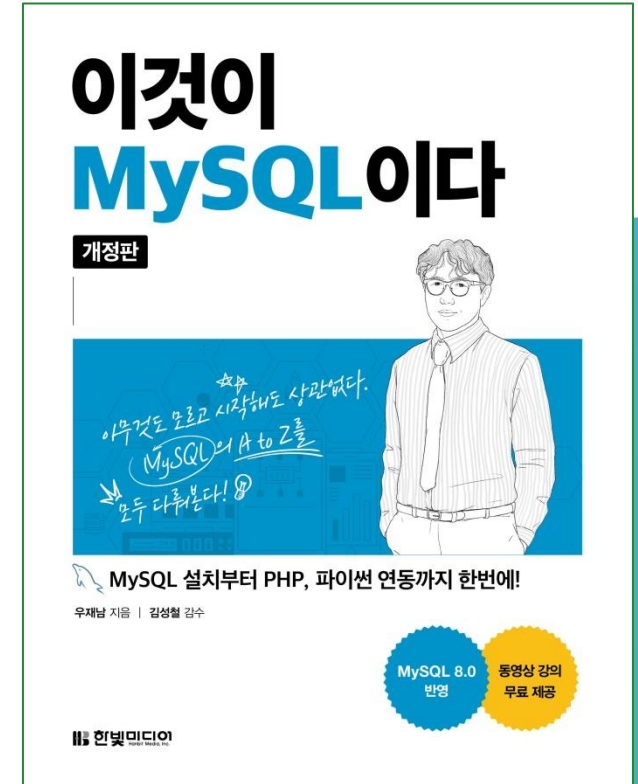


▶ Chapter 15: 파이썬과 MySQL 응용 프로그래밍

# 이것이 MySQL이다

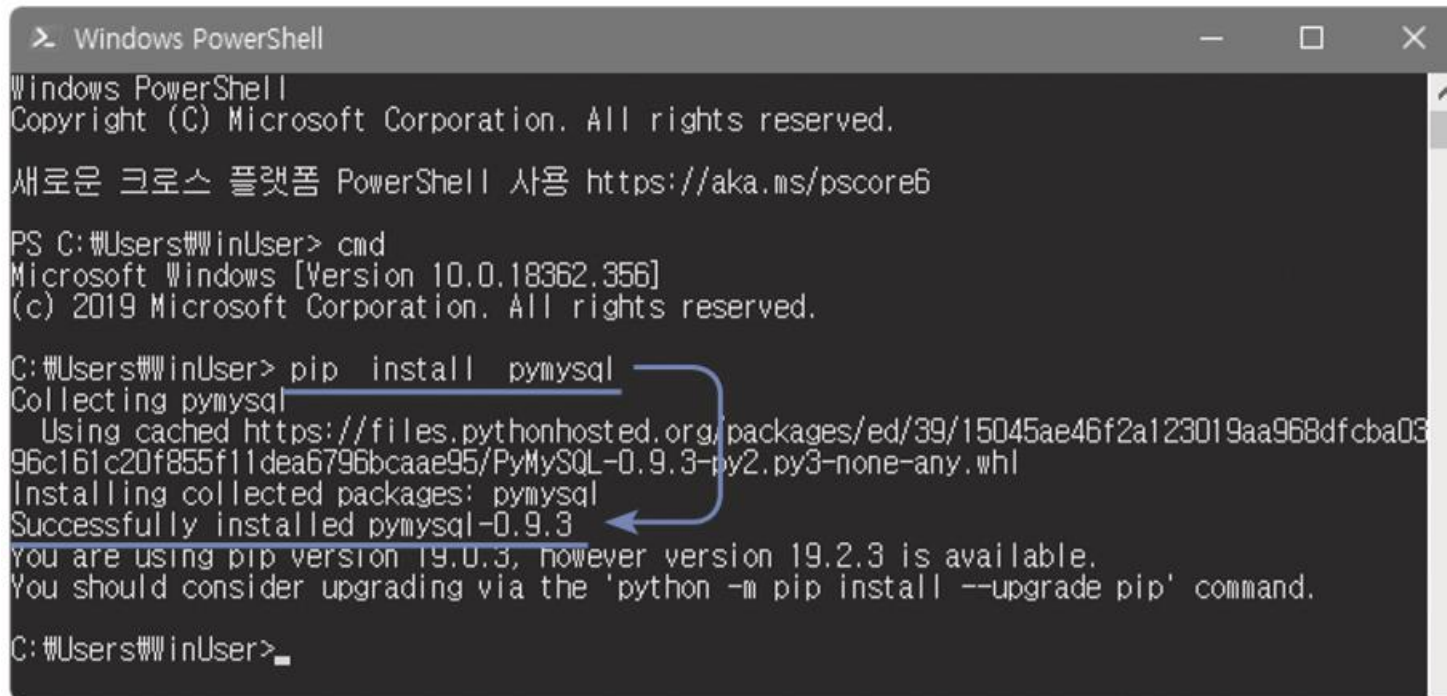


저자 : 우재남

# SECTION 03 파이썬과 MySQL 연동

## 파이썬에서 MySQL 활용

- pymysql 라이브러리 설치
  - 파이썬에는 MySQL 인식 기능 없음
  - 외부 라이브러리 pymysql 설치 해야함
  - 명령 프롬프트에 **pip install pymysql** 입력
  - import pymysql문을 통해서 MySQL과 관련된 함수 사용 가능



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

새로운 크로스 플랫폼 PowerShell 사용 https://aka.ms/pscore6

PS C:\Users\WinUser> cmd
Microsoft Windows [Version 10.0.18362.356]
