Adrian Grimm

#12 Case Study

1.

SELECT officer\_id "Officer ID", last || ', ' || first "Name"

FROM officers JOIN crime\_officers USING (officer\_id)

JOIN crimes USING (crime\_id)

WHERE crime\_id > ALL (SELECT AVG(COUNT(\*))

FROM crimes

GROUP BY officer\_id);

2.

SELECT criminal\_id "Criminal ID", first || ' ' || last "Name"

FROM criminals JOIN crimes USING (criminal\_id)

WHERE crime\_id < ALL (SELECT AVG(COUNT(\*))

FROM crimes

GROUP BY criminal\_id)

AND v\_status = 'N';

3.

SELECT \*, AVG(hearing\_date-filing\_date) "Days"

FROM appeals

WHERE 'Days' <ALL (SELECT (hearing\_date-filing\_date)

FROM appeals);

4.

SELECT p.prob\_id "ID", p.first || ' ' || p.last "Name"

FROM prob\_officers p JOIN sentences s

ON p.prob\_id = s.prob\_id

WHERE sentence\_id <ALL (SELECT AVG(COUNT(\*))

FROM sentences

GROUP BY prob\_id);

5.

SELECT c.crime\_id, a.hearing\_date

FROM crimes c JOIN appeals a

ON c.crime\_id = a.crime\_id

WHERE a.filing\_date >ALL (SELECT MAX(filing\_date)

FROM appeals);

6.

SELECT charge\_id, crime\_id, crime\_code, charge\_status

FROM crime\_charges

WHERE fine\_amount >ANY (SELECT AVG(fine\_amount)

FROM crime\_charges

WHERE amount\_paid <ALL (SELECT AVG(amount\_paid)

FROM crime\_charges));

7.

SELECT first || ' ' || last "Name"

FROM criminals JOIN crimes USING (criminal\_id)

WHERE crime\_id IN (SELECT crime\_id

FROM crimes JOIN crime\_charges USING (crime\_id)

WHERE crime\_code = 10089);

8.

SELECT last || ', ' || first "Criminal Name", criminal\_id "Criminal ID"

FROM criminals

WHERE EXISTS (SELECT criminal\_id

FROM sentences

WHERE prob\_id > 0);

9.

SELECT officer\_id "Officer ID", first || ' ' || last "Name"

FROM officers JOIN crime\_officers USING (officer\_id)

JOIN crimes USING (crime\_id)

WHERE crime\_id > ALL (SELECT AVG(COUNT(\*))

FROM crimes

GROUP BY officer\_id);

10.

MERGE INTO criminals\_dw a

USING criminals b

ON a.criminal\_id = b.criminal\_id

WHEN MATCHED THEN

UPDATE SET a.criminal\_id=b.criminal\_id,

a.last=b.last,

a.first=b.first,

a.street=b.street,

a.city=b.city,

a.state=b.state,

a.zip=b.zip,

a.phone=b.phone,

a.v\_status=b.v\_status,

a.p\_status=b.p\_status

WHEN NOT MATCHED THEN

INSERT INTO (criminal\_id, last, first, street, city, state, zip, phone, v\_status, p\_status)

Values (b.criminal\_id, b.last, b.first ,b.street, b.state,b.zip,.phone,b.v\_status, b.p\_status);