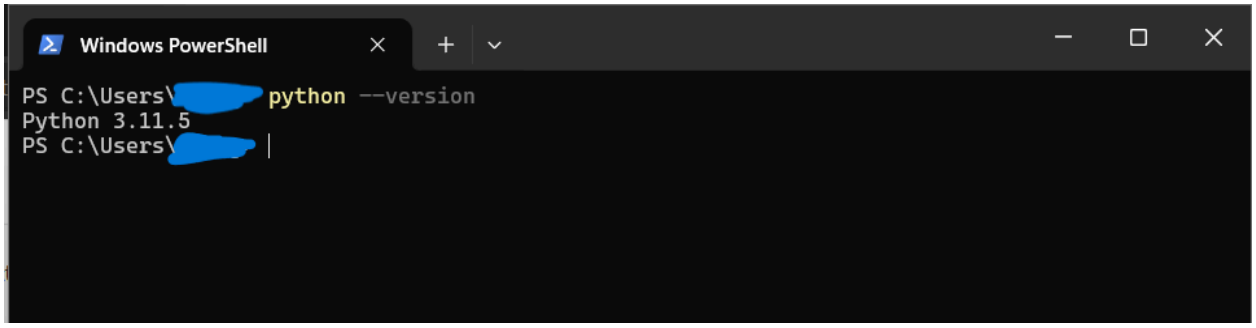
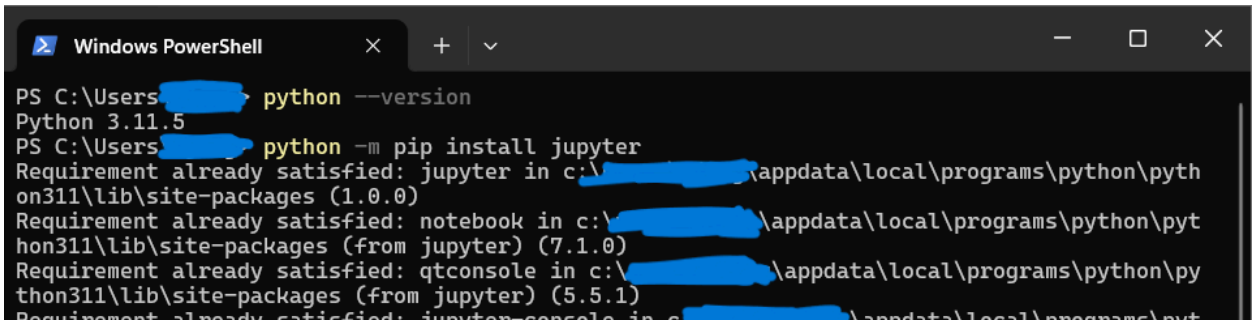
SET UP THE CLIENT PROJECT

Task 1:
Your first task is to navigate to your terminal and ensure that Python is installed and available on the command path.



Task 2:
Having established that an up-to-date version of python is installed on your machine you will need to install Jupyter.



Run Jupyter notebook.

Task 3:
Your third and final task is to establish a connection between Python and your database.

```
[1]: pip install mysql-connector-python

     Requirement already satisfied: mysql-connector-python in c:\users\kjang\appdata\local\programs\python\python311\lib\site-packages (8.3.0)
     Note: you may need to restart the kernel to use updated packages.

[2]: import mysql.connector as connector

[3]: connection = connector.connect(user = "db-capstone-user", password = "[redacted])

[ ]:
```

ADD QUERY FUNCTIONS

Task 1:
In the first task of this exercise, you are tasked with extending the environment to connect with your database and interact with the data it holds.

```
[4]: connection = connector.connect(user = "db-capstone-user", password = "[redacted]", db="littlelemondb")
```

Task 2:
In this second task, you now need to query the database to show all tables within the database.

```
[5]: show_tables_query = "SHOW tables"

[6]: cursor = connection.cursor()

[7]: cursor.execute(show_tables_query)

[8]: results = cursor.fetchall()

[9]: print(results)

     [('bookings',), ('customers',), ('menu',), ('menu items',), ('order delivery status',), ('orders',), ('ordersview',), ('staff',)]

[10]: for result in results:
          print(result)

      ('bookings',)
      ('customers',)
      ('menu',)
      ('menu items',)
      ('order delivery status',)
      ('orders',)
      ('ordersview',)
      ('staff',)
```

Task 3:

Query with table JOIN

For the third and final task, Little Lemon needs you to return specific details from your database. They require the full name and contact details for every customer that has placed an order greater than $60 for a promotional campaign.

[11]:
```
query_with_join = '''
    SELECT c.Customer_name AS "Customer Name", c.Contact_no AS "Contact Details", o.Total_cost AS "Total Cost" FROM
    customers c
    INNER JOIN bookings b ON c.Customer_id = b.Customer_id
    INNER JOIN orders o ON b.Booking_id = o.Booking_id
    WHERE o.Total_cost > 60;
'''
```

[12]:
```
cursor.execute(query_with_join)
```

[13]:
```
results = cursor.fetchall()
```

[15]:
```
print(cursor.column_names)
for result in results:
    print(result)
```

```
('Customer Name', 'Contact Details', 'Total Cost')
('John', 89124519, Decimal('245.75'))
('Steve', 89123748, Decimal('98.50'))
('John', 89124519, Decimal('72.30'))
```