Cheatography

122a-midterm2 Cheat Sheet

by lolitskevin via cheatography.com/61137/cs/15834/

selecting without having

```
SELECT d.did, d.budget,
avg(e.salary) FROM Emp e, Dept d,
Works w WHERE e.eid=w.did and
w.did=d.did and w.pcttime>=40 GROUP
BY d.did,d.budget;
```

selecting with joins

```
SELECT u.user_id, u.first_name,
u.last_name
FROM user u natural join class c
natural join takes t natural join
student s
WHERE c.dept = 'CS' and s.major =
'ART'
GROUP BY u.user_id
HAVING COUNT(u.user_id) >= 4;
```

selecting with fancy joins

```
SELECT u.user_id, u.email,
s.user_id as sid
FROM user u left join student s on
u.user_id = s.user_id WHERE
s.user_id is null;
```

union

```
SELECT first_name, last_name
FROM customer
UNION
SELECT first_name, last_name
FROM staff
ORDER BY 1, 2;
```

create/drop view

```
DROP VIEW CSStudentView;
CREATE VIEW CSStudentView(user_id,
first_name, last_name, class_no,
dept, cno, grade, title, level)
AS
SELECT s.user_id, u.first_name,
u.last_name, c.class_no, c.dept,
c.cno, t.grade, co.title, co.level
FROM student s natural join user u
natural join class c natural join
takes t natural join course co
WHERE s.user_id = u.user_id
GROUP BY user_id;
```

create/drop trigger

```
DROP TRIGGER update_popularity;
DELIMITER //
CREATE TRIGGER update_popularity
AFTER INSERT ON Likes FOR EACH ROW
BEGIN
    UPDATE Post
SET popularity = popularity + 1
WHERE Post.post_id = NEW.post_id;
END; //
DELIMITER ;
```

trigger with if

```
delimiter //
CREATE TRIGGER NoLowerAge BEFORE
UPDATE ON Emp FOR EACH ROW
BEGIN
IF NEW.age < OLD.age
THEN SET NEW.age = OLD.age;
END IF;
END;//
delimiter ;
```

relational algebra

π bid ((σ age = 35 ∧ rating >= 5 (Sailor)) Reserves) ∩ π bid ((σ rating < 5 (Sailor)) Reserves))

insert / delete

```
INSERT INTO products (productCode,
name, quantity, price) VALUES
('PEC', 'Pencil 2B', 10000, 0.48),
('PEC', 'Pencil 2H', 8000, 0.49);
DELETE FROM products WHERE price >
0.4;
```

available operators

AND, OR, NOT, XOR, IN, NOT IN, BETWEEN, NOT BETWEEN, IS NULL, IS NOT NULL, AS (ALIAS), ORDER BY .. ASC DESC, LIMIT
aggregate functions: COUNT, MAX, MIN, AVG, SUM, STD, GROUP_CONCAT