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



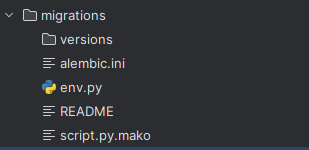
Ahora en la terminal, ejecuta el siguiente comando:



Revisa tu carpeta de proyecto y responde:

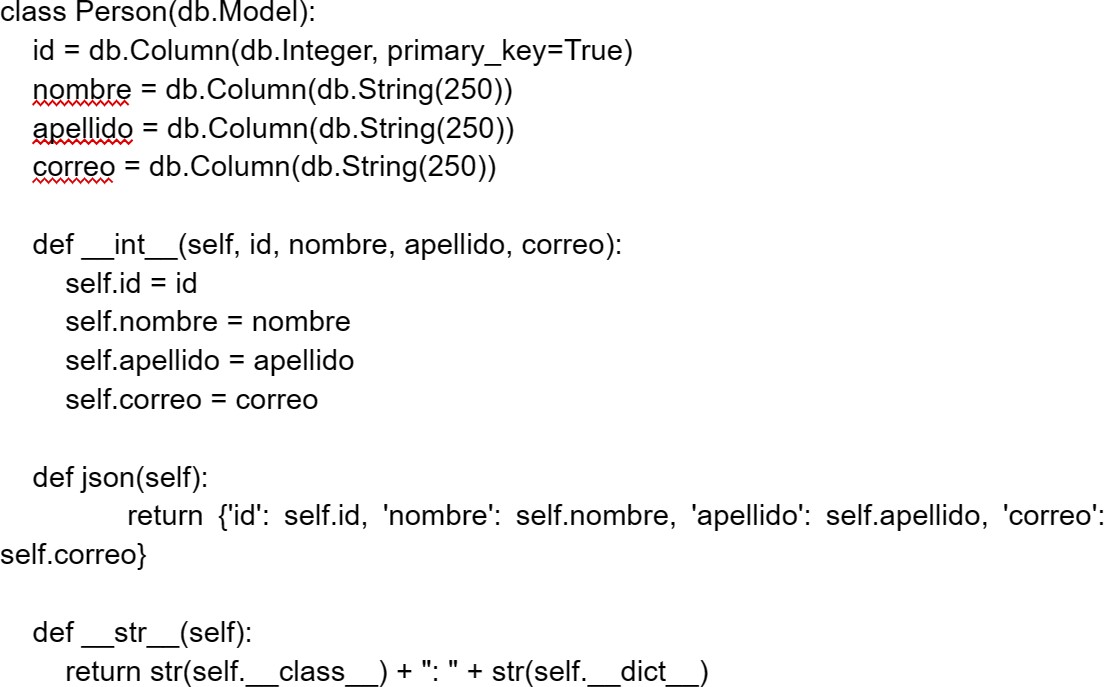
1. ¿Qué sucedió?

Creo la carpeta de migrations



Buen trabajo. ¡¡¡¡¡Continuemos!!!!!

Seguimos en el archivo app.py escribiendo líneas de código:



¿Qué crees tú que hace esa parte del código escrito anteriormente?

La primera Crea la clase Person, como medio para crear una tabla, sus y columnas correspondientes en la base de datos.

Bien, ahora en la terminal escribe el siguiente comando:



Revisa la base de datos que creaste en worbench (no se te olvide actualizar) y comenta que pasó

Se crearon nuevas carpetas y creo un tabla llamada alembic\_version.

Bien, Ahora revisa la carpeta migrations de tu proyecto. Dentro de ella existe una carpeta llamada versions. Abre el archivo py.

¿Explica que ves en ese archivo? Sé muy explicito

Prueba en casa:

Tiene una ID y fecha de creación.