(c) 2019 Microsoft Corporation. All rights reserved.

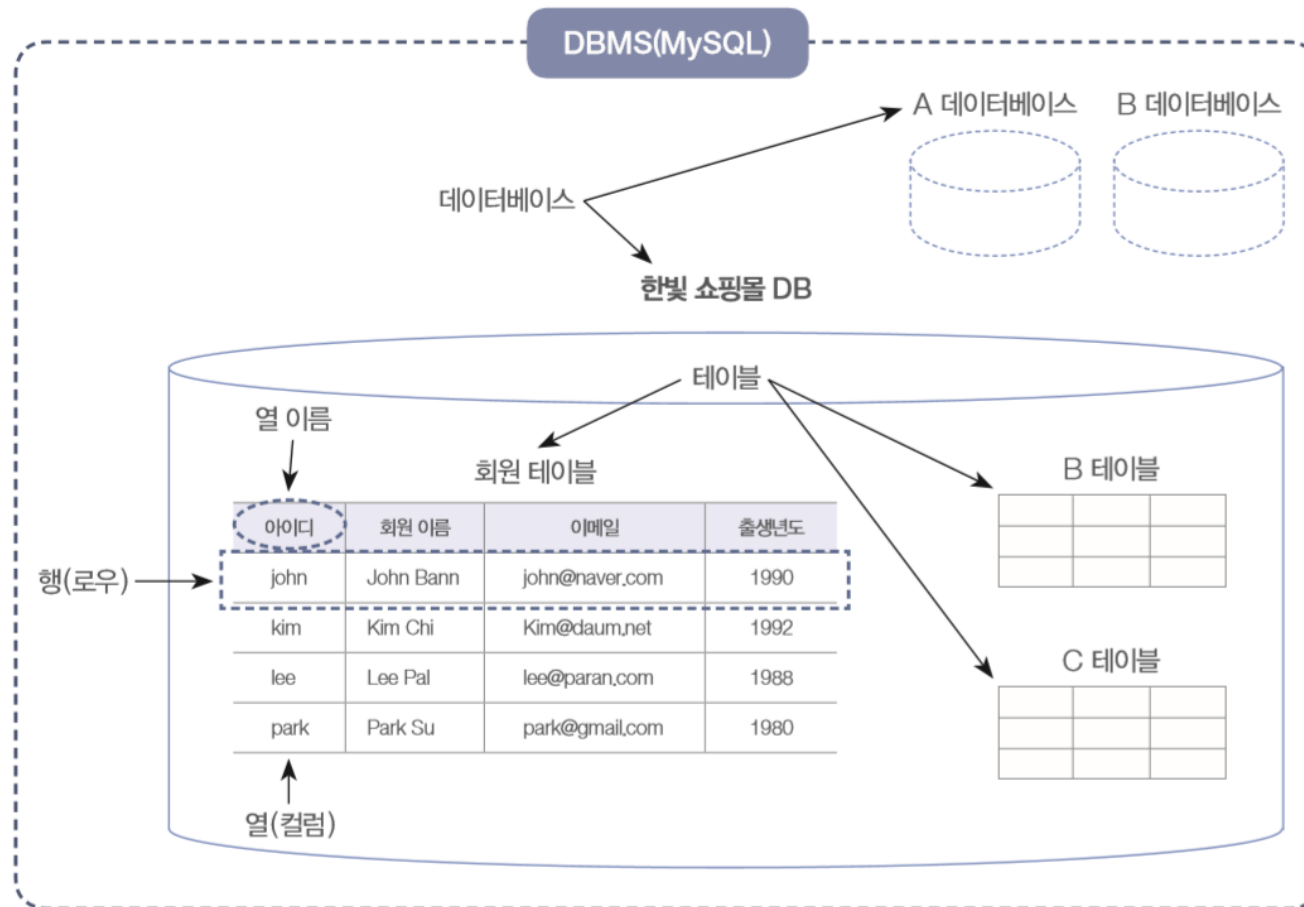
C:\Users\WinUser> pip install pymysql
Collecting pymysql
  Using cached https://files.pythonhosted.org/packages/ed/39/15045ae46f2a123019aa968dfcba0396c161c20f855f11dea6796bcaae95/PyMySQL-0.9.3-py2.py3-none-any.whl
Installing collected packages: pymysql
Successfully installed pymysql-0.9.3
You are using pip version 19.0.3, however version 19.2.3 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

