



Assignment #3 (150pt)



☐ Do Exercises

- ☐ 3.11, 3.13, 3.14, 3.21, 3.25, 3.26, 3.28
- ☐ 데이터가 없는 경우 임의로 넣으세요.
- ☐ Self Study : Chapter 3.8

☐ Due: Two Weeks Later

- ☐ Before the lecture – 9/28 (Wed)

☐ Submission form

- ☐ *.doc or *.sql both okay

☐ Method: upload your report in Cyber Campus

- ☐ Questions are uploaded in Assignment 3 folder
- ☐ Answers must be written in English !



2-1 (4opt)

- ❑ **3.11** Write the following queries in SQL, using the university schema.
 - ❑ a. Find the ID and name of each student who has taken at least one Comp. Sci. course; make sure there are no duplicate names in the result.
 - ❑ b. Find the ID and name of each student who has not taken any course offered before 2017.
 - ❑ c. For each department, find the maximum salary of instructors in that department. You may assume that every department has at least one instructor.
 - ❑ d. Find the lowest, across all departments, of the per-department maximum salary computed by the preceding query.



2-2 (20pt)

- ❑ **3.13** Write SQL DDL corresponding to the schema in **Figure 3.17**. Make any reasonable assumptions about data types, and be sure to declare primary and foreign keys.

```
person (driver_id, name, address)
car (license_plate, model, year)
accident (report_number, year, location)
owns (driver_id, license_plate)
participated (report_number, license_plate, driver_id, damage_amount)
```

Figure 3.17 Insurance database



2-3 (20pt)

- ❑ **3.14** Consider the insurance database of Figure 3.17, where the primary keys are underlined. Construct the following SQL queries for this relational database.
- ❑ a. Find the number of accidents involving a car belonging to a person named “John Smith”.
 - ❑ b. Update the damage amount for the car with license plate “AABB2000” in the accident with report number “AR2197” to \$3000.



2-4 (4opt)

```
member(memb_no, name)
book(isbn, title, authors, publisher)
borrowed(memb_no, isbn, date)
```

Figure 3.20 Library database.

- ❑ **3.21.** Consider the library database of Figure 3.20. Write the following queries in SQL.
- ❑ a. Find the member number and name of each member who has borrowed at least one book published by “McGraw-Hill”.
 - ❑ b. Find the member number and name of each member who has borrowed every book published by “McGraw-Hill”.
 - ❑ c. For each publisher, find the member number and name of each member who has borrowed more than five books of that publisher.
 - ❑ d. Find the average number of books borrowed per member. Take into account that if a member does not borrow any books, then that member does not appear in the *borrowed* relation at all, but that member still counts in the average.



2-5 (10pt)

- ❑ **3.25** Using the university schema, write an SQL query to find the names of those departments whose budget is higher than that of Philosophy. List them in alphabetic order.



2-6 (10pt)

- ❑ **3.26** Using the university schema, use SQL to do the following: For each student who has retaken a course at least twice (i.e., the student has taken the course at least three times), show the course ID and the student's ID. Please display your results in order of course ID and do not display duplicate rows.



2-7 (10pt)

- ❑ **3.28** Using the university schema, write an SQL query to find the names and IDs of those instructors who teach every course taught in his or her department (i.e., every course that appears in the *course* relation with the instructor's department name). Order result by name.