CSL3050: DBMS

Assignment 03

By Shashank Shekhar Asthana (B21CS093)

Q1. Below are the displayed screenshots, starting from the creation of the table and followed by queries.

salesman_id	name	address_city	coverage_city	commission	date_of_employment	date_of_release
1	Shashank Asthana	Lucknow	Pune	0.10	2022-01-22	NULL
2	Shreshth Vatsal Sharma	Aligarh	Gurugram	0.12	2019-05-12	2023-03-11
3	Anurag Verma	Bengaluru	Chennai	0.11	2022-01-22	NULL
4	Manan Srivastava	Chennai	Bengaluru	0.09	2019-05-12	2023-03-11
5	G Mukund	Kolkata	Howrah	0.08	2022-01-22	NULL
6	Naman Gupta	Hyderabad	Sikanderabad	0.10	NULL	NULL
7	Mayank	Pune	Mumbai	0.12	NULL	NULL
8	Sharonya Jain	Ahmedabad	Gandhinagar	0.11	NULL	NULL
9	Pranav Pant	Jaipur	Ajmer	0.09	NULL	NULL

a. Display commission rates of Mumbai and Chennai.

Result Grid Filter Rows:

commission
0.11
0.12

b. Display salesman_id and name of all salespeople who work in the same city as their address.

Result Grid | Filter Ro

salesman_id	name
NULL	NULL

c. Display salesman_id and name of all salespeople who have different addresses and coverage cities.

salesman_id	name
1	Shashank Asthana
2	Shreshth Vatsal Sharma
3	Anurag Verma
4	Manan Srivastava
5	G Mukund
6	Naman Gupta
7	Mayank
8	Sharonya Jain
9	Pranav Pant
10	Samaksh Verma

d. Display the highest commission rate of all salespeople working in Mumbai.

Result Grid | Filter

highest_commission
0.12

e. Display the coverage_city with the highest commission rate.

Result Grid | Filter Rows:

coverage_city
Gurugram
Mumbai

f. Display the average commission rate for all cities.

	AVG(commission)
▶	0.100000

g. Find the coverage_city where the commission rate is the same for all salespeople.

	coverage_city
▶	Ajmer
	Bengaluru
	Chennai
	Gandhinagar
	Gurugram
	Howrah
	Kanpur
	Mumbai
	Pune
	Sikanderabad

h. Find the commission rate that is common across all coverage cities.

Result Grid | Filter Rows:

	commission

i. Display all details of the salesperson who has worked for all cities.

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Cont

salesman_id	name	address_city	coverage_city	commission	date_of_employment	date_of_release
NULL	NULL	NULL	NULL	NULL	NULL	NULL

j. Add two new columns, 'date_of_employment' and 'date_of_release,' to the table to record the date of employment for all employees and the date of release if the salesperson has left the company.

date_of_employment	date_of_release
2022-01-22	NULL
2019-05-12	2023-03-11
2022-01-22	NULL
2019-05-12	2023-03-11
2022-01-22	NULL
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL

Q2. Screenshots of the tables created are shown below, followed by queries.

Classroom table,

Result Grid | Filter Rows:

building	room_number	capacity
Main Building	101	50
Main Building	102	60
Science Block	201	40
NULL	NULL	NULL

Department table

Result Grid | Filter Rows:

dept_name	building	budget
CSE	Main Building	1000000.00
Mathematics	Science Block	800000.00
Philosophy	Science Block	750000.00
NULL	NULL	NULL

Instructor table

Result Grid | Filter Rows: | Edit:

ID	instructor_name	dept_name	salary
101	Dr. Rajesh Kumar	CSE	85000.00
102	Prof. Priya Sharma	Mathematics	75000.00
103	Dr. Suresh Gupta	Philosophy	78000.00
NULL	NULL	NULL	NULL

Course table,

Result Grid | Filter Rows: | Edit:

course_id	title	dept_name	credit
1	Introduction to Programming	CSE	4
2	Calculus I	Mathematics	3
3	Literature	Philosophy	4
NULL	NULL	NULL	NULL

Section table

Result Grid | Filter Rows: | Edit: | Export/Import:

course_id	sec_id	semester	year	building	room_number	time_slot_id
1	1	Fall	2023	Main Building	101	1
2	1	Spring	2023	Main Building	102	2
3	1	Fall	2023	Science Block	201	3
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Teaches table

Result Grid | Filter Rows: | Ed

ID	course_id	sec_id	semester	year
101	1	1	Fall	2023
102	2	1	Spring	2023
103	3	1	Fall	2023
NULL	NULL	NULL	NULL	NULL

