Experiment 1: Accessing the database

Use SQL queries to execute the following commands:

1. Create the following Relation (Tables) with primary key integrity constraint

i.instructor

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

2 Create the following Relation (Tables) *teaches*

ID	Course_id	sec id	semester	year
10101	CS-101	1	Fall	2017
10101	CS-315	1	Spring	2018
10101	CS-347	1	Fall	2017
12121	FIN-201	1	Spring	2018
15151	MU-199	1	Spring	2018
22222	PHY-101	1	Fall	2017
32343	HIS-351	1	Spring	2018
45565	CS-101	1	Spring	2018
45565	CS-319	1	Spring	2018
76766	BIO-101	1	Summer	2017
76766	BIO-301	1	Summer	2018
83821	CS-190	1	Spring	2017
83821	CS-190	2	Spring	2017
83821	CS-319	2	Spring	2018
98345	EE-181	1	Spring	2017

3. Insert following additional tuple in *instructor* ('10211', 'Smith', 'Biology', 66000)
4. Delete this tuple from *instructor* ('10211', 'Smith', 'Biology', 66000)
5. Select tuples from *instructor* where dept_name = 'History'
6. Find the Cartesian product *instructor* x *teaches*.
7. Find the names of all instructors who have taught some course and the course_id
8. Find the names of all instructors whose name includes the substring "dar".
9. Find the names of all instructors with salary between 90,000 and 100,000 (that is, ≥ 90,000 and ≤ 100,000)

Experiment 2: Basic SQL

1. Order the tuples in the instructors relation as per their salary.
2. Find courses that ran in Fall 2017 or in Spring 2018
3. Find courses that ran in Fall 2017 and in Spring 2018
4. Find courses that ran in Fall 2017 but not in Spring 2018
5. Insert following additional tuples in *instructor* :('10211', 'Smith', 'Biology', 66000), ('10212', 'Tom', 'Biology', NULL)
6. Find all instructors whose salary is null.
7. Find the average salary of instructors in the Computer Science department.

Experiment 3: Intermediate SQL

1. Find the total number of instructors who teach a course in the Spring 2018 semester.
2. Find the number of tuples in the teaches relation
3. Find the average salary of instructors in each department
4. Find the names and average salaries of all departments whose average salary is greater than 42000
5. Name all instructors whose name is neither “Mozart” nor Einstein”.
6. Find names of instructors with salary greater than that of some (at least one) instructor in the Biology department.
7. Find the names of all instructors whose salary is greater than the salary of all instructors in the Biology department.
8. Find the average instructors’ salaries of those departments where the average salary is greater than 42,000.

Experiment 4: Intermediate, and advanced SQL

1. Find all departments where the total salary is greater than the average of the total salary at all departments
2. List the names of instructors along with the course ID of the courses that they taught.
3. List the names of instructors along with the course ID of the courses that they taught. In case, an instructor teaches no courses keep the course ID as null.
4. Create a view of instructors without their salary called faculty
5. Give select privileges on the view faculty to the new user.

Experiment 5: Advanced SQL

Perform the following actions on a database through SQL queries.

1. Create a view of instructors without their salary called faculty
2. Create a view of department salary totals
3. Create a role of student
4. Give select privileges on the view faculty to the role student.
5. Create a new user and assign her the role of student.
6. Login as this new user and find all instructors in the Biology department.
7. Revoke privileges of the new user
8. Remove the role of student.
9. Give select privileges on the view faculty to the new user.
10. Login as this new user and find all instructors in the finance department.
11. Login again as root user
12. Create table teaches2 with same columns as teaches but with additional constraint that that semester is one of fall, winter, spring or summer.
13. Create index ID column of teaches. Compare the difference in time to obtain query results with or without index.
14. Drop the index to free up the space.

Design a database with several users like university database, railways ticket database. Envisage the use of the database, the requirements of the management, end users etc. Fill the database with dummy entries. Assign different privileges to different users. Login as different users and test their privileges.

Experiment 6: Accessing the database through Python

Install mysql_connector on anaconda. Then run the following commands on a database through python.

