Benny Lisangi

Abdullah Al Rahat, PhD

Database Systems

23 September 2022

Homework 2

1. Chapter 3 introduces the fundamentals of the SQL query language. The chapter discusses DDL and DML statements for the SQL query language. For the data definition, the basic SQL data types are discussed along with SQL commands for creating, inserting into, and altering relations. As for the DML statements, the chapter introduces how to query into a relation using the select statement along with the where, having, and as classes. It also discusses aggregate and set functions, grouping classes, and the cartesian product of two relations.
2. 1. SELECT title FROM course WHERE credits = 3;
   2. SELECT takes.ID FROM takes, teaches, (SELECT ID FROM instructor WHERE name = 'Einstein') AS einstein WHERE takes.course\_id = teaches.course\_id AND einstein.ID = teaches.ID
   3. SELECT MAX (salary) FROM instructor
   4. SELECT name FROM instructor WHERE salary in (SELECT MAX (salary) FROM instructor)
   5. SELECT COUNT(sec\_id) as enrollment, sec\_id FROM takes WHERE semester = 'Fall' AND year = '2017' GROUP BY sec\_id
   6. (SELECT MAX(F.enrollment)AS max\_enrol FROM (SELECT COUNT(sec\_id) as enrollment, sec\_id FROM takes WHERE semester = 'Fall' AND year = '2017' GROUP BY sec\_id) AS F) AS G
   7. SELECT sec\_id, count(sec\_id) as enrollment FROM section, (SELECT MAX(F.enrollment)AS max\_enrol FROM (SELECT COUNT(sec\_id) as enrollment, sec\_id FROM takes WHERE semester = 'Fall' AND year = '2017' GROUP BY sec\_id) AS F) AS G

WHERE section.semester = ‘Fall’ AND semester.year = ‘2017’ GROUP BY section HAVING (enrollment = G.max\_enrol)

1. 1. UPDATE instructor SET salary = salary \* 1.1 WHERE dept\_name = 'Comp. Sci.'
   2. DELETE FROM course WHERE course\_id NOT IN (SELECT course\_id FROM section)
   3. INSERT INTO instructor SELECT ID, name, dept\_name, 10000 FROM student WHERE tot\_cred > 100;
2. 1. INSERT INTO course(course\_id, title, dept\_name, credits) VALUES ('CS-001', 'Weekly Seminar', 'Comp. Sci.', 0)
   2. INSERT INTO section (course\_id, sec\_id, semester, year) VALUES ('CS-001', '1', 'Fall', 2017);
   3. INSERT INTO takes(ID, course\_id, sec\_id, semester, year) SELECT student.ID, section.course\_id, section.sec\_id, section.semester, section.year FROM section, student WHERE student.dept\_name = 'Comp. Sci.' AND section.course\_id = 'CS-001'
   4. DELETE FROM takes WHERE takes.sec\_id = '1' AND takes.ID = '12345'
   5. DELETE FROM course WHERE course\_id = 'CS-001', the section will automatically delete since we’re using on delete cascade command.
   6. DELETE FROM takes WHERE course\_id IN (SELECT course\_id FROM course WHERE title LIKE '%advanced%');