Student table

Result Grid | Filter Rows: | Edit

ID	name	dept_name	total_credit
202	Priya Sharma	Mathematics	45
203	Rahul Verma	Philosophy	55
12345	Amit Patel	CSE	50
NULL	NULL	NULL	NULL

Takes table

Result Grid | Filter Rows: | Edit:

ID	course_id	sec_id	semester	year	grade
202	2	1	Spring	2023	B+
203	3	1	Fall	2023	A-
12345	1	1	Fall	2023	A+
NULL	NULL	NULL	NULL	NULL	NULL

Advisor table

Result Grid

s_ID	i_ID
12345	101
202	102
203	103
NULL	NULL

Time slot table,

Result Grid Filter Rows: Edit:

time_slot_id	day	start_time	end_time
1	Monday	09:00:00	10:30:00
2	Tuesday	10:00:00	11:30:00
3	Wednesday	13:30:00	15:00:00
NULL	NULL	NULL	NULL

Gradepoints table

Result Grid

grade	points
A	9.00
A-	8.50
A+	10.00
B	7.00
B-	6.50
B+	8.00
C	6.00
F	0.00
NULL	NULL

Prereq table

Result Grid | Filter Row

course_id	prereq_id
2	1
3	2
NULL	NULL

Now writing the queries as displayed below

a. Find the total grade points earned by the student with ID '12345', across all courses taken by the student.

Result Grid | Filter Rows:

ID	name	total_grade_points
12345	Amit Patel	40.00

b. Find the grade point average (GPA) for the above student, that is, the total grade points divided by the total credits for the associated courses.

Result Grid | Filter Rows:

ID	name	GPA
12345	Amit Patel	10.000000

c. Find the ID and the grade-point average of each student for the CSE department.

Result Grid | Filter Rows

ID	GPA
12345	10.000000

d. Find the ID, Name, and GPA for the topper of each department in the University.

Result Grid | Filter Rows: | Export:

ID	name	dept_name	GPA
202	Priya Sharma	Mathematics	8.000000
203	Rahul Verma	Philosophy	8.500000
12345	Amit Patel	CSE	10.000000

e. Find the names of those departments whose budget is higher than that of Philosophy. List them in alphabetic order.

Result Grid | Filter

dept_name
CSE
Mathematics
NULL

f. Find the IDs of those students who have retaken at least three distinct courses at least once.

Result Grid

ID

g. Find the names and IDs of those instructors who teach every course taught in his or her department (i.e., every course that appears in the course relation with the instructor's department name). Order results by name.

Result Grid Filter Rows:

ID	instructor_name
101	Dr. Rajesh Kumar
103	Dr. Suresh Gupta
102	Prof. Priya Sharma
NULL	NULL

h. Find the names and IDs of those instructors along with the details of the courses they taught in Fall 2018, such that the total number of students getting A+ is more than 3 in those courses.

Result Grid Filter Rows: Export:

ID	instructor_name	course_id	title	credit

i. Find the ID and name of each instructor who has never given an A grade in any course she or he has taught.

Result Grid | Filter Rows:

ID	instructor_name
101	Dr. Rajesh Kumar
102	Prof. Priya Sharma
103	Dr. Suresh Gupta
NULL	NULL

j. Define the view student grades (ID, GPA), giving the grade-point average of each student. Make sure your view definition correctly handles the case of null values for the grade attribute of the takes relation.

```
CREATE VIEW student_grades AS
SELECT s.ID, s.name, COALESCE(SUM(gp.points * c.credit) / NULLIF(SUM(c.credit), 0), 0) AS GPA
FROM student s
LEFT JOIN takes t ON s.ID = t.ID
LEFT JOIN course c ON t.course_id = c.course_id
LEFT JOIN grade_points gp ON t.grade = gp.grade
GROUP BY s.ID, s.name;
```

Q3. We expect the constraint “an instructor cannot teach sections in two different

classrooms in a semester in the same time slot” to hold.

● Write an SQL query that returns all (instructor, section) combinations that violate this constraint.

● Write an SQL assertion to enforce this constraint.

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

instructor_1	course_id	sec_id	semester	year	building_1	room_number_1	time_slot_id_1	instructor_2	building_2	room_number_2	time_slot

None of the instructors matched the conditions given in the question

*** The queries that have not returned any value do not have any errors in the code , while the values that were entered in the table while populating the table were not likely to be queried according to the condition for that particular query.**
