

Lecture 11 - Database

개요

- 복습
- 데이터베이스 설치하기
- DDL과 DML - 테이블 생성하기
- 데이터 삽입, 갱신, 삭제

복습 (10m)

csvsql 예시:

```
csvsql customer.csv --query "SELECT age, gender FROM customer;" | csvlook
```

너무 기니까 함수로 만들기:

```
query() { csvsql customer.csv product.csv tx.csv --query "$1" | csvlook; }
```

테스트:

```
query "SELECT age, gender FROM customer;"
```

기초 질의문:

```
-- 특정 칼럼만 가져오기
SELECT name, age FROM customer;

-- 여성만, 이름역순으로(DESCending) 정렬하여 첫 세 줄만(LIMIT 3)
SELECT name, age FROM customer WHERE gender='F' ORDER BY name DESC LIMIT 3;
```

조인:

```
-- tx 테이블, customer 테이블 조인
SELECT t.tid, t.qty, t.cid, c.name
FROM tx t
LEFT OUTER JOIN customer c ON (t.cid=c.cid);
```

집계:

```
-- 성별 연령대 평균, 최소, 최대값
SELECT gender, AVG(age), MIN(age), MAX(age) FROM customer GROUP by gender;
```

모두 합치기

```
-- 25세 초과인 사람들이 가장 많이 구매한 상품은?
SELECT t.pid, p.name, SUM(t.qty), SUM(p.price)
FROM tx t
INNER JOIN customer c ON (t.cid=c.cid)
INNER JOIN product p ON (t.pid=p.pid)
WHERE age > 25
GROUP BY t.pid
ORDER BY SUM(t.qty) DESC;
```

데이터베이스 설치하기 (30m)

데이터베이스란?

- DBMS(DataBase Management System; 데이터베이스 관리 시스템)
- 데이터 저장, 조회, 무결성 검사, 접근 권한 관리 등을 수행
- 여러 종류의 구현이 있음: MySQL, PostgreSQL, Microsoft SQL Server, Oracle 등
- 그럼 csvsql은?
 - sqlite라는 경량 데이터베이스 엔진을 활용하여 내부적으로 메모리 데이터베이스(In-memory DB)를 생성하는 방식
 - 작은 CSV 파일들을 다루기에 적합
 - 실무에서는 거의 쓰이지 않음. SQL 연습용(?)으로는 적절

EC2 접속

PostgreSQL 설치하기

```
sudo apt-get update
sudo apt-get install postgresql postgresql-contrib
```

사용자 전환

```
sudo -i -u postgres
```

실행

```
psql
```

빠져나가려면 Ctrl+D

데이터베이스 생성

```
createdb fastcamp
```

생성된 데이터베이스에 접속하기

```
psql -d fastcamp
```

휴식 (10m)

DDL과 DML - 테이블 생성하기 (30m)

DDL? Data Definition Language

- 테이블 생성, 변경, 삭제
- 권한 관리 등

DML? Data Manipulation Language

- 생성된 테이블에 데이터 삽입, 갱신, 삭제
- 데이터 질의(SELECT * FROM ...)

고객 테이블 생성하기

```
CREATE TABLE customer (  
    cid SERIAL PRIMARY KEY,  
    name varchar(32) NOT NULL,  
    gender char(1) NOT NULL  
);
```

생성된 테이블 확인하기

```
SELECT * FROM customer;
```

데이터 삽입, 갱신, 삭제 (30m)

데이터 삽입하기

```
INSERT INTO customer (name, gender) VALUES ('Alan Kang', 'M');
```

확인하기

```
SELECT * FROM customer;
```

데이터 갱신하기

```
UPDATE customer SET name='Allen Kang' WHERE cid=1;
```

데이터 삭제하기

```
DELETE customer WHERE cid=1;
```