Importo módulos, creo funciones que contienen los atributos que establecimos en la configuración de app.py

**Prueba en clase:**   
Debido al siguiente error no pude continuar con la practica desde aquí:

**sqlalchemy.exc.OperationalError: (MySQLdb.OperationalError) (1045, "Access denied for user 'root'@'localhost' (using password: YES)")**

**(Background on this error at:** [**https://sqlalche.me/e/20/e3q8**](https://sqlalche.me/e/20/e3q8)**)**

El error por lo investigado, seguramente se deba a protocolos de seguridad impuestos en el gestor de motor de base de datos XAMPP. Posibles soluciones para hacer que funcione, sean agregando en la configuración de conexión en “app.py” el puerto asignado para MySQL, otra opción es desinstalar e instalar nuevamente el XAMPP y así proceder con las configuraciones predeterminadas donde no presente esos problemas.

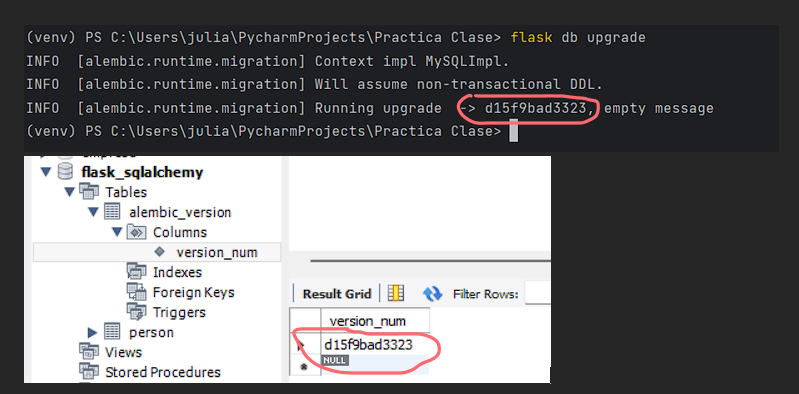
**Prueba en casa:**

Ahora, en la terminal escribe el siguiente comando:



1. **Revisa la base de datos en worbench (no se te olvide actualizar el esquema) y explica lo que sucedió.**

En la tabla alemic\_version, y en el atributo de versión\_num le asigno un valor

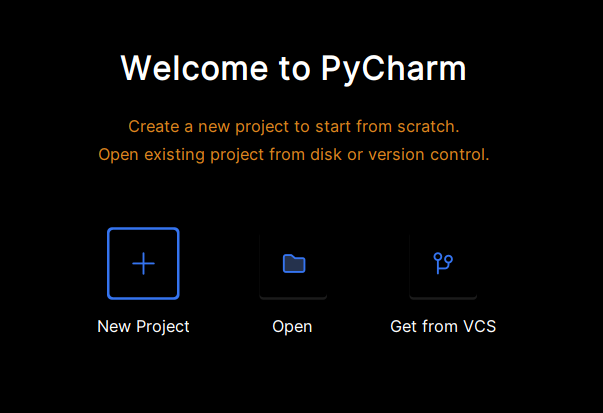


1. **Con base a la práctica, explica con tus palabras que es un ORM**

ORM es como un traductor entre la base de datos y nuestro código, uno le da las instrucciones y el las pasa a la base de datos sin que usemos comandos de la base de datos, por ejemplo este ejercicio nos ahorro tener que hacer los insert into.

1. **Crea un repositorio y sube la practica1 (con su debida explicación. Fotos con el paso a paso) y la practica 2 (foto con el paso a paso, incluyendo foto de la terminal).**

**Se crea un nuevo proyecto:**

****

Asignamos un nombre y la ruta donde se creará el proyecto

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

Esperar hasta que se cree el entorno virtual

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

**Abrir la terminal para instalar los módulos**

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

Instalar Flask (Framework escrito en Python)

