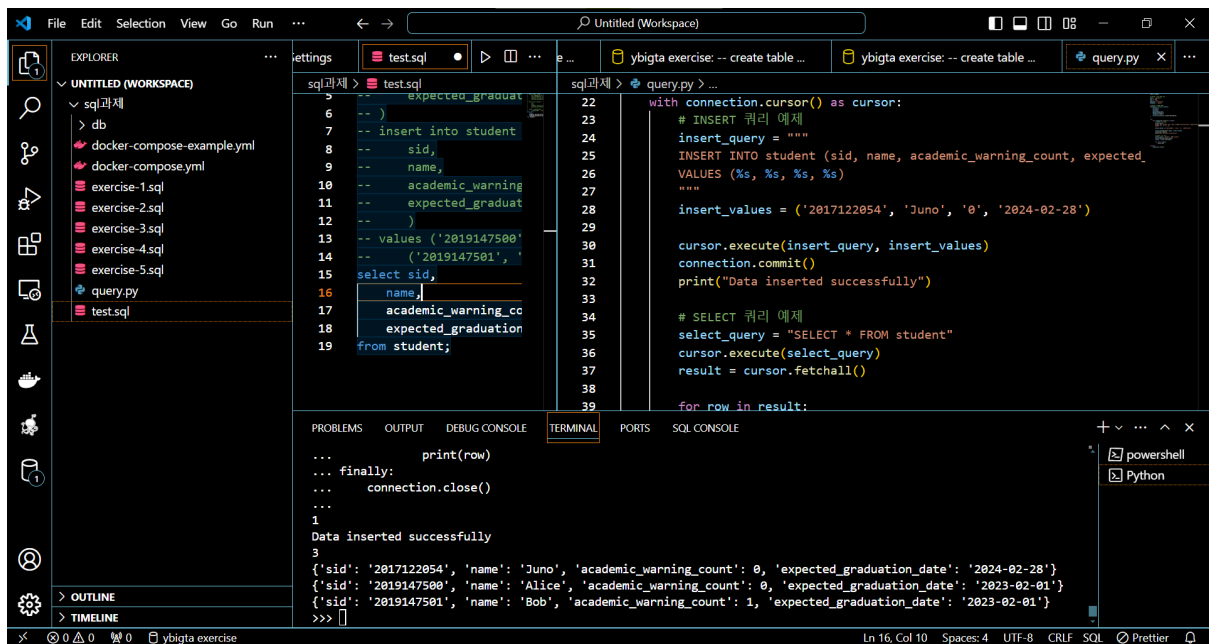


Report

김준호

PyMySQL을 이용하여 INSERT와 SELECT 쿼리를 구현했다. 실습 때 만든 student 테이블에 그대로 나의 학부 정보를 INSERT하고, SELECT하는 방식으로 코드를 구성했다.



```
File Edit Selection View Go Run ...
Untitled (Workspace)

EXPLORER
  UNTITLED (WORKSPACE)
  sql과제
    db
      docker-compose-example.yml
      docker-compose.yml
      exercise-1.sql
      exercise-2.sql
      exercise-3.sql
      exercise-4.sql
      exercise-5.sql
      query.py
      test.sql

test.sql
  5 -- expected_graduation_date
  6 -- )
  7 -- insert into student
  8 --   sid,
  9 --   name,
 10 --   academic_warning_count,
 11 --   expected_graduation_date
 12 -- )
 13 -- values ('2019147500',
 14 --         ('2019147501',
 15 --         name,
 16 --         academic_warning_count,
 17 --         expected_graduation_date
 18 -- )
 19 from student;

query.py
  22 with connection.cursor() as cursor:
  23     # INSERT 쿼리 예제
  24     insert_query = """
  25     INSERT INTO student (sid, name, academic_warning_count, expected_
  26     VALUES (%s, %s, %s, %s)
  27     """
  28     insert_values = ('2017122054', 'Juno', '0', '2024-02-28')
  29
  30     cursor.execute(insert_query, insert_values)
  31     connection.commit()
  32     print("Data inserted successfully")
  33
  34     # SELECT 쿼리 예제
  35     select_query = "SELECT * FROM student"
  36     cursor.execute(select_query)
  37     result = cursor.fetchall()
  38
  39     for row in result:
  40         print(row)
  41     finally:
  42         connection.close()
  43     ...
  44     1
  45     Data inserted successfully
  46     3
  47     {'sid': '2017122054', 'name': 'Juno', 'academic_warning_count': 0, 'expected_graduation_date': '2024-02-28'}
  48     {'sid': '2019147500', 'name': 'Alice', 'academic_warning_count': 0, 'expected_graduation_date': '2023-02-01'}
  49     {'sid': '2019147501', 'name': 'Bob', 'academic_warning_count': 1, 'expected_graduation_date': '2023-02-01'}
  50     >>> ]

TERMINAL
  ...
  ... finally:
  ...     connection.close()
  ...
  ...
  1
  Data inserted successfully
  3
  {'sid': '2017122054', 'name': 'Juno', 'academic_warning_count': 0, 'expected_graduation_date': '2024-02-28'}
  {'sid': '2019147500', 'name': 'Alice', 'academic_warning_count': 0, 'expected_graduation_date': '2023-02-01'}
  {'sid': '2019147501', 'name': 'Bob', 'academic_warning_count': 1, 'expected_graduation_date': '2023-02-01'}
  >>> ]

Ln 16, Col 10 Spaces: 4 UTF-8 CRLF SQL Prettier
```

위 화면에서 query.py를 실행한 결과, Data inserted successfully가 나오고, 성공적으로 INSERT된 student 테이블을 SELECT한 결과 화면을 확인해 볼 수 있다.

보람차고 재미있는 과제였다. 다음에는 SQLAlchemy도 한 번 사용해 볼 계획이다.

다음 장에 내용 더 있음...

The screenshot shows a SQL IDE interface with a workspace titled 'Untitled (Workspace)'. The Explorer panel on the left shows a project structure with files like 'docker-compose-example.yml', 'docker-compose.yml', 'exercise-1.sql', 'exercise-2.sql', 'exercise-3.sql', 'exercise-4.sql', 'exercise-5.sql', 'query.py', and 'test.sql'. The main editor displays the contents of 'test.sql', which contains SQL code for creating a table, inserting data, and selecting it. The SQL code is as follows:

```
-- create table ...
-- expected_graduation_date VARCHAR(10)
--
-- insert into student (
--   sid,
--   name,
--   academic_warning_count,
--   expected_graduation_date
-- )
-- values ('2019147500', 'Alice', 0, '2023-02-01')
-- ('2019147501', 'Bob', 1, '2023-02-01')
select sid,
       name,
       academic_warning_count,
       expected_graduation_date
from student;
```

The SQL Console on the right shows the results of the query. It displays a table with 4 columns: 'sid', 'name', 'academic warni...', and 'expected_grad'. The table contains 3 rows of data:

sid	name	academic warni...	expected_grad
2017122054	Juno	0	2024-02-28
2019147500	Alice	0	2023-02-01
2019147501	Bob	1	2023-02-01

The Terminal panel at the bottom shows the output of the SQL execution, indicating that the data was inserted successfully and displaying the results of the SELECT query:

```
... print(row)
... finally:
... connection.close()
...
1
Data inserted successfully
3
{'sid': '2017122054', 'name': 'Juno', 'academic_warning_count': 0, 'expected_graduation_date': '2024-02-28'}
{'sid': '2019147500', 'name': 'Alice', 'academic_warning_count': 0, 'expected_graduation_date': '2023-02-01'}
{'sid': '2019147501', 'name': 'Bob', 'academic_warning_count': 1, 'expected_graduation_date': '2023-02-01'}
```

혹시 몰라 test.sql에서도 다시 한번 SELECT를 해 보았다. 역시나 성공적으로 INSERT된 모습을 확인할 수 있다.