**Laboratorio Práctico de Inyección SQL con Python y SQLite**

**Objetivos del Laboratorio**

* Comprender cómo funciona la inyección SQL.
* Identificar vulnerabilidades en aplicaciones web desarrolladas con Python.
* Implementar medidas de protección contra inyecciones SQL.

**Requisitos Previos**

* Conocimientos básicos de SQL y bases de datos.
* Familiaridad con Python y las bibliotecas relacionadas con bases de datos (como sqlite3).
* Un entorno de desarrollo de Python configurado (como Visual Studio Code o PyCharm).

**Parte 1: Configuración del Entorno**

1. **Crear una Base de Datos SQLite Vulnerable**:
   * Crea un archivo llamado setup\_db.py y ejecuta el siguiente código para crear una base de datos usuarios.db con una tabla de usuarios:



1. **Crear una Aplicación Python Vulnerable**:

* Crea un archivo app.py que contenga un código Python simple para simular un inicio de sesión vulnerable:



**Parte 2: Realización de la Inyección SQL**

1. **Prueba de inyección básica**:
   * Ejecuta app.py y, cuando se te pida el nombre de usuario, ingresa lo siguiente:

**' OR '1'='1**

 Ingresa cualquier cosa o deja el campo de contraseña en blanco.

 Observa cómo el script imprime "Inicio de sesión exitoso", lo que indica que la inyección fue exitosa.

**Parte 3: Implementación de Medidas de Protección**

1. **Usar consultas preparadas (consultas parametrizadas)**:
   * Modifica la función login en app.py para usar consultas seguras:

