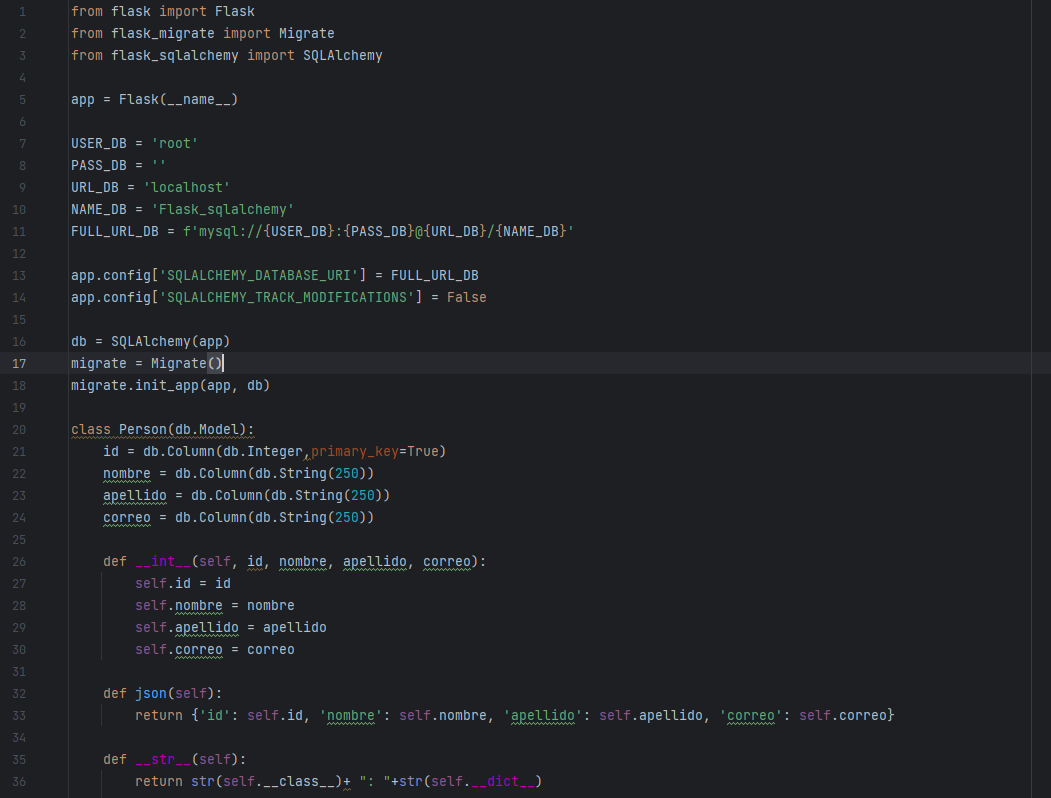
**Practica ORM**

**Marlon Conda  
2558104**

**SENA**

**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 Directorios:** El comando crea dos directorios: uno para almacenar las migraciones (migrations) y otro para almacenar las versiones de las migraciones (versions). Estos directorios son esenciales para el proceso de migración.

**Generación de Archivos de Configuración**: El comando también genera varios archivos necesarios para la configuración y ejecución del proceso de migración. Entre estos archivos se incluyen:

**alembic.ini**: Un archivo de configuración donde puedes establecer opciones de conexión a la base de datos, ubicaciones de migraciones, etc.

**env.py**: Un archivo de configuración específico de Flask-Migrate que se utiliza para cargar la configuración y establecer la conexión a la base de datos.

**Iniciar Ajustes:** El comando genera archivos de configuración predeterminados, pero es importante revisarlos y ajustarlos según las necesidades de la aplicación. Debes editar alembic.ini y env.py para establecer detalles como la URL de la base de datos, opciones de conexión y configuración de registro.

**¿Qué hace flask db migrate?**

Este comando crea automáticamente un nuevo script de migración en la carpeta migrations/versions en función de los cambios detectados en los modelos de datos y en la base de datos. Flask-Migrate compara la estructura actual de la base de datos con los modelos definidos y genera un script que puede aplicarse para realizar esos cambios en la base de datos.  
  
Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

**4. ¿Qué sucede después de ejecutar flask db upgrade?**

Este comando aplica las migraciones pendientes a la base de datos. Ejecuta los scripts de migración generados previamente con flask db migrate. Cada script de migración realiza cambios específicos en la base de datos, como crear tablas, modificar columnas, etc. Después de ejecutar flask db upgrade, los cambios definidos en los scripts se aplican a la base de datos y la estructura se actualiza en consecuencia.

**5. ¿Qué es un ORM (Object-Relational Mapping)?**

Un ORM es una técnica que permite mapear objetos de programación a tablas en una base de datos relacional. En lugar de interactuar directamente con SQL, un ORM permite a los desarrolladores trabajar con objetos y relaciones, abstrayendo gran parte de la complejidad de las operaciones de base de datos. SQLAlchemy es un ejemplo de ORM en Python.