C:\Users\WinUser>
```

# SECTION 03 파이썬과 MySQL 연동

## 파이썬에서 MySQL 활용

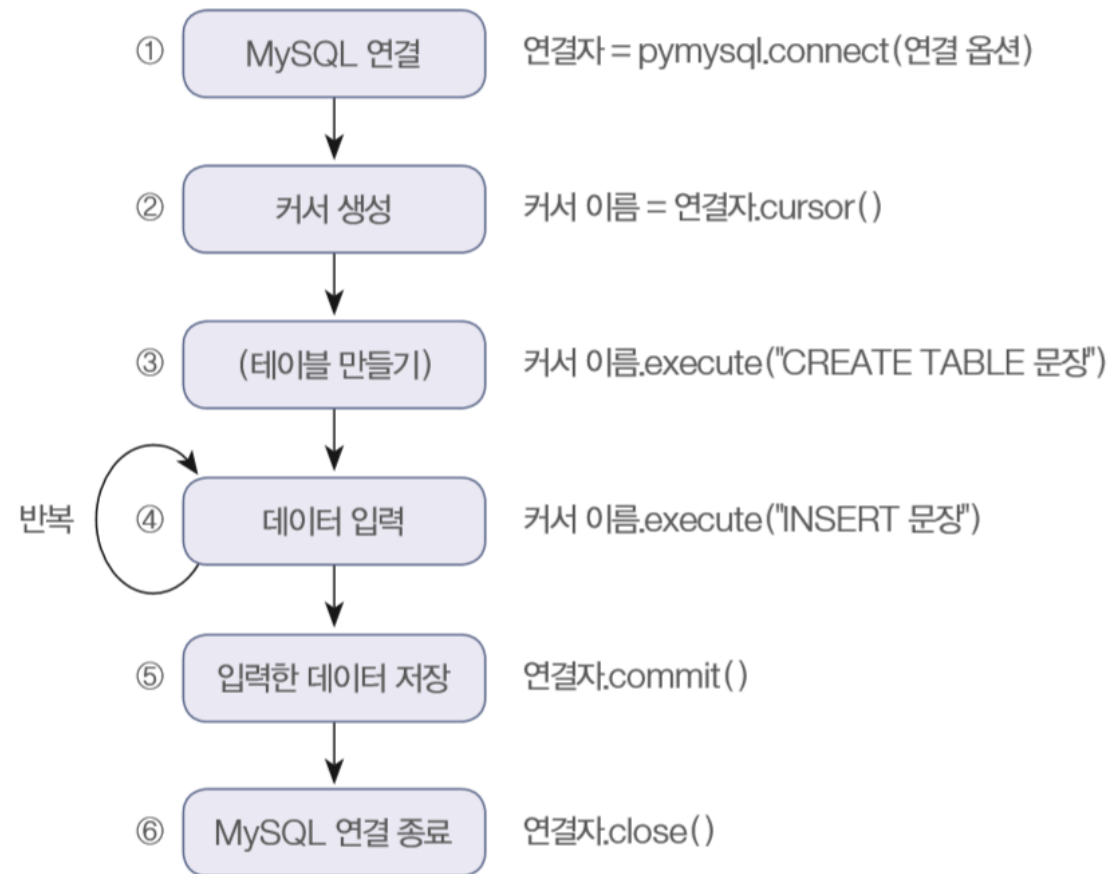
- MySQL에 데이터 **입력**을 위한 파이썬 코딩 순서
  - 데이터베이스 구축 실습



## SECTION 03 파이썬과 MySQL 연동

### 파이썬에서 MySQL 활용

- MySQL에 데이터 **입력**을 위한 파이썬 코딩 순서
  - 데이터베이스 구축 실습



## SECTION 03 파이썬과 MySQL 연동

### 파이썬에서 MySQL 활용

- 데이터베이스 구축 실습

- pymysql을 임포트한 후에, pymysql.connect (host=서버, user=사용자, password=암호, db=데이터베이스, charset=문자세트)문으로 데이터 베이스 연결

