

Database Systems Project -- Riley Phan

Query results:

Level 1

1.

```
1  -- LEVEL 1
2  -- 1. One query must be a basic SELECT/FROM/WHERE query: select all part-time employees who are wanting to work night shift
3  -- (simple query, it works)
4  • SELECT * FROM employees
5  WHERE shift_preference = "night" AND employment_type = "part time";
```

employeeID	firstName	lastName	shift_preference	employment_type	hourly_rate	date_of_birth	email	phone_number
1	Rikki	Wims	night	part time	15.37	1989-01-17 00:00:00	rwims0@dell.com	574-209-1201
3	Vicki	Roche	night	part time	19.53	1989-08-26 00:00:00	vroche2@hostgator.com	511-962-9310
4	Winnifred	Pendleton	night	part time	15.56	1999-08-30 00:00:00	wpendleton3@51.la	596-870-4710
6	Hendrik	Sammes	night	part time	15.26	1993-05-29 00:00:00	hsammes5@java.com	643-927-0712
7	Marylee	Larret	night	part time	39.76	1999-01-22 00:00:00	mlarret6@bbb.org	633-437-6354
8	Pincus	Neasam	night	part time	24.84	1990-10-30 00:00:00	pneasam7@youku.com	476-913-0941
9	Griffin	Iannuzzi	night	part time	22.47	1996-06-03 00:00:00	giannuzzi8@creativecommons.org	649-591-1065
16	Steve	Abbes	night	part time	20.55	1993-05-01 00:00:00	sabbesf@walmart.com	449-243-8258
30	Emmett	Boake	night	part time	28.91	2002-06-14 00:00:00	eboaket@webmd.com	490-319-3510
34	Wes	Powley	night	part time	34.04	1996-05-31 00:00:00	wpowleyx@globo.com	462-394-0141
40	Owen	MacGrath	night	part time	16.28	1999-12-27 00:00:00	omacgrath13@marriott.com	523-369-5583
41	Brian	McGuirk	night	part time	32.19	1986-03-29 00:00:00	bmcguirk14@howstuffworks.com	272-500-6049
43	Ida	Balstone	night	part time	17.04	1989-03-23 00:00:00	ibalstone16@cafepress.com	294-626-8049
44	Anthe	Else	night	part time	16.3	1998-05-17 00:00:00	aelse17@behance.net	502-582-3688
45	Kathi	Healey	night	part time	24.81	1995-08-23 00:00:00	khealey18@narod.ru	735-323-1119
47	Harriette	Welland	night	part time	32.64	1994-04-09 00:00:00	hwelland1a@homestead.com	614-286-8110
48	Rosaleen	Francomb	night	part time	16.2	1994-07-09 00:00:00	rfrancomb1b@archive.org	187-516-9406
NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

2.

```
5  -- 2. One query must answer a question that needs to join 2 or more tables: find the inventory that need restocking
6  -- (remaining quantity less than stock threshold) and the suppliers of those inventory
7  • SELECT i.inventoryName, "supplies needed", s.name, s.contact FROM suppliers s
8  INNER JOIN inventory i
9  ON s.product_provided = i.inventoryName
10 WHERE i.reorderThreshold < i.remaining_quantity;
11
```

inventoryName	supplies needed	name	contact
butter	supplies needed	Grant, Stehr and O'Kon	903-924-5646
butter	supplies needed	Gorczy and Sons	806-637-1339
butter	supplies needed	Will LLC	539-628-9753
chocolate	supplies needed	Crist Group	200-844-8197
chocolate	supplies needed	Bashirian, Kutch and Muller	493-604-9443
cinnamon	supplies needed	Huel Inc	779-966-0593
cleaning supplies	supplies needed	Hintz and Sons	701-145-3629
coffee beans	supplies needed	Borer-Ryan	173-648-9546
cups	supplies needed	Sipes, Quitzon and Veum	998-835-4825
eggs	supplies needed	Lockman-VonRueden	499-920-0786
eggs	supplies needed	Huels-Swaniawski	748-134-9764
eggs	supplies needed	Labadie, Harvey and Emm...	789-816-6400
eggs	supplies needed	O'Connell, Harber and Ern...	630-948-8494
flour	supplies needed	Kuhn, Wolf and Schoen	353-676-8551
flour	supplies needed	Glover-Roob	354-260-0111
flour	supplies needed	Ziemann Group	989-498-1546
fruit	supplies needed	Yost, Ullrich and Ruecker	970-452-6172
milk	supplies needed	Orn-Beahan	657-810-9688
milk	supplies needed	Collier-Parker	923-393-2996
nuts	supplies needed	Doyle, Reynolds and Kshl...	205-575-1392
nuts	supplies needed	Bartell-Collier	341-202-1586
tea	supplies needed	Jones Group	704-923-4768
tea	supplies needed	Bernaum, Littell and Morar	689-765-9316

3.

```

14 -- 3. One query must have a subquery: find id and name of employees who have worked the most hours by role
15 -- (cashier, baker, barista, shift manager)
16 • SELECT s1.employeeID, CONCAT(e.firstName, " ", e.lastName) AS eName, s1.role, s1.total_hours
17 FROM schedule s1
18 INNER JOIN employees e
19 ON s1.employeeID = e.employeeID
20 WHERE s1.total_hours = (
21     SELECT MAX(s2.total_hours)
22     FROM schedule s2
23     WHERE s1.role = s2.role
24 );
25

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content: I A	Result Grid	Form Editor
employeeID	eName	role	total_hours		
8	Pincus Neasam	barista	7.27		
22	Saloma Woodyear	shift manager	7.94		
23	Hillery Pheasant	cashier	7.63		
17	Eugenia Merigeau	baker	7.49		

4.

