

3.1

a.

```
SELECT TITLE  
  
FROM COURSE  
  
WHERE DEPT_NAME = 'Comp.Sci.';
```

b.

```
SELECT STUDENT.NAME  
  
FROM STUDENT, INSTRUCTOR, ADVISOR  
  
WHERE STUDENT.ID = ADVISOR.S_ID  
  
AND ADVISOR.I_ID = INSTRUCTOR.ID  
  
AND INSTRUCTOR.NAME = 'Einstein';
```

c.

```
SELECT MAX(SALARY) OVER ()  
  
FROM INSTRUCTOR;
```

d.

```
SELECT INSTRUCTOR.NAME  
  
FROM INSTRUCTOR  
  
WHERE INSTRUCTOR.SALARY = (SELECT MAX(SALARY) OVER () FROM INSTRUCTOR);
```

e.

```
SELECT  
  
    COURSE_ID,  
  
    SEC_ID,
```

```

(
    SELECT
        COUNT(ID)
    FROM
        TAKES
    WHERE
        TAKES.YEAR = SECTION.YEAR
        AND TAKES.SEMESTER=SECTION.SEMESTER
        AND TAKES.COURSE_ID=SECTION.COURSE_ID
        AND TAKES.SEC_ID=SECTION.SEC_ID
    ) AS ENROLLMENT
FROM
    SECTION
WHERE
    SEMESTER='Fall'
    AND YEAR = 2017;

```

f.

```

SELECT
    MAX(ENROLLMENT)
FROM
    (
        SELECT
            COUNT(ID) AS ENROLLMENT
        FROM
            SECTION,

```

```

    TAKES

WHERE

    TAKES.YEAR = SECTION.YEAR

    AND TAKES.SEMESTER = SECTION.SEMESTER

    AND TAKES.COURSE-ID = SECTION.COURSE-ID

    AND TAKES.SEC_ID = SECTION.SEC_ID

    AND TAKES.SEMESTER = 'Fall'

    AND TAKES.YEAR = 2017

GROUP BY

    TAKES.COURSE -ID,

    TAKES.SEC-ID

);

```

g.

```

WITH SEC-ENROLLMENT AS (

    SELECT

        TAKES.COURSE _ID,

        TAKES.SEC_ID,

        COUNT(ID) AS ENROLLMENT

    FROM

        SECTION,

        TAKES

    WHERE

        TAKES.YEAR = SECTION.YEAR

        AND TAKES.SEMESTER = SECTION.SEMESTER

```

```

        AND TAKES.COURSE-ID = SECTION.COURSE-ID

        AND TAKES.SEC_ID = SECTION.SEC-ID

        AND TAKES.SEMESTER = 'Fall'

        AND TAKES.YEAR = 2017

    GROUP BY

        TAKES.COURSE-ID,

        TAKES.SEC_ID

) SELECT COURSE -ID, SEC-ID FROM SEC-ENROLLMENT WHERE ENROLLMENT = (

    SELECT

        MAX(ENROLLMENT)

    FROM

        SEC-ENROLLMENT

);

```

3.3

a.

```

UPDATE INSTRUCTOR

SET

    SALARY = SALARY * 1.1

WHERE

    DEPT_NAME = 'Comp.Sci.';

```

b.

```

DELETE FROM COURSE

WHERE

    COURSE_ID NOT IN (

        SELECT

```

```
COURSE_ID  
  
FROM  
  
SECTION  
  
>);
```

c.

```
INSERT INTO INSTRUCTOR  
  
SELECT  
  
ID,  
  
NAME,  
  
DEPT_NAME,  
  
10000  
  
FROM  
  
STUDENT  
  
WHERE  
  
TOT_CRED > 100;
```

3.5

a.

```
SELECT  
  
ID,  
  
CASE  
  
WHEN SCORE < 40 THEN  
  
'F'  
  
WHEN SCORE < 60 THEN  
  
'C'
```

```
        WHEN SCORE < 80 THEN

            'B'

        ELSE

            'A'

    END

FROM

    MARKS;
```

b.

```
WITHG RADES AS

SELECT

    ID,

    CASE

        WHEN SCORE < 40 THEN

            'F'

        WHEN SCORE < 60 THEN

            'C'

        WHEN SCORE < 80 THEN

            'B'

        ELSE

            'A'

    END AS GRADE

FROM

    MARKS

SELECT

    GRADE,
```

```
        COUNT(ID)

FROM

    GRADES

GROUP BY

    GRADE;
```