```
import pymysql
conn = pymysql.connect(host='127.0.0.1', user='root', password='1234',
                        db='hanbitDB', charset='utf8')
```

출력 결과:  
아무것도 나오지 않음

## SECTION 03 파이썬과 MySQL 연동

### 파이썬에서 MySQL 활용

- 데이터베이스 구축 실습
  - 커서 생성
    - 데이터베이스에 SQL문을 실행하거나, 실행된 결과를 돌려받는 통로로 생각

```
cur = conn.cursor()
```

출력 결과:

아무것도 나오지 않음

## SECTION 03 파이썬과 MySQL 연동

### 파이썬에서 MySQL 활용

- 데이터베이스 구축 실습

- 테이블 만들기

- 테이블을 만드는 SQL문을 커서이름.execute() 함수의 매개 변수로 넘겨주면 SQL문이 데이터베이스에 실행됨

```
cur.execute("CREATE TABLE IF NOT EXISTS userTable (id char(4), userName  
char(15), email char(20), birthYear int)")
```

출력 결과:  
번호

## SECTION 03 파이썬과 MySQL 연동

### 파이썬에서 MySQL 활용

- 데이터베이스 구축 실습
  - 데이터 입력

```
cur.execute("INSERT INTO userTable VALUES( 'john' , 'John Bann' ,  
        'john@naver.com' , 1990)")  
cur.execute("INSERT INTO userTable VALUES( 'kim' , 'Kim Chi' ,  
        'kim@daum.net' , 1992)")  
cur.execute("INSERT INTO userTable VALUES( 'lee' , 'Lee Pal' ,  
        'lee@paran.com' , 1988)")  
cur.execute("INSERT INTO userTable VALUES( 'park' , 'Park Su' ,  
        'park@gmail.com' , 1980)")
```

출력 결과:  
번호



## SECTION 03 파이썬과 MySQL 연동

### 파이썬에서 MySQL 활용

- 데이터베이스 구축 실습
  - 입력한 데이터 저장
    - 입력한 데이터는 데이터베이스에 완전히 저장된 것이 아니고 임시로 저장된 상태
    - 확실하게 저장하는 것을 커밋(commit)이라고 함