```

16 -- 4. One query must have an aggregate function and GROUP BY clause: find the product name of the top 10 most ordered products
17 -- as well as the number of times they were ordered in descending order (works)
18 • SELECT p.productName, COUNT(o.productID) AS order_count
19 FROM orders o
20 LEFT JOIN products p
21 ON o.productID = p.productID
22 GROUP BY p.productName
23 ORDER BY order_count DESC
24 LIMIT 10;
25

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content: I A	Fetch rows:	Result Grid	Form Editor	Field Types	Query Stats
productName	order_count							
muffin	16							
bagel	14							
avocado toast	14							
croissant	12							
iced coffee	12							
matcha latte	11							
biscotti	11							
tea	10							
cappuccino	10							
americano	10							

Level 2

1.

```
36 -- LEVEL 2
37 -- 1. One query must add a record to one or more tables.
38 • INSERT INTO employees (firstName, lastName, shift_preference, employment_type, hourly_rate, date_of_birth, email, phone_number)
39 VALUES ("Steve", "Rogers", "night", "full time", 20, "1918-07-04", "srogers@avengers.com", "012-345-789");
40 -- Query the newly added row
41 • SELECT * FROM employees WHERE phone_number = "012-345-789";
42
```

employeeID	firstName	lastName	shift_preference	employment_type	hourly_rate	date_of_birth	email	phone_number
51	Steve	Rogers	night	full time	20	1918-07-04 00:00:00	srogers@avengers.com	012-345-789

2.

```
43 -- 2. One query must delete a record from one or more tables.
44 -- Query before modifying
45 • SELECT * FROM schedule WHERE employeeID = 22;
46
```

scheduleID	date	employeeID	role	shift	total_hours	start_time	end_time	overtime_hours
6	2024-06-20 00:00:00	22	shift manager	night	7.94	5:00 PM	1:30 PM	5.14
12	2024-04-11 00:00:00	22	shift manager	day	4.63	5:38 AM	8:42 PM	5.55

```
43 -- 2. One query must delete a record from one or more tables.
44 -- Query before modifying
45 • SELECT * FROM schedule WHERE employeeID = 22;
46 -- Delete employee record
47 • DELETE FROM schedule
48 WHERE employeeID = 22;
49 -- Requery the record from schedule table
50 • SELECT * FROM schedule WHERE employeeID = 22;
51
```

scheduleID	date	employeeID	role	shift	total_hours	start_time	end_time	overtime_hours

schedule 20 x Apply Revert

Output

Action Output

#	Time	Action	Message
263	15:56:40	Apply changes to schedule	No changes detected
264	15:58:56	SELECT * FROM schedule WHERE employeeID = 22 LIMIT 0, 1000	2 row(s) returned
265	15:59:06	DELETE FROM schedule WHERE employeeID = 22	2 row(s) affected
266	15:59:10	SELECT * FROM schedule WHERE employeeID = 22 LIMIT 0, 1000	0 row(s) returned

3.

```

52 -- 3. You have foreign key ON UPDATE constraints that make sense on a table/tables
53 -- and write an SQL query to demonstrate how one of them works
54 -- I have constraint ON UPDATE NO ACTION for foreign key product_provided in suppliers tables, since I want to update my
55 -- inventory record without losing the contacts info of my supplier
56 -- Updating reorderThreshold for "chocolate" from 100 to 200 will work (no error) but doesn't result in any difference
57 -- in the suppliers table)
58 -- (Updating, for example, record of "chocolate" to "chocolate chips" won't work since
59 -- that will violates referential integrity)
60 • UPDATE inventory
61   SET reorderThreshold = 200
62   WHERE inventoryName = "Chocolate";
63
64 -- Select suppliers for chocolate
65 • SELECT * FROM suppliers WHERE product_provided = "Chocolate";
66

```

Result Grid				
	id	name	contact	product_provided
▶	14	Crist Group	200-844-8197	chocolate
	17	Bashirian, Kutch and Muller	493-604-9443	chocolate
•	NULL	NULL	NULL	NULL

suppliers 15 x

Apply Revert

Output

Action Output

#	Time	Action	Message
✓ 243	15:43:11	UPDATE inventory SET reorderThreshold = 200 WHERE inventoryName = "Chocolate"	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0

4.

```

55 -- 4. You have foreign key ON DELETE constraints that make sense on a table/tables
56 -- and write an SQL query to demonstrate how one of them works.
57 -- I have constraint ON DELETE RESTRICT for the foreign key customerID in feedback table, since I want to retain info
58 -- of all customers with a loyalty customerID, as well as their feedbacks
59 -- Delete customer records that match the first 2 row in feedback table, which returns an error
60 • DELETE FROM customers
61   WHERE customerID = (
62     SELECT customerID FROM feedback
63     LIMIT 1
64   );

```

Context Help Snippets

Output

Action Output

#	Time	Action	Message	Duration / Fetch
✓ 238	15:33:33	SELECT customerID FROM feedback LIMIT 1	1 row(s) returned	0.000 sec / 0.000 sec
✗ 239	15:33:39	DELETE FROM customers WHERE customerID = (SELECT customerID FROM feedback LIMIT 1	Error Code: 1451. Cannot delete or update a parent row: a foreign key constraint fails ('cafe_db_1file', 'f...	0.047 sec