**Interfaz de usuario gráfica, Texto

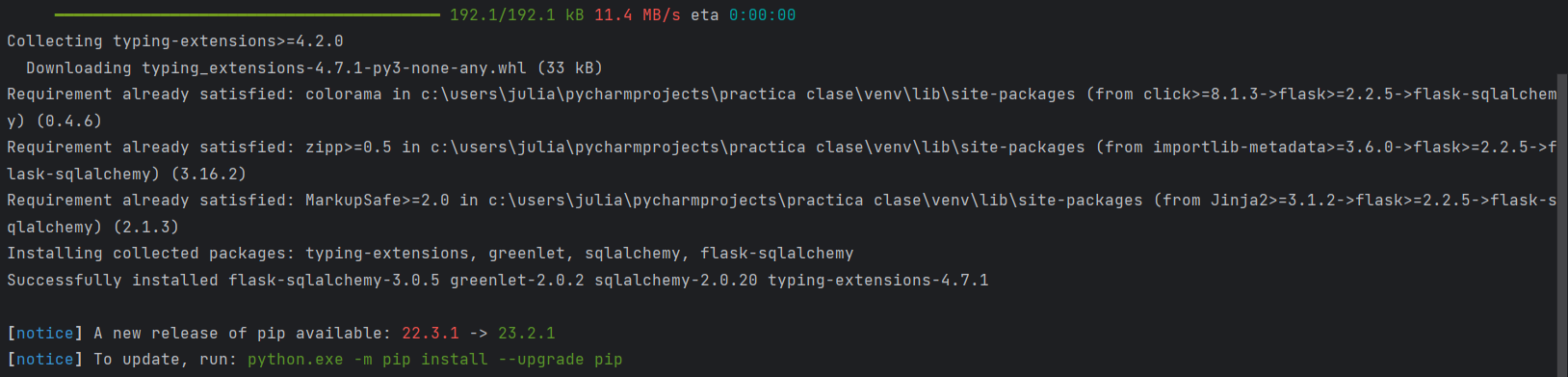
Descripción generada automáticamente**

****

Instalar Flask-SQLalchemy, extensión de Flask que facilita la creación y manipulación de bases de datos en aplicaciones web.

**Texto

Descripción generada automáticamente**

****

Instalar Marshmallow-SQLAlchemy, una extensión de las bibliotecas Marshmallow y SQLAlchemy que facilita la serialización y deserialización de objetos SQLAlchemy en formato JSON

Texto

Descripción generada automáticamente

Instalar flask-migrate, una extensión para el framework Flask en Python que proporciona una manera conveniente y controlada de manejar y aplicar cambios en el esquema de la base de datos cuando se utilizan bibliotecas ORM como SQLAlchemy.



Texto

Descripción generada automáticamente

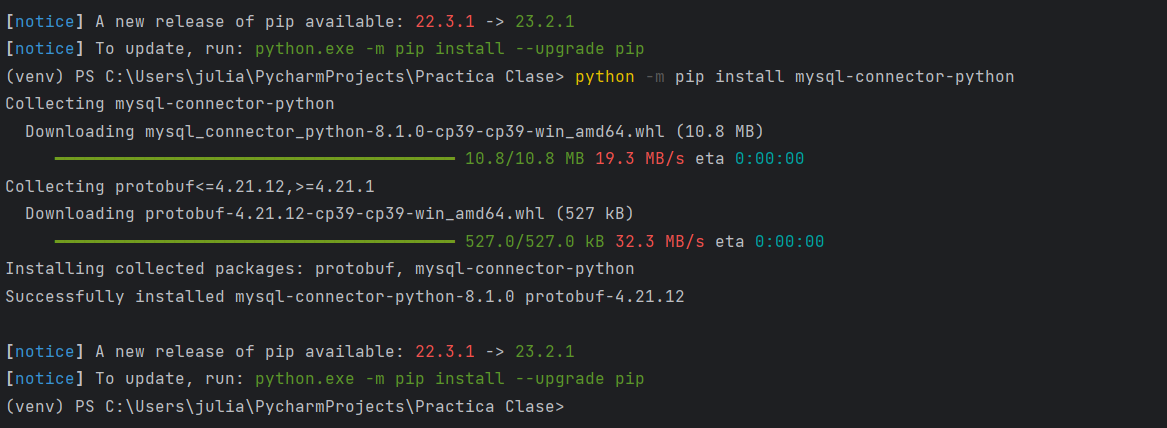
Instalar flask-mysqldb, extensión para el framework Flask en Python que proporciona una forma más sencilla y directa de conectarse y trabajar con bases de datos MySQL



Texto

Descripción generada automáticamente

Instalar los drivers para el motor de bases de datos



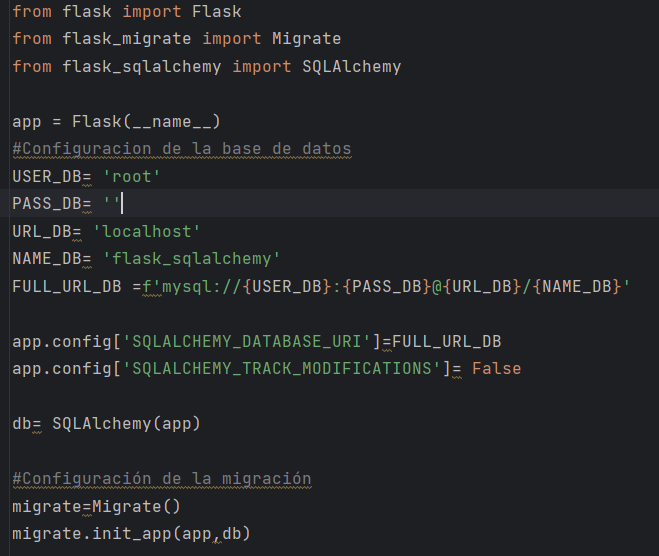
Crear archivo app.py, donde se realizarán las configuración de la conexión de la base de datos con la aplicación

**Interfaz de usuario gráfica, Aplicación

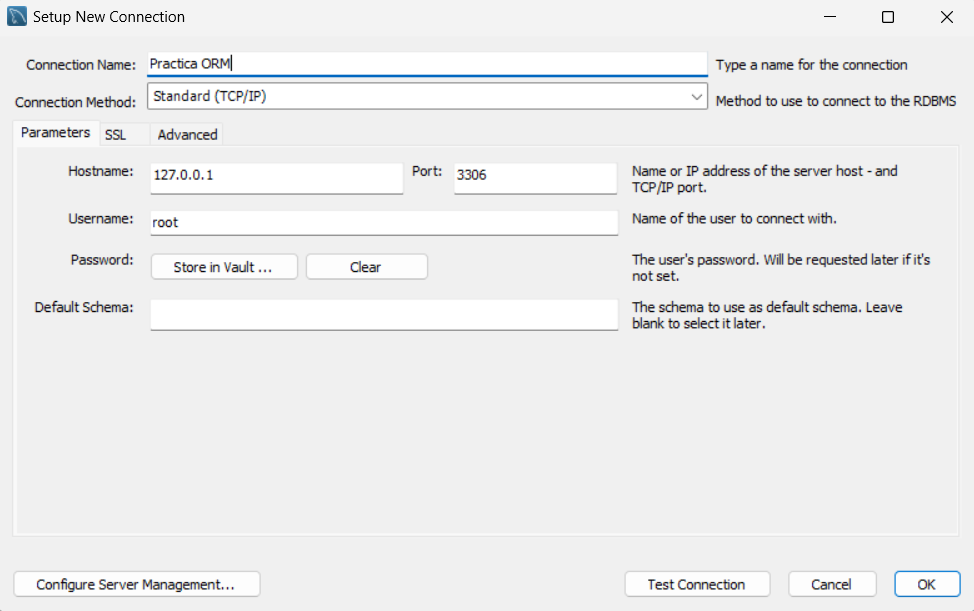
Descripción generada automáticamente**

Se importan los módulos de Flask, Migrate y SQLalchemy al archivo para iniciar la configuracion.

