

DATABASE CLASS D-4

-I can not make 우선순위 . because I do not understand it.

1. Review

- update
-UPDATE (table_name) SET(key) = (value) WHERE (condition)
- delete
- DELETE FROM (table_name) WHERE (condition)

★★ We must write 'WHERE' when we use 'UPDATE' and 'DELETE'★★

2. FK 제약

- RESTRICT - DEFAULT 값, 제한
- CASCADE - 연쇄 동작 EX) PK 지워지면 FK도 같이 지워짐
- SET NULL - 빈 값 설정
- NO ACTION - 동작없음
- SET DEFAULT - 기본값 설정
 - 주로 RESTRICT or CASCADE 자주 사용

ex) we wanna delete PK.

step 1.	1. PK가 들어가는 모든 FK들에서 키 삭제하기 1. delete all FK had pk ALTER TABLE (table_name) DROP FOREIGN KEY (fk);
step 2.	2.CASCADE 사용하여 새로운 규칙으로 외래키 추가 new rule add FK to use CASCADE code ALTER TABLE (table_name) ADD CONSTRAINT (fk) FOREIGN KEY(pk) references key(pk) ON DELETE TABLE CASCADE;

3. Data Analysis by PyMySQL

- DB connect ⇨ cursor() ⇨ execute() ⇨ commit()

```
import pymysql
conn = pymysql.connect(
    host = ' ',
    user= ' ',
    password= ' ',
    db= ' ',
    charset= utf8m64
    cursorclass = pymysql.cursors.DictCursor
)
cursor = conn.cursor()
```

- **connect** : DB라는 식당에 입장하기 / **cursor** : 주문을 전달할 점원 고용 / **execute** : 점원에게 sql주문서 전달하기
- **cur= conn.cursor()** (점원호출), **cur.close()** (점원 퇴근시키기) => 한번에 **with conn.cursor() as cur**
- **curr.execute()** (요리 요청), **conn.commit()** (서빙확정)
 - **execute** : **fetchone()** - 결과 1행만 실행, **dict** 1개 반환
 - **execute** : **fetchall()** - 전체 결과, **list[dict]** 반환

- **JOIN + GROUP BY + ORDER BY + LIMIT**
(EX)

```
cur.execute("""
SELECT d.first_name, COUNT(*) AS visits
FROM appointments a
JOIN doctors d ON a.doctor_id = d.doctor_id
GROUP BY d.doctor_id
ORDER BY visits DESC
LIMIT 3
""")
*** CREATE , INSERT , UPDATE , DELETE 가능
```

*** python

1. Insert Update Delete	2. Commit
----------------------------------	-----------

- **COMMIT() / ROLLBACK()** 변경사항 적용/ 되돌리기

- **commit** : 확정 / **rollback()** : 취소, **execute** 도중 **error** 발생 시 전부 없던 일로 하고 처음으로 돌아감

TRY ~ EXCEPT AS - ex -

try:

```
cur.execute(
"UPDATE patient_info SET city=%s WHERE id=%s",
('Busan', 2)
)
conn.commit()
print('정상 업데이트 -> commit 완료')
```

except Exception as e:

```
conn.rollback()
print('에러발생!! 롤백 실행')
```