

## PARTE 1: CONCEPTOS Y CREACION DE LA BASE DE DATOS.

¿Qué hace el comando `sqlite3.connect('productos.db')`?

RTA: `sqlite3` proporciona el acceso a las bases de datos `sqlite`, `connect` establece la conexión a la base de datos, si el archivo no existe lo creará y si ya existe solo creará la conexión con este. En este caso lo crea.

¿Cuál es la funcionalidad de `c = conn.cursor()` y por qué motivo no se ejecutan los comandos sobre el objeto `conn`?

RTA: El método `cursor` permite leer la consulta fila por fila, facilitando las operaciones individuales, no se ejecuta ningún comando sobre el objeto ya que no se está consultando nada, solo se está estableciendo una conexión con la base de datos.

¿Dentro del código para crear la tabla aparece `INTEGER PRIMARY KEY`, ¿qué funciones cumple estos comandos?

RTA: Definir el `INTEGER` permite que el valor ingresado solo sea un número entero y definir `PRIMARY KEY` indica que esta es la llave primaria de la tabla, un valor único e irrepetible para cada registro de esta tabla.

¿Dónde se insertan los 10.000 productos se usa el comando `c.executemany()`, ¿cuál es la principal ventaja de este método en lugar de un `for` que llame a `c.execute` repetidamente?

RTA: Usar el `executemany` mejora la eficiencia y el rendimiento realizando el trabajo en una sola operación y el bucle debe estar conectando a cada momento lo que lo hace más eficiente.

¿Qué pasa si se olvida escribir `conn.commit()` al final del código por el programador?

RTA: Las operaciones realizadas en la base de datos no quedarán de forma permanente y solo serán registradas al momento de ejecutar el código.

## PARTE 2:

Escribir SQL para atraer campos de Proveedor1

- `Select nombre, categoría, precio from Proveedor1;`

¿Qué ventajas ofrece `pd.read_sql_query()` en el código?

RTA: Integra directamente la ejecución de consultas directamente en la conexión sin la necesidad de pasos adicionales.

## PARTE 3:

¿Por qué se usan los signos de interrogación en las inserciones de SQL en el código?

RTA: Separan los datos al ejecutar la conexión y evita inyecciones en la base de datos al marcar cada posición siendo espacios reservados para los datos.

¿Por qué es necesario validar los datos antes de realizar la inserción en la base de datos?

RTA: Para evitar errores con los tipos de datos en la base de datos y evitar inyecciones en la base de datos al anteriormente haber validado en el código.

- ERROR EN LA INSERCIÓN DE DATOS.

En el código el error sucede porque se está agregando `id`, esto se soluciona con colocar `None` en los valores ingresados o directamente quitar el campo de inserción al hacer el campo `AUTOINCREMENT` entonces no sería necesario agregar nada.

- CONSULTA BASE DE DATOS CON OPERADOR.

Select (precio \* cantidad) AS total from inventario;