```
conn.commit()
```

출력 결과:

아무것도 나오지 않음

## SECTION 03 파이썬과 MySQL 연동

### 파이썬에서 MySQL 활용

- 데이터베이스 구축 실습
  - 데이터베이스 닫기
    - 데이터베이스를 모두 사용했다면 연결한 데이터베이스를 닫아야 함

```
conn.close()
```

출력 결과:

아무것도 나오지 않음

# SECTION 03 파이썬과 MySQL 연동

## 파이썬에서 MySQL 활용

- 데이터 입력 프로그램 작성
  - userTable에 [Enter]를 입력할 때까지 반복해서 한 행씩 데이터 입력

[소스 15-19] 데이터 입력 프로그램

```
1  import pymysql
2
3  # 전역변수 선언부
4  conn, cur = None, None
5  data1, data2, data3, data4 = "", "", "", ""
6  sql=""
7
8  # 메인 코드
9  conn = pymysql.connect(host='127.0.0.1', user='root', password='1234',
10                          db='hanbitDB', charset='utf8')
10 cur = conn.cursor()
11
```

# SECTION 03 파이썬과 MySQL 연동

## 파이썬에서 MySQL 활용

- 데이터 입력 프로그램 작성

- userTable에 [Enter]를 입력할 때까지 반복해서 한 행씩 데이터 입력

```
12 while (True) :
13     data1 = input("사용자 ID ==> ")
14     if data1 == "" :
15         break;
16     data2 = input("사용자 이름 ==> ")
17     data3 = input("사용자 이메일 ==> ")
18     data4 = input("사용자 출생년도 ==> ")
19     sql = "INSERT INTO userTable VALUES('" + data1 + "','" + data2 + "','" +
           data3 + "','" + data4 + "')"
20     cur.execute(sql)
21
22 conn.commit()
23 conn.close()
```

## SECTION 03 파이썬과 MySQL 연동

### 파이썬에서 MySQL 활용

- 데이터 입력 프로그램 작성
  - userTable에 [Enter]를 입력할 때까지 반복해서 한 행씩 데이터 입력

출력 결과:

사용자 ID ==> su ← 사용자가 입력한 값

사용자 이름 ==> Su Ji ← 사용자가 입력한 값

사용자 이메일 ==> suji@hanbit.com ← 사용자가 입력한 값

사용자 출생년도 ==> 1994 ← 사용자가 입력한 값

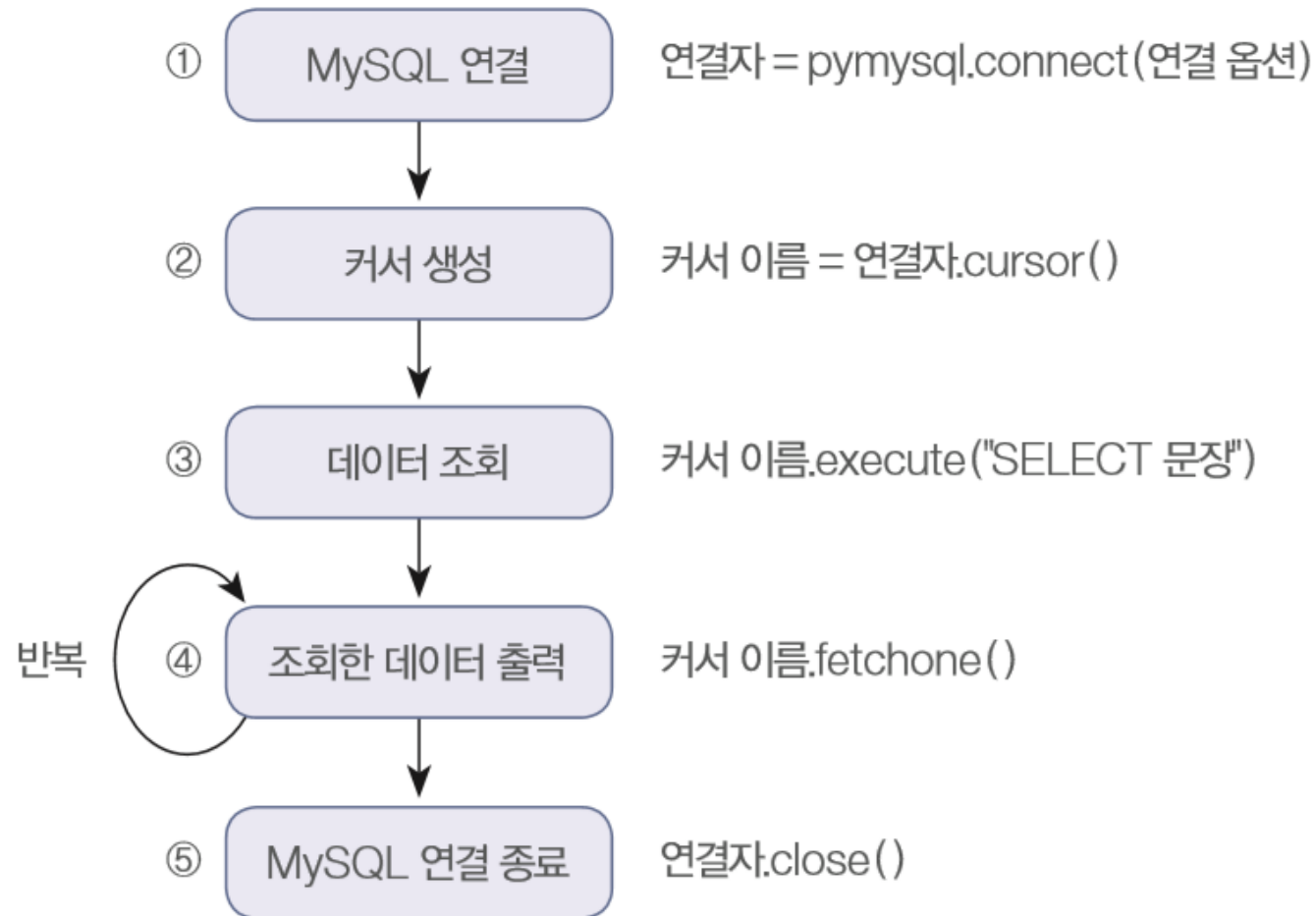
~~~ 반복해서 입력 ~~~

