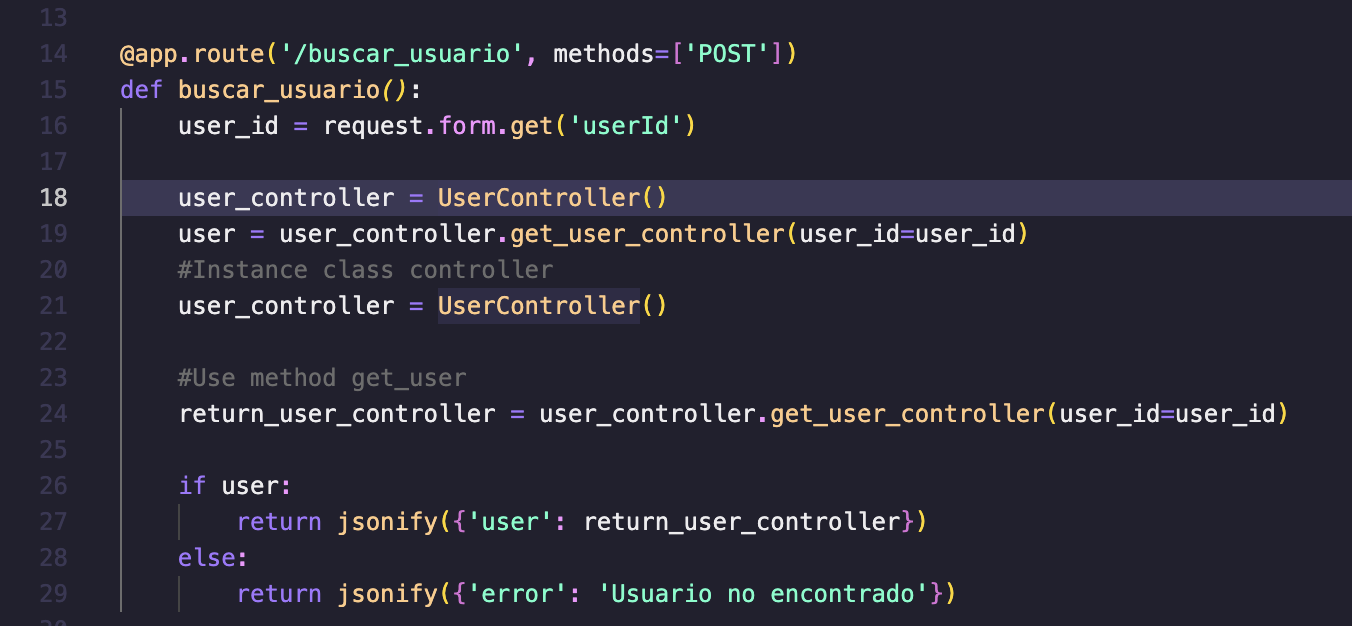
# Code Smells

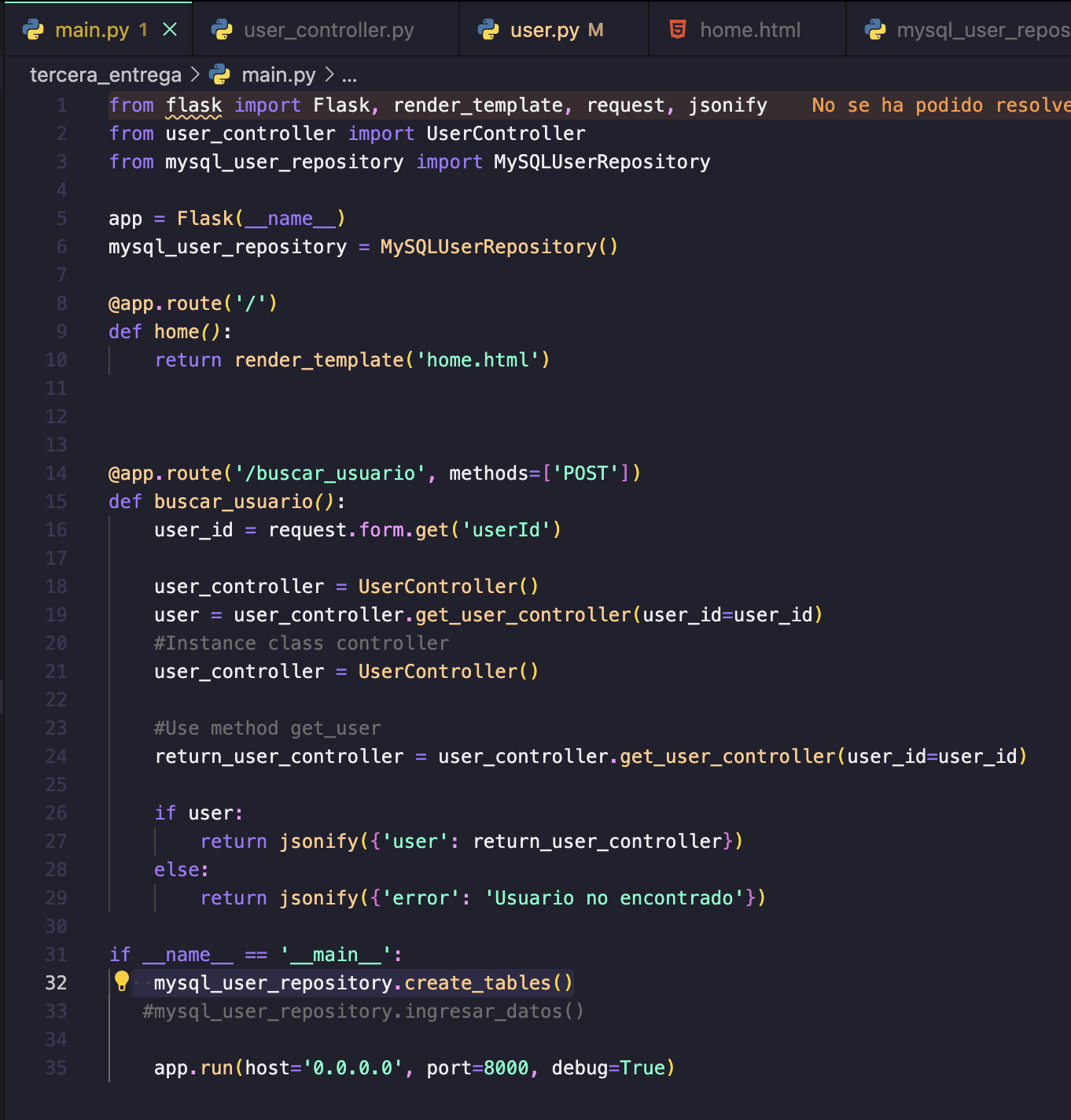


**Código duplicado:**

Las líneas 18 y 21 están duplicadas, las dos instancian el UserController().

