

# Conectar Python a MySQL



Ayúdame a crear un programa de Python para conectarse a una MySQL



## Memoria actualizada

Claro, puedo ayudarte con eso. Para conectarte a una base de datos MySQL desde Python, necesitas utilizar un módulo llamado `mysql-connector-python`. Si aún no lo tienes instalado, puedes instalarlo usando pip:

```
bash
```

```
pip install mysql-connector-python
```

Una vez que lo hayas instalado, aquí te muestro cómo puedes conectar y ejecutar consultas en una base de datos MySQL desde Python:

```
python
```

```
import mysql.connector

# Establecer la conexión
conexion = mysql.connector.connect(
    host="tu_host",
    user="tu_usuario",
    password="tu_contraseña",
    database="tu_base_de_datos"
)

if conexion.is_connected():
    print("Conexión establecida correctamente")

    # Crear un cursor para ejecutar consultas
    cursor = conexion.cursor()

    # Ejecutar una consulta (ejemplo)
    cursor.execute("SELECT * FROM tu_tabla")

    # Obtener resultados
    resultados = cursor.fetchall()

    # Mostrar los resultados
    for fila in resultados:
        print(fila)

    # Cerrar el cursor y la conexión
    cursor.close()
    conexion.close()
```

```
print("Conexión cerrada")
else:
    print("Error al conectar a la base de datos")
```

Asegúrate de reemplazar `tu\_host`, `tu\_usuario`, `tu\_contraseña`, `tu\_base\_de\_datos` y `tu\_tabla` con los valores correctos para tu base de datos MySQL. Este código establece una conexión, ejecuta una consulta de ejemplo y muestra los resultados. Puedes adaptarlo según tus necesidades específicas. ¿Hay algo en particular que te gustaría ajustar o explorar más?



Ahora quiero consultar los registros de la tabla curso y que los convierta en formato Dataframe y los exporte a un Excel



Para consultar los registros de la tabla `curso`, convertirlos en un DataFrame de pandas y luego exportarlos a un archivo Excel, necesitarás utilizar las bibliotecas `pandas` y `mysql-connector-python`. Asegúrate de tener instaladas ambas bibliotecas antes de comenzar. Aquí tienes un ejemplo de cómo hacerlo:

python

```
import pandas as pd
import mysql.connector

# Establecer la conexión a MySQL
conexion = mysql.connector.connect(
    host="tu_host",
    user="tu_usuario",
    password="tu_contraseña",
    database="tu_base_de_datos"
)

if conexion.is_connected():
    print("Conexión establecida correctamente")

    # Consultar datos desde MySQL y cargarlos en un DataFrame
    query = "SELECT * FROM curso"
    df = pd.read_sql(query, con=conexion)

    # Mostrar los primeros registros del DataFrame (opcional)
    print(df.head())

    # Exportar el DataFrame a un archivo Excel
    nombre_archivo = "registros_curso.xlsx"
    df.to_excel(nombre_archivo, index=False)
    print(f"Datos exportados exitosamente a '{nombre_archivo}'")

    # Cerrar la conexión
    conexion.close()
    print("Conexión cerrada")
```

```
else:
    print("Error al conectar a la base de datos")
```

Asegúrate de reemplazar `tu\_host`, `tu\_usuario`, `tu\_contraseña` y `tu\_base\_de\_datos` con tus propias credenciales y nombre de base de datos. Este script primero consulta todos los registros de la tabla `curso`, los carga en un DataFrame de pandas y luego los exporta a un archivo Excel llamado `registros\_curso.xlsx` en el directorio actual.

Si tienes alguna otra consulta o necesitas más ayuda, ¡déjame saber!



Como obtengo los nombres de los campos del dataframe que se genera.



Cuando obtienes datos de una tabla MySQL y los cargas en un DataFrame de pandas, los nombres de las columnas del DataFrame corresponden a los nombres de los campos (columnas) de la tabla en la base de datos. Aquí te muestro cómo puedes obtener los nombres de las columnas de tu DataFrame después de cargar los datos desde MySQL:

python

```
import pandas as pd
import mysql.connector

# Establecer la conexión a MySQL
conexion = mysql.connector.connect(
    host="tu_host",
    user="tu_usuario",
    password="tu_contraseña",
    database="tu_base_de_datos"
)

if conexion.is_connected():
    print("Conexión establecida correctamente")

    # Consultar datos desde MySQL y cargarlos en un DataFrame
    query = "SELECT * FROM curso"
    df = pd.read_sql(query, con=conexion)

    # Obtener nombres de las columnas del DataFrame
    nombres_columnas = df.columns.tolist()
    print("Nombres de las columnas:")
    print(nombres_columnas)

    # Exportar el DataFrame a un archivo Excel
    nombre_archivo = "registros_curso.xlsx"
    df.to_excel(nombre_archivo, index=False)
    print(f"Datos exportados exitosamente a '{nombre_archivo}'")

    # Cerrar la conexión
    conexion.close()
    print("Conexión cerrada")
```

```
else:  
    print("Error al conectar a la base de datos")
```

En este código, después de cargar los datos en `df` utilizando `pd.read_sql`, la línea `df.columns.tolist()` obtiene los nombres de las columnas como una lista de Python. Luego, esos nombres se imprimen en la consola para que puedas verlos. Esto te permitirá verificar que los nombres de las columnas en el DataFrame coincidan con los nombres de los campos en la tabla `curso` de tu base de datos MySQL.

Si tienes más preguntas o algo más en lo que pueda ayudarte, házmelo saber.

¿Esta conversación te resultó útil hasta ahora?

ChatGPT puede cometer errores. Comprueba la información importante.