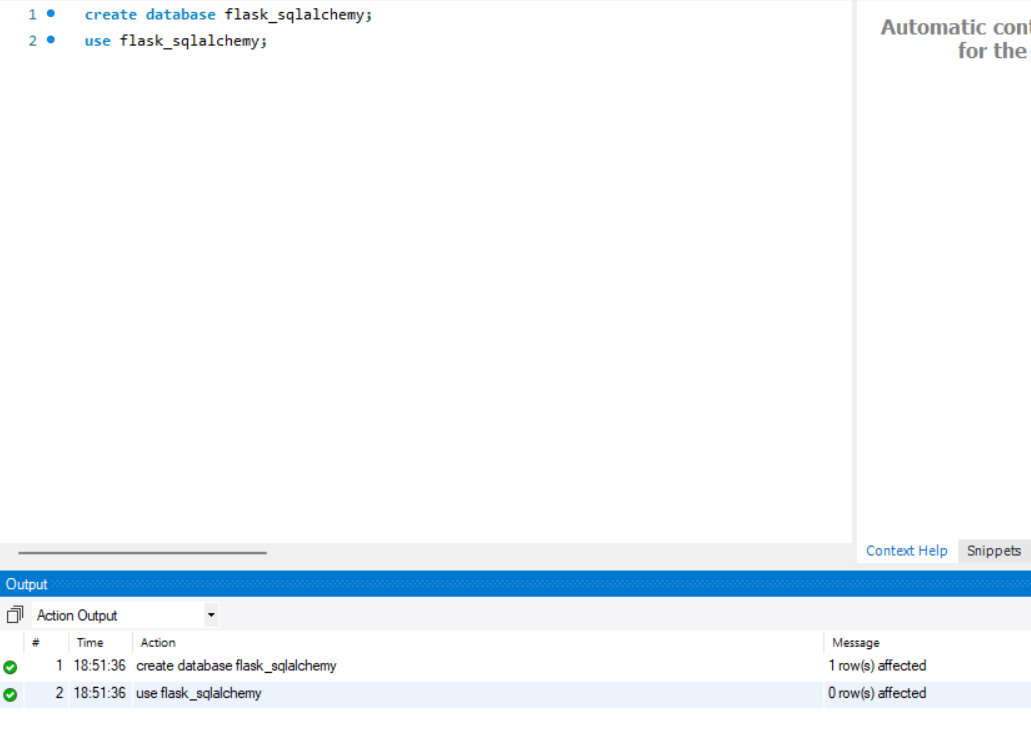
Se realiza la configuración de la conexión a la base de datos, estableciendo los parámetros de usuario, contraseña, host o url del servidor y el nombre de la base de datos a conectar.



Crear la instancia y la base de datos a la cual se conectará el proyecto



Crear la base de datos “flask\_sqlalchemy”.

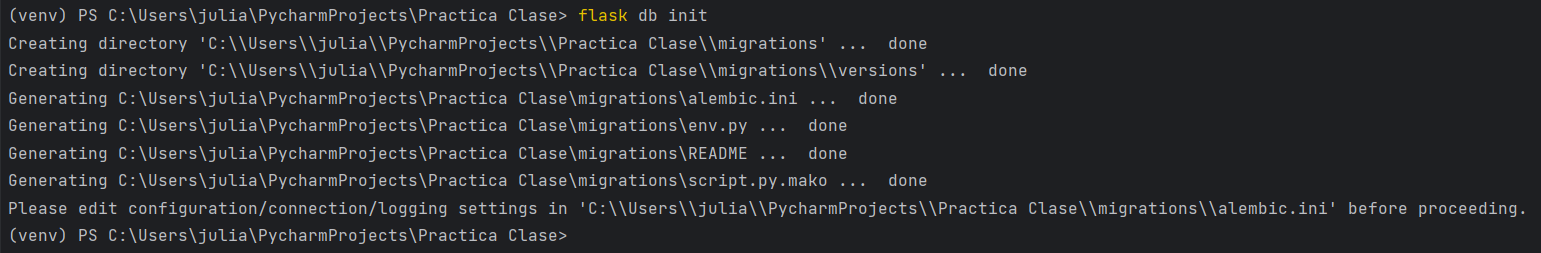


Esta sección del código define una clase llamada Person que hereda de db.Model, indicando que es un modelo de datos para interactuar con la base de datos utilizando SQLAlchemy.