1. Create the following Relation (Tables) with primary key integrity constraint

i.instructor

ID	name	dept_name	salary
10101	Srinivasan	Comp. Sci.	65000
12121	Wu	Finance	90000
15151	Mozart	Music	40000
22222	Einstein	Physics	95000
32343	El Said	History	60000
33456	Gold	Physics	87000
45565	Katz	Comp. Sci.	75000
58583	Califieri	History	62000
76543	Singh	Finance	80000
76766	Crick	Biology	72000
83821	Brandt	Comp. Sci.	92000
98345	Kim	Elec. Eng.	80000

2 Create the following Relation (Tables) *teaches*

ID	Course_id	sec id	semester	year
10101	CS-101	1	Fall	2017
10101	CS-315	1	Spring	2018
10101	CS-347	1	Fall	2017
12121	FIN-201	1	Spring	2018
15151	MU-199	1	Spring	2018
22222	PHY-101	1	Fall	2017
32343	HIS-351	1	Spring	2018
45565	CS-101	1	Spring	2018
45565	CS-319	1	Spring	2018
76766	BIO-101	1	Summer	2017
76766	BIO-301	1	Summer	2018
83821	CS-190	1	Spring	2017
83821	CS-190	2	Spring	2017
83821	CS-319	2	Spring	2018
98345	EE-181	1	Spring	2017

1. Insert following additional tuple in *instructor* : ('10211', 'Smith', 'Biology', 66000)
2. Delete this tuple from *instructor* : ('10211', 'Smith', 'Biology', 66000)
3. Select tuples from *instructor* where dept_name = 'History'
4. Find the Cartesian product *instructor* x *teaches*.
5. Find the names of all instructors who have taught some course and the course_id
6. Find the names of all instructors whose name includes the substring "dar".
7. Find the names of all instructors with salary between 90,000 and 100,000 (that is, ≥ 90,000 and ≤ 100,000)

Experiment 7: Accessing the database through Python: Advanced techniques

Run the following commands on a database through python.

1. Order the tuples in the instructors relation as per their salary.
2. Find courses that ran in Fall 2017 or in Spring 2018
3. Find courses that ran in Fall 2017 and in Spring 2018
4. Find courses that ran in Fall 2017 but not in Spring 2018
5. Insert following additional tuples in *instructor*
   ('10211', 'Smith', 'Biology', 66000)
   ('10212', 'Tom', 'Biology', NULL
6. Find all instructors whose salary is null.
7. Find the average salary of instructors in the Computer Science department.
8. Find the total number of instructors who teach a course in the Spring 2018 semester.
9. Find the number of tuples in the teaches relation
10. Find the average salary of instructors in each department
11. Find the names and average salaries of all departments whose average salary is greater than 42000
12. Name all instructors whose name is neither "Mozart" nor Einstein".
13. Find names of instructors with salary greater than that of some (at least one) instructor in the Biology department.
14. Find the names of all instructors whose salary is greater than the salary of all instructors in the Biology department.
15. Find the average instructors' salaries of those departments where the average salary is greater than 42,000.
16. Find all departments where the total salary is greater than the average of the total salary at all departments
17. List the names of instructors along with the course ID of the courses that they taught.
18. List the names of instructors along with the course ID of the courses that they taught. In case, an instructor teaches no courses keep the course ID as null.

Measure the time difference it takes to execute a query on python and SQL.

Experiment 8: Introduction to Git

Create an account on the github website, and install git desktop.

create a git repository at your local system

Synchronize the git repository on your local system with the git webpage.

Commit the work at regular intervals

Revert to an earlier commit and start the coding from there again.

Write a script which runs the following commands on a database through python while using version control through github.

1. Create a view of instructors without their salary called faculty
2. Create a view of department salary totals
3. Create a role of student
4. Give select privileges on the view faculty to the role student.
5. Create a new user and assign her the role of student.
6. Revoke privileges of the new user
7. Remove the role of student.
8. Give select privileges on the view faculty to the new user.
9. Create table teaches2 with same columns as teaches but with additional constraint that that semester is one of fall, winter, spring or summer.
10. Create index ID column of teaches. Compare the difference in time to obtain query results with or without index.
11. Drop the index to free up the space.

Experiment 9: Create an app connected to a database

Write a program in python which performs the following tasks on MySQL. Also ensure that you use github for version control. Specifically use the commands commit, branching etc to manage your work.

1. Given input as salary of three months (jan, feb, march), write a program that returns the total salary for quarter 1 (Q1).
2. Commit your work on github
3. Create table 'instructor' with the columns as ID, name, dept_name, salary, age. Add the following tuples to the table.

   (10101,'Srinivasan','Comp. Sci.',65000, 34),

   (12121,'Wu','Finance',90000, 38),

   (15151,'Mozart','Music',40000, 45),

   (22222,'Einstein','Physics',95000, 55)

4. Create a function that helps distinguish whether an instructor is eligible for sabbatical or not. An instructor can avail sabbatical only if his/her age is greater than 40. So, write a program that takes as input an integer (representing age) and returns affirmatively if the integer is greater than 40.
5. Commit your work on github.
6. Find all instructors who can avail sabbatical from the database.
7. Commit your work on github.

Experiment 10: Term project

Create a movie ticket booking system. The users should be able to choose the movie, theater, screen and city of their choice to watch the movie. Then input these values into a connected database. Additionally use git for version control.

You may work upon other project ideas.