BRAINSTER

Workshop 11: SQL SELECT Statement

Brainster Web Development Academy

01. Create table `titles` and populate it with the data as in the screenshot on the right.

id	name
1	Manager
2	Executive
3	Assistant Manager
4	Team Lead
5	Coder
6	Agent

02. Create table `workers` and populate it.

Note: The title_id column is a foreign key referencing the `titles` table.

id	firstname	lastname	salary	department	title_id	join_date
1	Hanna	Chapman	1000	HR	1	2019-01-01
2	Izabel	Tang	950	HR	3	2019-01-05
3	Erica	Porter	1200	Accounts	2	2019-01-01
4	Shelbie	Noble	1500	IT	5	2019-02-01
5	Scarletta	Correa	2000	Marketing	6	2019-02-05
6	Juan	Barron	800	R&D	2	2019-03-01
7	Joy	Castro	500	Production	6	2019-03-12
8	Izabel	Tag	1400	R&D	2	2019-03-18
9	Anika	Rabi	2110	Marketing	6	2019-03-28
10	Kristof	White	1850	HR	6	2019-04-04
11	Tom	Scott	1000	Production	6	2019-05-05
12	Mick	Daley	1450	Production	6	2019-05-10
13	Elis	Smith	980	IT	5	2019-06-01
14	Tonya	Wright	700	Marketing	6	2019-06-06
15	Pukki	Chase	2000	Marketing	4	2019-06-06

03. Fetch the firstname from the workers table using the alias `name` for the column.

name
Hanna
Izabel
Erica
Shelbie
Scarletta
Juan
Joy
Izabel
Anika
Kristof
Tom
Mick
Elis
Tonya
Pukki

04. Fetch the firstname from the workers table using the alias `name` for the column, but return the name with uppercase letters only.

Hint: use the UPPER or UCASE functions.

name
HANNA
IZABEL
ERICA
SHELBIE
SCARLETTA
JUAN
JOY
IZABEL
ANIKA
KRISTOF
TOM
MICK
ELIS
TONYA
PUKKI

05. Fetch the unique department names from the workers table.

Hint: use the SELECT DISTINCT statement.

department
HR
Accounts
IT
Marketing
R&D
Production

06. Fetch the first three letters from each worker's name.

Hint: use SUBSTR or SUBSTRING functions.

name
Han
Iza
Eri
She
Sca
Jua
Joy
Iza
Ani
Kri
Tom
Mic
Eli
Ton
Puk

07. Write a query that will print only the unique lengths of department names.

Example: if departments are `HR`, `IT` and `Marketing` the query will return following values: 2, 4. It should first calculate the length of each department name (2, 2, 4) and then return the unique lengths only.

unique_lengths
2
8
9
3
10

08. Fetch all workers while replacing all lowercase letters `a` with the uppercase letter `A`.

Hint: use the REPLACE function.

name
HAnnA
IzAbel
EricA
Shelbie
ScArlettA
JuAn
Joy
IzAbel
AnikA
Kristof
Tom
Mick
Elis
TonyA
Pukki

09. Fetch the worker's firstname and lastname as a single column named full_name.

Note: in the full_name column the firstname and lastname should be separated by whitespace.
Hint: use the CONCAT function.

full_name
Hanna Chapman
Izabel Tang
Erica Porter
Shelbie Noble
Scarletta Correa
Juan Barron
Joy Castro
Izabel Tang
Anika Rabi
Kristof White
Tom Scott
Mick Daley
Elis Smith
Tonya Wright
Pukki Chase