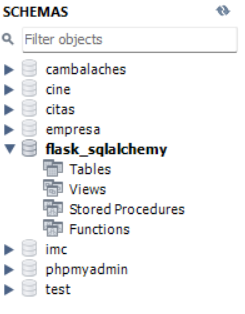
Texto

Descripción generada automáticamente

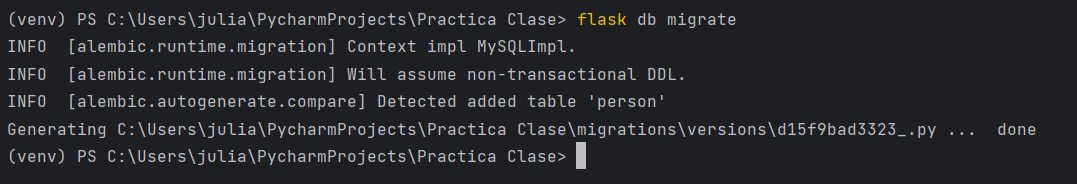
Crear la carpeta migrations, contendrá una estructura de directorios y archivos necesarios para gestionar las migraciones de datos del proyecto.

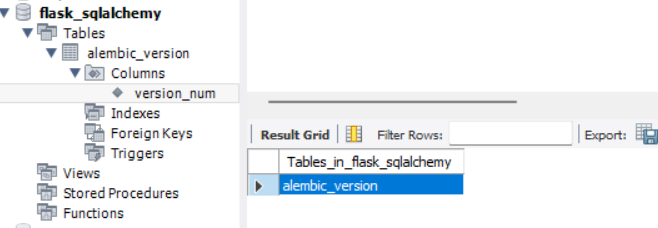


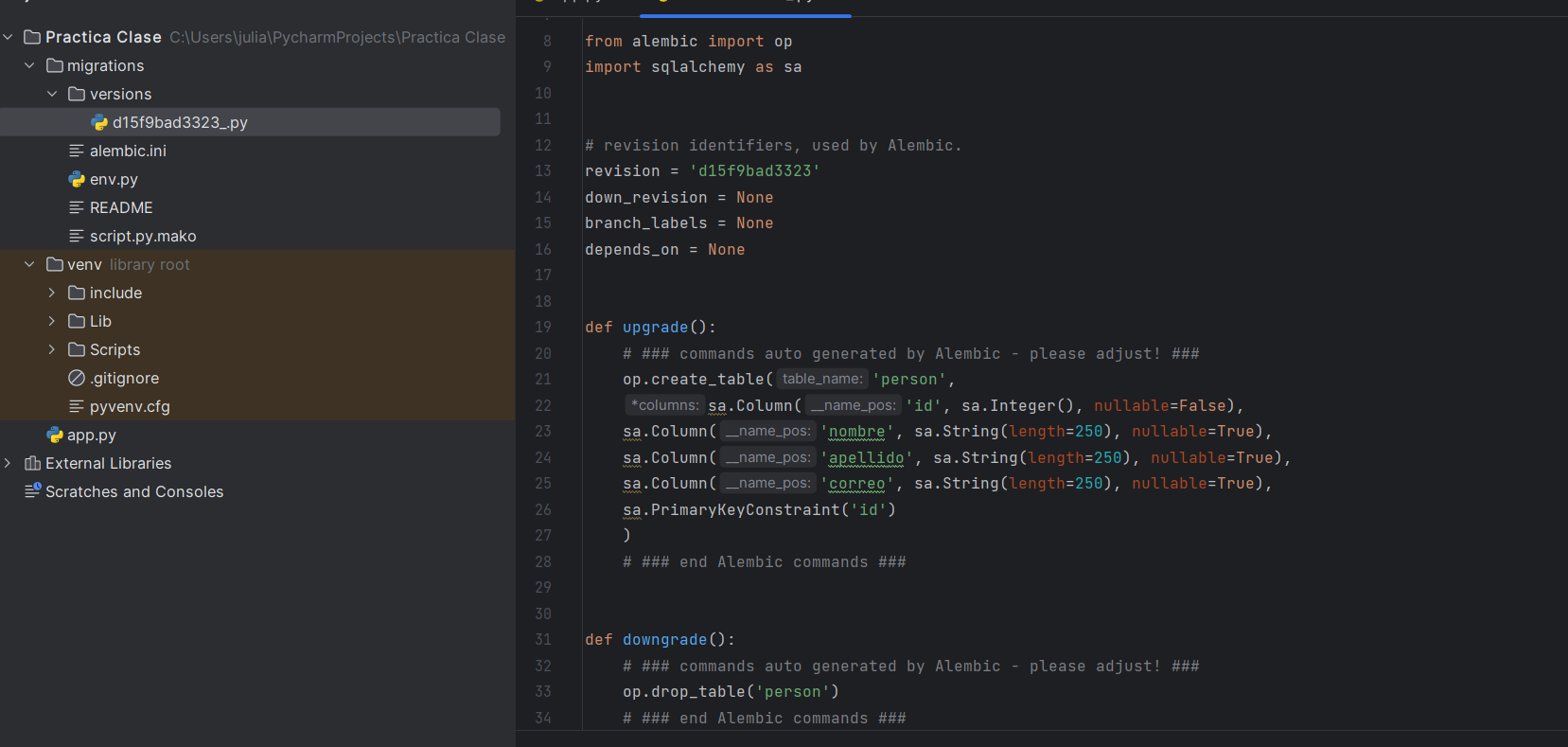
Preparar el entorno para que puedas utilizar la extensión Flask-Migrate para crear, aplicar y gestionar cambios en el esquema de la base de datos de manera controlada y eficiente