3.11

a.

```
SELECT

    NAME

FROM

    STUDENT

    NATURAL JOIN TAKES

    NATURAL JOIN COURSE

WHERE

    COURSE.DEPT='Comp.Sci.';
```

b.

```
SELECT

    ID,

    NAME

FROM

    STUDENT EXCEPT

    SELECT

        ID,

        NAME
```

FROM

STUDENT

NATURAL JOIN TAKES

WHERE

YEAR<2017;

c.

SELECT

DEPT,

MAX(SALARY)

FROM

INSTRUCTOR

GROUP BY

DEPT;

d.

SELECT

MIN(MAXSALARY)

FROM

(

SELECT

DEPT,

MAX(SALARY)AS MAXSALARY

FROM

INSTRUCTOR

GROUP BY

DEPT

);

3.13

person (*driver_id*, *name*, *address*)
car (*license*, *model*, *year*)
accident (*report_number*, *date*, *location*)
owns (*driver_id*, *license*)
participated (*report_number*, *license*, *driver_id*, *damage_amount*)

3.14

a.

SELECT

COUNT(*)

FROM

ACCIDENT

WHERE

EXISTS (

SELECT

*

FROM

PARTICIPATED,

OWNS,

PERSON

WHERE

OWNS.DRIVERID=PERSON.DRIVERID

AND PERSON.NAME='JohnSmith'

AND OWNS.LICENSE=PARTICIPATED.LICENSE

```
AND ACCIDENT.REPORT_NUMBER=PARTICIPATED.REPORT_NUMBER
```

```
);
```

b.

```
UPDATE PARTICIPATED
```

```
SET
```

```
DAMAGE_AMOUNT=3000
```

```
WHERE
```

```
REPORT_NUMBER='AR2197'
```

```
AND LICENSE='AABB2000';
```

3. 17

a.

```
UPDATE WORKS
```

```
SET
```

```
SALARY=SALARY *1.1
```

```
WHERE
```

```
COMPANY_NAME='First Bank Corporation';
```

b.

```
UPDATE WORKS
```

```
SET
```

```
SALARY=SALARY *1.1
```

```
WHERE
```

```
EMPLOYEE_NAME IN(
```

```
SELECT
```

```
MANAGER_NAME
```

```
FROM  
  
MANAGES  
  
)  
  
AND COMPANY_NAME='First Bank Corporation';
```

c.

```
DELETE FROM WORKS  
  
WHERE  
  
COMPANY_NAME='Small Bank Corporation';
```

3.22

```
where(  
  
(  
  
SELECT  
  
COUNT(TITLE)  
  
FROM  
  
COURSE  
  
)= (SELEC COUNT(DISTINCT TITLE) FROM COURSE));
```

3.24

```
SELECT  
  
S.ID,  
  
S.NAME  
  
FROM  
  
STUDENT AS S  
  
INNER JOIN ADVISOR AS A  
  
ON S.ID = A.S_ID
```

```
INNER JOIN INSTRUCTOR AS I

ON A.I_ID = I.ID

WHERE

S.DEPT_NAME = 'Accounting'

AND I.DEPT_NAME = 'Physics';
```

3.25

```
SELECT

DEPT_NAME

FROM

DEPARTMENT

WHERE

BUDGET > (

SELECT

BUDGET

FROM

DEPARTMENT

WHERE

DEPT_NAME = 'Philosophy'

)

ORDER BY

DEPT_NAME ASC;
```

3.31

```
SELECT

ID,
```

NAME

FROM

INSTRUCTOR AS I

WHERE

'A' NOT IN (

SELECT

TAKES.GRADE

FROM

TAKES

INNER JOIN TEACHES

ON (TAKES.COURSE_ID,

TAKES.SEC_ID,

TAKES.SEMESTER,

TAKES.YEAR) = (TEACHES.COURSE_ID,

TEACHES.SEC_ID,

TEACHES.SEMESTER,

TEACHES.YEAR)

WHERE

TEACHES.ID = I.ID

);