

MIGIDE ALLGER SYMLES

169202, ICS 3.1D

Advanced SQL Presentation.

7. Insert appropriate tuple into each of the views faculty and CSinstructors, to see what updates your database allows on views; explain what happens.

```
1 • DELETE FROM faculty
2 WHERE ID IN ('99999', '88888');
3 • INSERT INTO faculty (ID, name, dept_name)
4 VALUES ('99999', 'Test Instructor', 'Physics');
5 • INSERT INTO CSinstructors (ID, name, dept_name, salary)
6 VALUES ('88888', 'CS Instructor', 'Comp. Sci.', 60000);
```

Result

✓	243	20:58:02	DELETE FROM faculty WHERE ID IN ('99999', '88888')	2 row(s) affected	0.032 sec
✓	244	20:58:03	INSERT INTO faculty (ID, name, dept_name) VALUES ('99999', 'Test Instructor', 'Physics')	1 row(s) affected	0.015 sec
✓	245	20:58:03	INSERT INTO CSinstructors (ID, name, dept_name, salary) VALUES ('88888', 'CS Instructor', 'Comp. Sci.', 60000)	1 row(s) affected	0.016 sec

8. Grant permission to one of your friends to view all data in your student relation.

```
8 • CREATE USER 'hope'@'localhost' IDENTIFIED BY 'hope';
9 • SELECT user, host FROM mysql.user WHERE user = 'hope';
10
11 • GRANT SELECT ON university1.student TO 'hope'@'localhost';
12 • SELECT * FROM university1.student;
13
```

Result

✓	247	21:04:04	CREATE USER 'hope'@'localhost' IDENTIFIED BY 'hope'	0 row(s) affected	0.031 sec
✓	248	21:04:04	SELECT user, host FROM mysql.user WHERE user = 'hope' LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
✓	249	21:04:04	GRANT SELECT ON university1.student TO 'hope'@'localhost'	0 row(s) affected	0.000 sec
✓	250	21:04:04	SELECT * FROM university1.student LIMIT 0, 1000	13 row(s) returned	0.016 sec / 0.000 sec

9. Now grant permission to all users to see all data in your faculty view. Conversely, find a friend who has granted you permission on their faculty view, and execute a select query on that view.

```
14 • GRANT SELECT ON university1.faculty TO PUBLIC;
15
16
```

Result

✗	264	21:09:51	GRANT SELECT ON faculty TO PUBLIC	Error Code: 1410. You are not allowed to create a user with GRANT	0.000 sec
---	-----	----------	-----------------------------------	---	-----------

I wasn't able to test the permission-related queries because my MySQL user lacks the GRANT OPTION privilege.

Triggers

STEP 15. Confirm that the updated scheduled event is being fired every 2 minutes Execute the following statement to view the data in the ticketing relation: SELECT * FROM `classicmodels`.`customer_service_ticket`;

```
SELECT * FROM `classicmodels`.`customer_service_ticket`;
```

Result

	customer_service_ticket_ID	customer_service_ticket_resolved	customer_service_ticket_raise_time	customer_service_total_wait_time_minutes
▶	1	0	2025-06-30 10:46:10	63
	2	0	2025-06-30 10:47:45	61
	3	0	2025-06-30 10:47:45	61
	4	0	2025-06-30 10:49:56	59
	5	0	2025-06-30 10:49:56	59
	6	0	2025-06-30 10:49:56	59
	7	0	2025-06-30 10:50:09	59

customer_service_total_wait_time_minutes	customer_service_ticket_last_update	customerNumber
63	The last 2-minute recurring update was made at...	145
61	The last 2-minute recurring update was made at...	145
61	The last 2-minute recurring update was made at...	496
59	The last 2-minute recurring update was made at...	145
59	The last 2-minute recurring update was made at...	496
59	The last 2-minute recurring update was made at...	168
59	The last 2-minute recurring update was made at...	145