* CREATE TABLE person( driver\_id varchar(15), name varchar(25), address varchar(50), PRIMARY KEY (driver\_id));
* CREATE TABLE car (license\_plate VARCHAR(8), model VARCHAR(15),year NUMERIC(4,0), PRIMARY KEY (license\_plate));
* CREATE TABLE accident (report\_number VARCHAR(10), year NUMERIC(4,0),location VARCHAR(25), PRIMARY KEY (report\_number));
* CREATE TABLE owns (driver\_id VARCHAR(15), license\_plate VARCHAR(8), FOREIGN KEY (driver\_id) REFERENCES person ON DELETE SET NULL, FOREIGN KEY (license\_plate) REFERENCES car ON DELETE SET NULL, PRIMARY KEY(driver\_id, license\_plate));
* CREATE TABLE participated ( report\_number VARCHAR(10), license\_plate VARCHAR(8), driver\_id VARCHAR(15), damage\_amount NUMERIC(10,2) CHECK (damage\_amount >= 0), FOREIGN KEY (driver\_id) REFERENCES person ON DELETE SET NULL, FOREIGN KEY (license\_plate) REFERENCES car ON DELETE SET NULL, FOREIGN KEY (report\_number) REFERENCES accident ON DELETE SET NULL, PRIMARY KEY(report\_number, license\_plate));
* INSERT INTO person VALUES ('J-001', 'Julia Lisangi', '123 Main Str Township');
* INSERT INTO person VALUES ('123', 'Benny Lisangi', '123 Main Str Township');
* INSERT INTO PERSON (driver\_id, name) SELECT ID, name FROM instructor;
* INSERT INTO car (license\_plate, year) SELECT ID, 2017 FROM student LIMIT 5
  1. SELECT COUNT(driver\_id) FROM owns WHERE driver\_id IN (SELECT driver\_id FROM participated as p, accident as WHERE p.report\_number = a.report AND a.year = 2017)
  2. DELETE FROM cars WHERE year = 2010 AND license\_plate IN (SELECT license\_plate FROM owns WHERE ID = '12345');
  3. SELECT COUNT(license\_plate) FROM participated WHERE license\_plate IN (SELECT license\_plate FROM owns WHERE driver\_id IN (SELECT driver\_id FROM person WHERE name = 'John Smith'))
  4. UPDATE participated SET damage\_amount = 3000 WHERE license\_plate = 'AABB200' AND report\_number = 'AR2197';



* CREATE TABLE branch (branch\_name VARCHAR(20), branch\_city VARCHAR(20), assets NUMERIC(20,2), PRIMARY KEY(branch\_name));
* CREATE TABLE customer (ID VARCHAR(10), customer\_name VARCHAR(25), customer\_street VARCHAR(15), customer\_city VARCHAR(15), PRIMARY KEY(ID));
* CREATE TABLE loan (loan\_number VARCHAR(8), branch\_name VARCHAR(20),amount NUMERIC(10,2), FOREIGN KEY (branch\_name) REFERENCES(branch) ON DELETE SET NULL, PRIMARY KEY(loan\_number));
* CREATE TABLE borrower (ID VARCHAR (10), loan\_number NUMERIC(8,0), PRIMARY KEY(ID, loan\_number), FOREIGN KEY (ID) REFERENCES customer, FOREIGN KEY(loan\_number) REFERENCES loan);
* CREATE TABLE account (account\_number VARCHAR(8), PRIMARY KEY(account\_number), branch\_name VARCHAR(20), FOREIGN KEY (branch\_name) REFERENCES branch, balance NUMERIC(10,2));
* CREATE TABLE depositor(ID VARCHAR(10), FOREIGN KEY (ID) REFERENCES customer, account\_number NUMERIC(8,0), FOREIGN KEY (account\_number) REFERENCES account, PRIMARY KEY (ID, account\_number));
* INSERT INTO branch VALUES (‘uptown branch’, ‘Township’, 200000000);
* INSERT INTO customer VALUES (‘12346’,‘John Smith’, ‘Main Street’, ‘Township’)
* INSERT INTO loan VALUES (‘12355’, ‘Uptown branch’, 12344.00);
* INSERT INTO borrower VALUES (‘12346’,’12355’);
* INSERT INTO account VALUES (‘198455’, ‘Uptown branch’, 34566.86);
* INSERT INTO depositor VALUES (‘12346’, ‘198455’);
  1. SELECT ID FROM customer WHERE ID NOT IN (SELECT ID FROM loan);
  2. SELECT customer.ID FROM customer, (SELECT customer\_city, customer\_street FROM customer WHERE ID = ‘12345’) AS p WHERE customer.customer\_city = p.customer\_city AND customer.customer.street = p.customer\_city;
  3. SELECT branch\_name FROM account WHERE account\_number IN (SELECT account\_number FROM depositor, customer WHERE customer.customer\_city = 'Harrison' and customer.ID = depositor.ID)