COMPLEX QUERIES

1. Retrieve the names of student with maximum number of visitors.

SELECT student\_name

FROM student

WHERE student\_id IN

(SELECT s\_id

FROM visitors

GROUP BY s\_id

ORDER BY COUNT(\*) DESC);

student\_name

--------------

raghu

(1 row)

1. Give a salary hike of 15% to employees who serviced for atleast 2 years.

UPDATE employees

SET emp\_salary = emp\_salary \*1.15

WHERE emp\_name IN (

SELECT emp\_name

FROM employees

WHERE emp\_id LIKE '2017%');

hostels=# \i advancedqueries.sql

UPDATE 6

1. Select the day which had maximum number of visitors.

SELECT visit\_date

FROM visitors

GROUP BY visit\_date

ORDER BY COUNT(\*) DESC

LIMIT 1;

visit\_date

------------

2019-03-17

(1 row)

1. Add an extra of 10,000 rupees to salary of an employee who is a mess incharge.

UPDATE employees

SET emp\_salary=emp\_salary + 10000

WHERE emp\_name IN(

SELECT emp\_name

FROM employees,mess

WHERE month=’january’ AND mess.mess\_incharge=employees.emp\_name);

hostels=# \i advancedqueries.sql

UPDATE 1

1. Select total number of students in a particular block.

SELECT COUNT(student\_name) AS total\_students

FROM STUDENT

WHERE STUDENT.room\_no LIKE '4%';

total\_students

----------------

3

(1 row)