사용자 ID ==> ← 사용자가 입력한 값(그냥 는 종료됨)

## SECTION 03 파이썬과 MySQL 연동

### 파이썬에서 MySQL 활용

- MySQL에 데이터 조회를 위한 파이썬 코딩 순서



# SECTION 03 파이썬과 MySQL 연동

## 파이썬에서 MySQL 활용

- 데이터 조회 프로그램 작성

[소스 15-20] 데이터 조회 프로그램

```
1  import pymysql
2
3  # 전역변수 선언부
4  con, cur = None, None
5  data1, data2, data3, data4 = "", "", "", ""
6  row=None
7
8  # 메인 코드
9  conn = pymysql.connect(host='127.0.0.1', user='root', password='1234',
                          db='hanbitDB', charset='utf8')
10 cur = conn.cursor()
11
12 cur.execute("SELECT * FROM userTable")
13
```

## SECTION 03 파이썬과 MySQL 연동

### 파이썬에서 MySQL 활용

- 데이터 조회 프로그램 작성

```
14 print("사용자ID      사용자이름      이메일      출생년도")
15 print("-----")
16
17 while (True) :
18     row = cur.fetchone()
19     if row== None :
20         break
21     data1 = row[0]
22     data2 = row[1]
23     data3 = row[2]
24     data4 = row[3]
25     print("%5s      %15s      %20s      %d" % (data1, data2, data3, data4))
26
27 conn.close()
```



## SECTION 03 파이썬과 MySQL 연동

### 파이썬에서 MySQL 활용

- 데이터 조회 프로그램 작성

출력 결과:

| 사용자ID | 사용자이름     | 이메일            | 출생년도 |
|-------|-----------|----------------|------|
| john  | John Bann | john@naver.com | 1990 |
| kim   | Kim Chi   | kim@daum.net   | 1992 |
| lee   | Lee Pal   | lee@paran.com  | 1988 |
| park  | Park Su   | park@gmail.com | 1980 |

▶ 이것이 MySQL 이다

# Thank You!

