresql. Tú deber es analizarlo y configurarlo para mysql. Pega aquí como quedo tu configuración.

Creación de Estructuras de Almacenamiento:

Al ejecutar este comando, se establecen dos estructuras de almacenamiento esenciales para administrar las migraciones de tu aplicación. Una de ellas se utiliza para guardar las migraciones en sí (denominada "migrations"), mientras que la otra tiene la función de registrar las diferentes versiones de estas migraciones (llamada "versions"). Estas estructuras son vitales para llevar a cabo el proceso de migración de manera organizada y efectiva.

Generación de Archivos de Configuración Fundamental:

El comando también automatiza la generación de varios archivos de configuración esenciales que son necesarios para la correcta configuración y ejecución del proceso de migración. Entre los archivos clave que se crean, se incluyen:

Alembic.ini: Este archivo de configuración te permite definir opciones de conexión a la base de datos, rutas de almacenamiento de migraciones y otras configuraciones fundamentales.

Env.py: Este archivo de configuración específico de Flask-Migrate se utiliza para cargar la configuración de tu aplicación y establecer la conexión a la base de datos.

Personalización Inicial:

Es importante destacar que, a pesar de la generación automática de estos archivos de configuración, debes revisarlos y personalizarlos de acuerdo a las necesidades particulares de tu aplicación. Dentro de estos archivos, tendrás que editar detalles como la URL de la base de datos, opciones de conexión y configuración de registro para asegurarte de que se ajusten perfectamente a los requisitos específicos de tu proyecto.

¿Qué hace flask db migrate?

flask db migrate es un comando que se utiliza para generar archivos de migración que describen los cambios necesarios en la base de datos para mantenerla sincronizada con los modelos de datos de tu aplicación Flask. Después de generar una migración, puedes aplicarla con el comando flask db upgrade para efectuar los cambios en la base de datos.

¿Qué sucede después de ejecutar flask db upgrade?

flask db upgrade aplica las migraciones pendientes a la base de datos, realizando cambios como crear tablas o modificar columnas para que coincidan con la estructura definida en tu aplicación Flask. Esto garantiza que la base de datos esté actualizada y en línea con los modelos de datos de tu aplicación. Es una acción fundamental para mantener la integridad y la consistencia de la base de datos a medida que evoluciona tu aplicación.

¿Qué es ORM?

Un ORM (Mapeo Objeto-Relacional) es una herramienta que permite a los programadores trabajar con bases de datos de una manera más parecida a cómo trabajan con objetos en la programación. En lugar de escribir consultas de bases de datos en SQL, pueden usar objetos y clases en su lenguaje de programación para interactuar con los datos de la base de datos. El ORM se encarga de traducir estas interacciones en operaciones de base de datos subyacentes. Esto facilita el desarrollo y la administración de aplicaciones que utilizan bases de datos relacionales.