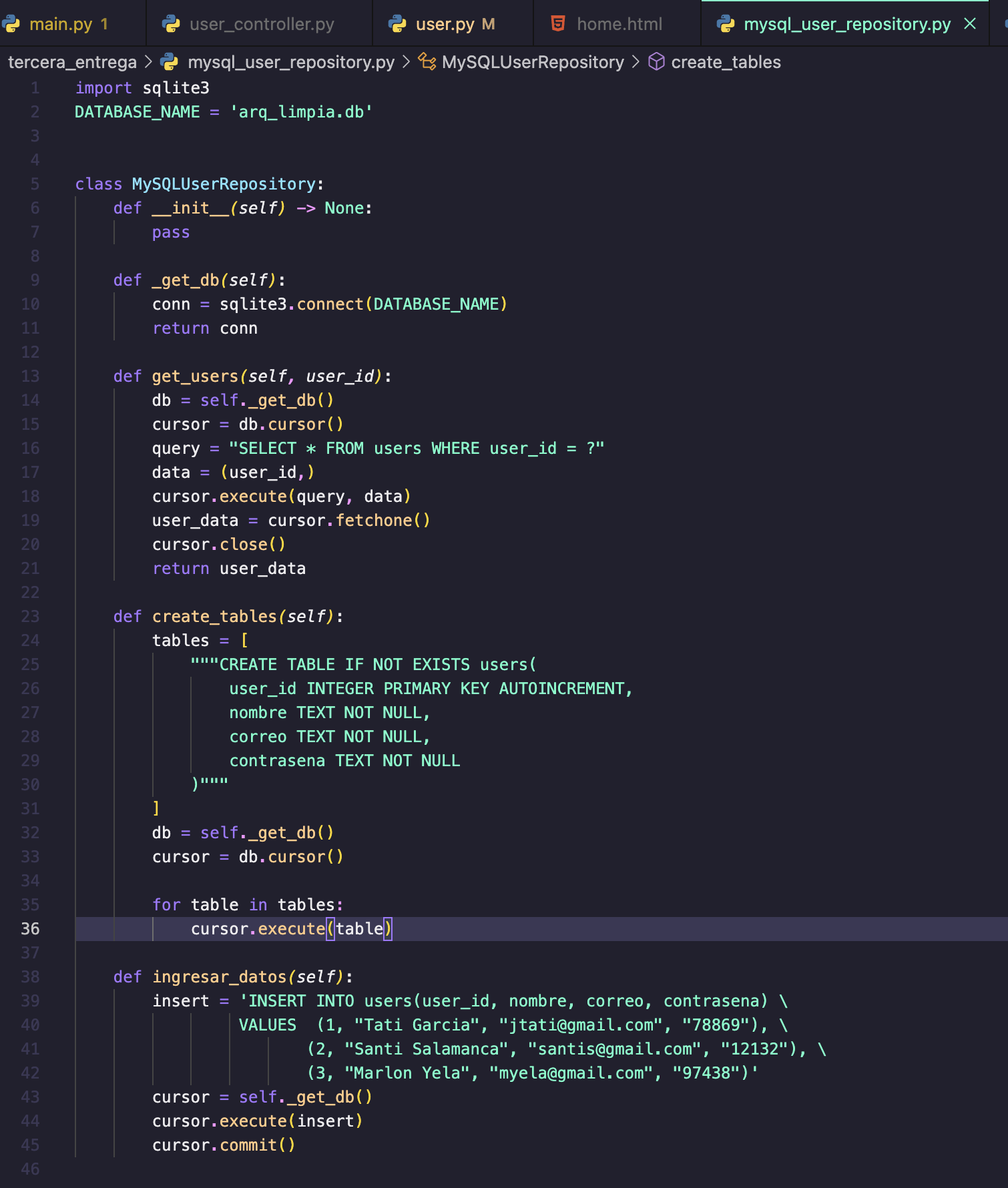
Las líneas 19 y 24 están duplicadas, los dos usan el método ‘*get\_user\_controller(user\_id)’.*

En la línea 24 el nombre del método *get\_user\_controller(user\_id)* no es claro, parece estar devolviendo un controlador, pero en realidad está devolviendo un array con los resultados de la consulta del usuario.



**Creación de tablas en el punto de entrada principal**: La creación de tablas se realiza directamente en el punto de entrada principal del programa (if \_\_name\_\_ == '\_\_main\_\_':). Esto puede no ser ideal, ya que generalmente se prefiere separar la lógica de inicialización de la base de datos de la lógica de la aplicación.

**Código comentado**: Hay código comentado en la línea 33. Esto puede hacer que el código sea confuso para leer y mantener. Si el código no es necesario, es mejor eliminarlo. Si es necesario en el futuro, se puede recuperar del control de versiones.



1. **Datos codificados**: Los datos de usuario están codificados directamente en el método ingresar\_datos. Esto hace que el método sea inflexible y difícil de reutilizar para diferentes usuarios. Sería mejor pasar los datos de usuario como parámetros.
2. **Falta de manejo de errores**: No hay manejo de errores para las operaciones de base de datos. Si la creación de la tabla o la inserción de datos falla por alguna razón, el programa se bloqueará.
3. **Falta de cierre de recursos**: El cursor de la base de datos no se cierra después de su uso. Esto puede llevar a fugas de recursos.
4. **Métodos de gran tamaño**: El método create\_tables es bastante grande y hace varias cosas (crea una lista de tablas, obtiene una base de datos, crea un cursor, itera sobre las tablas). Podría ser útil dividir este método en varios métodos más pequeños, cada uno con una responsabilidad única.
5. **Inyección SQL**: El código actual puede ser vulnerable a la inyección SQL, ya que los valores se insertan directamente en la consulta SQL. Sería mejor usar consultas parametrizadas para evitar esto.
6. **Nombres de métodos no descriptivos**: El nombre del método ingresar\_datos es bastante genérico. Un nombre más descriptivo podría hacer que el código sea más fácil de entender y mantener.