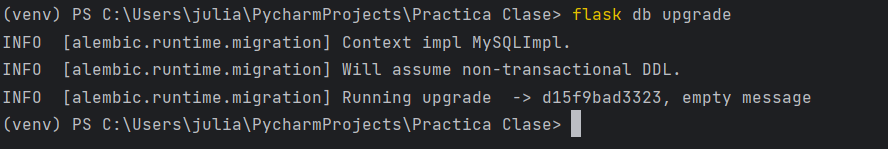
Usar el comando flask db migrate, es utilizado en conjunto con la extensión Flask-Migrate para generar automáticamente archivos de migración de base de datos en una aplicación Flask que utiliza SQLAlchemy. Estos archivos de migración contienen instrucciones para aplicar cambios en el esquema de la base de datos, como la creación de nuevas tablas, la adición de columnas o la modificación de tipos de datos.



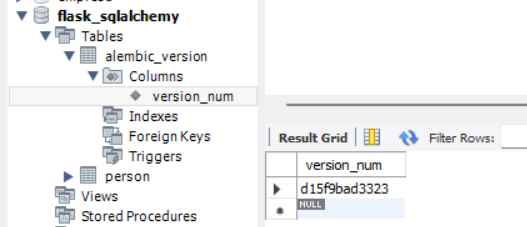




Usar el comando flask db upgrade, es utilizado en conjunto con la extensión Flask-Migrate para aplicar las migraciones de base de datos generadas previamente en una aplicación Flask que utiliza SQLAlchemy.





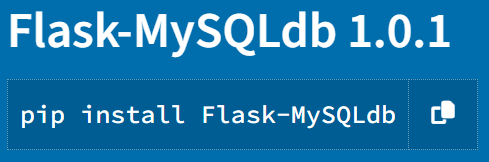




1. **Ingresa a territorium y adjunta este archivo con las preguntas resueltas, además del link del repositorio**

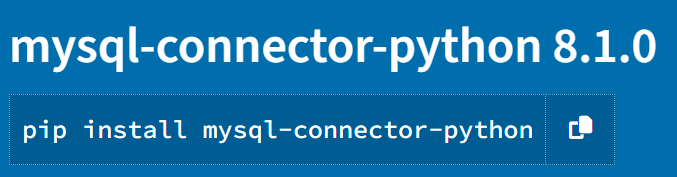
**Respuestas Pregunta 2:**

**Python – m pip install flask-mysqldb**

****

**Respuesta driver del motor de la base de datos:**

**Python -m pip install mysql-connector-python**

****

**sqlalchemy.exc.OperationalError: (MySQLdb.OperationalError) (1045, "Access denied for user 'root'@'localhost' (using password: YES)")**

**(Background on this error at: https://sqlalche.me/e/20/e3q8)**