10. Fetch all rows from the workers table ordered alphabetically by firstname (A-Z).

id	firstname	lastname	salary	department	title_id	join_date
9	Anika	Rabi	2110	Marketing	6	2019-03-28
13	Elis	Smith	980	IT	5	2019-06-01
3	Erica	Porter	1200	Accounts	2	2019-01-01
1	Hanna	Chapman	1000	HR	1	2019-01-01
8	Izabel	Tag	1400	R&D	2	2019-03-18
2	Izabel	Tang	950	HR	3	2019-01-05
7	Joy	Castro	500	Production	6	2019-03-12
6	Juan	Barron	800	R&D	2	2019-03-01
10	Kristof	White	1850	HR	6	2019-04-04
12	Mick	Daley	1450	Production	6	2019-05-10
15	Pukki	Chase	2000	Marketing	4	2019-06-06
5	Scarletta	Correa	2000	Marketing	6	2019-02-05
4	Shelbie	Noble	1500	IT	5	2019-02-01
11	Tom	Scott	1000	Production	6	2019-05-05
14	Tonya	Wright	700	Marketing	6	2019-06-06

11. Fetch all workersordered by department (A-Z). Same departments should order workers alphabetically by lastname (Z-A).

id	firstname	lastname	salary	department	title_id	join_date
3	Erica	Porter	1200	Accounts	2	2019-01-01
10	Kristof	White	1850	HR	6	2019-04-04
2	Izabel	Tang	950	HR	3	2019-01-05
1	Hanna	Chapman	1000	HR	1	2019-01-01
13	Elis	Smith	980	IT	5	2019-06-01
4	Shelbie	Noble	1500	IT	5	2019-02-01
14	Tonya	Wright	700	Marketing	6	2019-06-06
9	Anika	Rabi	2110	Marketing	6	2019-03-28
5	Scarletta	Correa	2000	Marketing	6	2019-02-05
15	Pukki	Chase	2000	Marketing	4	2019-06-06
11	Tom	Scott	1000	Production	6	2019-05-05
12	Mick	Daley	1450	Production	6	2019-05-10
7	Joy	Castro	500	Production	6	2019-03-12
8	Izabel	Tag	1400	R&D	2	2019-03-18
6	Juan	Barron	800	R&D	2	2019-03-01

12. Find all workers named `Joy` or `Tom`.

id	firstname	lastname	salary	department	title_id	join_date
7	Joy	Castro	500	Production	6	2019-03-12
11	Tom	Scott	1000	Production	6	2019-05-05

13. Find all the departments starting with the letter `A`.

department
Accounts

14. Find all the departments containing the letters `A` or `a`.

department
Accounts
Marketing

15. Find all the departments ending in letter `g`.

department
Marketing

16. Find all the workers that have a 5-letter name ending in the letter `a`.

id	firstname	lastname	salary	department	title_id	join_date
1	Hanna	Chapman	1000	HR	1	2019-01-01
3	Erica	Porter	1200	Accounts	2	2019-01-01
9	Anika	Rabi	2110	Marketing	6	2019-03-28
14	Tonya	Wright	700	Marketing	6	2019-06-06

17. Find all the workers that have salaries between 1000 and 1500.

id	firstname	lastname	salary	department	title_id	join_date
1	Hanna	Chapman	1000	HR	1	2019-01-01
3	Erica	Porter	1200	Accounts	2	2019-01-01
4	Shelbie	Noble	1500	IT	5	2019-02-01
8	Izabel	Tang	1400	R&D	2	2019-03-18
11	Tom	Scott	1000	Production	6	2019-05-05
12	Mick	Daley	1450	Production	6	2019-05-10

18. Find all the workers that joined the company in February 2019.

Hint: use the YEAR and MONTH functions.

id	firstname	lastname	salary	department	title_id	join_date
4	Shelbie	Noble	1500	IT	5	2019-02-01
5	Scarletta	Correa	2000	Marketing	6	2019-02-05

19. Write a query that will find how many workers are there in the HR department.

total_workers
3

20. Find how many workers are there in each department. Order the result set from department with most workers to department with least workers.

department	total_workers
Marketing	4
Production	3
HR	3
R&D	2
IT	2
Accounts	1

21. Print fullnames of all the workers that are managers.

department
Hanna Chapman
Izabel Tang

22. Find if there are any workers that have the same name and the same surname.

firstname	lastname
Izabel	Tang

23. Fetch all workers that have odd id values.

Hint: use the MOD function.

