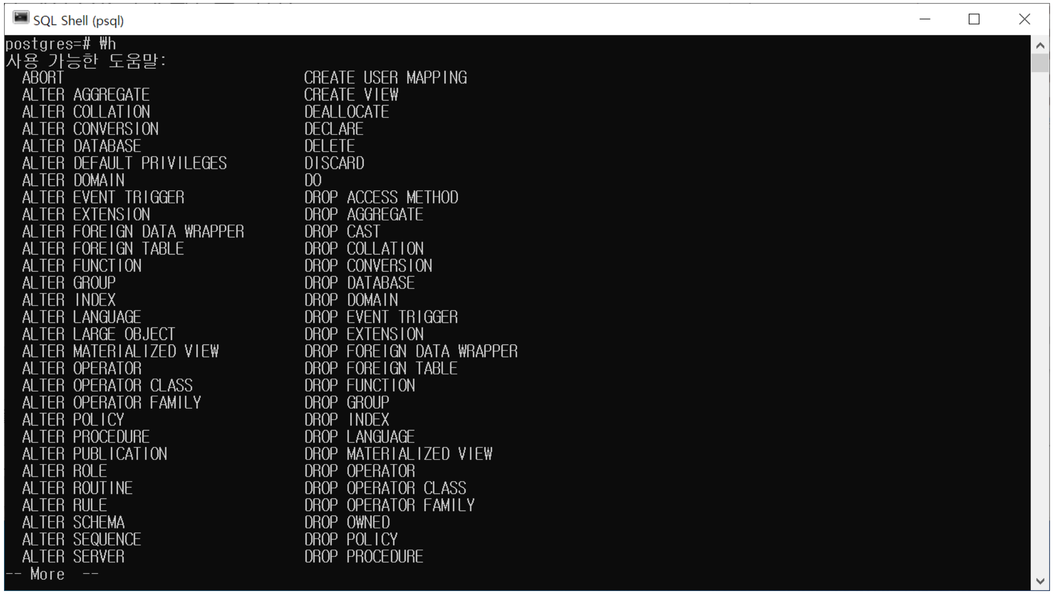
KOREA UNIVERSITY
DATABASE LAB

Lab

Chapter 3 – SQL 1/2

Useful PostgreSQL Commands

- \h: help, \h *command*: help on the command
- \d: list tables, \d *table_name*: describe table
- \i *file_name*: import SQL script
- \c *database_name*: connect to the database
- \l : list databases
- \q: quit PostgreSQL
- History 기능 제공 (위, 아래 화살표 사용)



```
postgres=# ₩h
사용 가능한 도움말:
  ABORT                            CREATE USER MAPPING
  ALTER AGGREGATE                  CREATE VIEW
  ALTER COLLATION                  DEALLOCATE
  ALTER CONVERSION                 DECLARE
  ALTER DATABASE                   DELETE
  ALTER DEFAULT PRIVILEGES         DISCARD
  ALTER DOMAIN                     DO
  ALTER EVENT TRIGGER              DROP ACCESS METHOD
  ALTER EXTENSION                  DROP AGGREGATE
  ALTER FOREIGN DATA WRAPPER       DROP CAST
  ALTER FOREIGN TABLE              DROP COLLATION
  ALTER FUNCTION                   DROP CONVERSION
  ALTER GROUP                      DROP DATABASE
  ALTER INDEX                      DROP DOMAIN
  ALTER LANGUAGE                   DROP EVENT TRIGGER
  ALTER LARGE OBJECT               DROP EXTENSION
  ALTER MATERIALIZED VIEW          DROP FOREIGN DATA WRAPPER
  ALTER OPERATOR                   DROP FOREIGN TABLE
  ALTER OPERATOR CLASS             DROP FUNCTION
  ALTER OPERATOR FAMILY            DROP GROUP
  ALTER POLICY                     DROP INDEX
  ALTER PROCEDURE                  DROP LANGUAGE
  ALTER PUBLICATION                DROP MATERIALIZED VIEW
  ALTER ROLE                       DROP OPERATOR
  ALTER ROUTINE                    DROP OPERATOR CLASS
  ALTER RULE                       DROP OPERATOR FAMILY
  ALTER SCHEMA                     DROP OWNED
  ALTER SEQUENCE                   DROP POLICY
  ALTER SERVER                     DROP PROCEDURE
-- More  --
```

Database Setup

1. Download the following sql file from blackboard
   - University.sql
2. Make university schema and insert the data into relations using sql files
   a. Execute PostgreSQL SQL Shell(psql)
   b. Create a new database using ‘CREATE DATABASE chapter3;’ command
   c. Run ‘\c chapter3’ // connection to database ‘chapter3’
   d. Run ‘\i [filepath]/University.sql’ (Don’t use whitespace or backslash ‘\’ in the filepath)
      - \i 'C:\\Users\\account\\한글 폴더\\University.sql’ (double backslash wrapped in single quotation marks)
      - 문제가 있으면 파일을 조건에 맞는 디렉토리로 옮겨서 사용

Exercise 1

- Based on the university schema, write the following queries in SQL.
  a. Find the titles of courses in the 'Comp. Sci.' department that have 3 credits.
  b. Find the IDs of all students who were taught by an instructor named 'Srinivasan'; make sure there are no duplicates in the result.
  c. Find the ID and name of instructors who have not given grades to students (i.e., where the grade is null in the *takes* relation).
  d. Find the name and department name of instructors whose department name starts with "C", listed alphabetically by department name.
  e. Find the ID and salary of the instructor(s) with the highest salary while satisfying the following conditions.
     1. Use Set Operation ("EXCEPT")
     2. Use Aggregate Function

Exercise 2

- **Make a relation** $grade\_points(\underline{grade}, points)$, which provides a conversion from letter grades in the $takes$ relation to numeric scores.
- **The tuples of the** $grade\_points$ **relation**: (A+, 4.3), (A, 4.0), (A-, 3.7), (B+, 3.3), (B, 3.0), (B-, 2.7), (C+, 2.3), (C, 2.0), (C-, 1.7), (D+, 1.3), (D, 1.0), (D-, 0.7), (F, 0.0)
- The **grade-points** a student earns for a course offering (section) are calculated by multiplying the number of credits for the course by the numeric grade points received for that course.
- For simplicity, you can assume that no $takes$ tuple has a null value for the grade attribute.
  a. Find the total grade-points earned by the student with ID 12345, across all courses taken by the student.
  b. Find the grade-point average (GPA) for the above student, that is, the total grade-points divided by the total credits for the associated courses.
     - 평균 평점 = (과목별 점수 * 과목의 학점 수) / 전체 학점 수
  c. Find the ID and the grade-points average of every student.
  d. Find the ID and the grade-points average of students whose GPA is greater than 3.0.

Homework

- Complete today’s practice exercises
- Write your queries and take screenshots of execution results
- Submit your report on blackboard
  - 10:29:59, 2024/04/11
  - **Only PDF files** are accepted
  - **No late submission**

End of Lab