id	firstname	lastname	salary	department	title_id	join_date
1	Hanna	Chapman	1000	HR	1	2019-01-01
3	Erica	Porter	1200	Accounts	2	2019-01-01
5	Scarletta	Correa	2000	Marketing	6	2019-02-05
7	Joy	Castro	500	Production	6	2019-03-12
9	Anika	Rabi	2110	Marketing	6	2019-03-28
11	Tom	Scott	1000	Production	6	2019-05-05
13	Elis	Smith	980	IT	5	2019-06-01
15	Pukki	Chase	2000	Marketing	4	2019-06-06

24. Fetch all workers that have even id values.

id	firstname	lastname	salary	department	title_id	join_date
2	Izabel	Tang	950	HR	3	2019-01-05
4	Shelbie	Noble	1500	IT	5	2019-02-01
6	Juan	Barron	800	R&D	2	2019-03-01
8	Izabel	Tag	1400	R&D	2	2019-03-18
10	Kristof	White	1850	HR	6	2019-04-04
12	Mick	Daley	1450	Production	6	2019-05-10
14	Tonya	Wright	700	Marketing	6	2019-06-06

25. Create table `interns` and populate it with the data as in the screenshot on the right.

id	firstname	lastname	department
1	Monica	Arora	HR
2	Vivek	Bhati	Accounts
3	Juan	Carter	IT
4	Marley	Carter	IT
5	Andrea	Smith	Accounts
6	John	Light	IT

26. Write a query that will print the firstname and lastname of each worker and intern in the same result set. Next to each pair of names information about the employment type should be printed.

Hint: use UNION to merge the result of 2 queries.

firstname	lastname	type
Hanna	Chapman	worker
Izabel	Tang	worker
Erica	Porter	worker
Shelbie	Noble	worker
Scarletta	Correa	worker
Juan	Barron	worker
Joy	Castro	worker
Izabel	Tag	worker
Anika	Rabi	worker
Kristof	White	worker
Tom	Scott	worker
Mick	Daley	worker
Elis	Smith	worker
Tonya	Wright	worker
Pukki	Chase	worker
Monika	Arora	intern
Vivek	Bhati	intern
Juan	Carter	intern
Marley	Carter	intern
Andrea	Smith	intern
John	Light	Intern

27. Find the names that appear in both workers and interns tables (if any).

firstname
Juan

28. Find the names with which there are workers, but there are no interns.

name
Hanna
Izabel
Erica
Shelbie
Scarletta
Joy
Anika
Kristof
Tom
Mick
Elis
Tonya
Pukki

29. Find the person that has the 5th highest salary in the company.

id	firstname	lastname	salary	department	title_id	join_date
12	Mick	Daley	1450	Production	6	2019-05-10

30. Find the person that has the 2nd highest salary in the company. If more than 1 worker has the same salary, return all of them.

Hint: use subquery.

id	firstname	lastname	salary	department	title_id	join_date
5	Scarletta	Correa	2000	Marketing	6	2019-02-05
15	Pukki	Chase	2000	Marketing	4	2019-06-06

31. Find the ids of all the employees that have the same salary.

Hint: use the GROUP_CONCAT function.

worker_ids_with_same_salary
1, 11
5, 15

32. Find all the departments having less than three employees.

department
Accounts
IT
R&D

33. Print the name of every department along with the max salary in each one of the.

department	max_salary
Accounts	1200
HR	1850
IT	1500
Marketing	2110
Production	1450
R&D	1400

34. Follow-up

As you probably already noticed, the database is poorly designed taking into consideration the department column being stored as a string in the workers table. Let's fix that by extracting the department column to a new table without losing any of the existing data.

Steps:

1. Create new table called departments;
2. Write an sql query to fetch all unique department names from the workers table;
3. For each unique department enter new row in the departments table;
4. Add a department_id column in the workers table;
5. Make department_id FOREIGN KEY referencing the departments table.
6. Write update query that will automatically update all department_id values to the new ids;
7. Drop the department column from the workers table;