

# Exercises: Subqueries and JOINS

This document defines the **exercise assignments** for the [MySQL course @ Software University](#).

For problems from 1 to 11 (inclusively) use "soft\_uni" database and for the others – "geography".

## 1. Employee Address

Write a query that selects:

- **employee\_id**
- **job\_title**
- **address\_id**
- **address\_text**

Return the first 5 rows sorted by **address\_id** in ascending order.

### Example:

employee_id	job_title	address_id	address_text
142	Production Technician	1	108 Lakeside Court
30	Human Resources Manager	2	1341 Prospect St
...	...	...	...

## 2. Addresses with Towns

Write a query that selects:

- **first\_name**
- **last\_name**
- **town**
- **address\_text**

Sort the result by **first\_name** in ascending order then by **last\_name**. Select first 5 employees.

### Example:

first_name	last_name	town	address_text
A.Scott	Wright	Newport Hills	1400 Gate Drive
Alan	Brewer	Kenmore	8192 Seagull Court
...	...	...	...

## 3. Sales Employee

Write a query that selects:

- **employee\_id**
- **first\_name**
- **last\_name**
- **department\_name**

Sort the result by **employee\_id** in **descending order**. Select only **employees** from the "Sales" department.

### Example:

employee_id	first_name	last_name	department_name
290	Lynn	Tsoflias	Sales
289	Rachel	Valdez	Sales
...	...	...	...

## 4. Employee Departments

Write a query that selects:

- **employee\_id**
- **first\_name**
- **salary**
- **department\_name**

Filter only **employees** with **salary** higher than 15000. Return the first 5 rows sorted by **department\_id** in **descending order**.

### Example:

employee_id	first_name	salary	department_name
109	Ken	125500.00	Executive
140	Laura	60100.00	Executive
...	...	...	...

## 5. Employees Without Project

Write a query that selects:

- **employee\_id**
- **first\_name**

Filter only **employees** without a project. Return the first 3 rows sorted by **employee\_id** in **descending order**.

### Example:

employee_id	first_name
293	George
292	Martin

291	Svetlin
-----	---------

## 6. Employees Hired After

Write a query that selects:

- `first_name`
- `last_name`
- `hire_date`
- `dept_name`

Filter only **employees** hired after 1/1/1999 and from either the **"Sales"** or the **"Finance"** departments. Sort the result by `hire_date` (ascending).

### Example:

first_name	last_name	hire_date	dept_name
Debora	Poe	2001-01-19 00:00:00	Finance
Wendy	Kahn	2001-01-26 00:00:00	Finance
...	...	...	...

## 7. Employees with Project

Write a query that selects:

- `employee_id`
- `first_name`
- `project_name`

Filter only **employees** with a project, which has started after **13.08.2002** and it is still **ongoing** (no end date). Return the first 5 rows sorted by `first_name` then by `project_name` both in ascending order.

### Example

employee_id	first_name	project_name
44	A. Scott	Hitch Rack - 4-Bike
170	Alan	LL Touring Handlebars
...	...	...

## 8. Employee 24

Write a query that selects:

- `employee_id`
- `first_name`
- `project_name`

Filter all the **projects** of employees with **id 24**. If the project has started after **2005 inclusively** the return value should be **NULL**. Sort the result by **project\_name** alphabetically.

### Example

employee_id	first_name	project_name
24	David	NULL
24	David	NULL
24	David	Road-650

## 9. Employee Manager

Write a query that selects:

- **employee\_id**
- **first\_name**
- **manager\_id**
- **manager\_name**

Filter all **employees** with a manager who has **id equal to 3 or 7**. Return all rows sorted by **employee first\_name** in ascending order.

### Example

employee_id	first_name	manager_id	manager_name
122	Bryan	7	JoLynn
158	Dylan	3	Roberto
...	...	...	...

## 10. Employee Summary

Write a query that selects:

- **employee\_id**
- **employee\_name**
- **manager\_name**
- **department\_name**

Show the first 5 **employees** (only for employees who have a manager) with their **managers** and the **departments** they are in (show the departments of the **employees**). Order by **employee\_id**.

### Example

employee_id	employee_name	manager_name	department_name
1	Guy Gilbert	Jo Brown	Production
2	Kevin Brown	David Bradley	Marketing
...	...	...	...

## 11. Min Average Salary

Write a query that returns the value of the **lowest average salary** of all **departments**.

### Example:

min_average_salary
10866.6666

## 12. Highest Peaks in Bulgaria

Write a query that selects:

- **country\_code**
- **mountain\_range**
- **peak\_name**
- **elevation**

Filter all **peaks** in **Bulgaria** with **elevation** over **2835**. Return all rows sorted by **elevation** in **descending order**.

### Example

country_code	mountain_range	peak_name	elevation
BG	Rila	Musala	2925
BG	Pirin	Vihren	2914
...	...	...	...

## 13. Count Mountain Ranges

Write a query that selects:

- **country\_code**
- **mountain\_range**

Filter the **count** of the **mountain ranges** in the **United States, Russia and Bulgaria**. Sort result by **mountain\_range count** in **decreasing order**.

### Example

country_code	mountain_range
BG	6
RU	1
US	1

## 14. Countries with Rivers

Write a query that selects:

- **country\_name**

- **river\_name**

Find the first 5 **countries** with or without **rivers** in **Africa**. Sort them by **country\_name** in **ascending order**.

### Example

country_name	river_name
Algeria	Niger
Angola	Congo
Benin	Niger
Botswana	NULL
Burkina Faso	Niger

## 15. \*Continents and Currencies

Write a query that selects:

- **continent\_code**
- **currency\_code**
- **currency\_usage**

Find all **continents** and their most used **currency**. Filter any **currency** that is used in only one **country**. Sort the result by **continent\_code** and **currency\_code**.

### Example

continent_code	currency_code	currency_usage
AF	XOF	8
AS	AUD	2
AS	ILS	2
EU	EUR	26
NA	XCD	8
OC	USD	8

## 16. Countries Without Any Mountains

Find the count of all **countries** which don't have a **mountain**.

### Example

country_count
231

## 17. Highest Peak and Longest River by Country

For each **country**, find the **elevation of the highest peak** and the **length of the longest river**, sorted by the highest **peak\_elevation (from highest to lowest)**, then by the **longest river\_length (from longest to smallest)**, then by **country\_name (alphabetically)**. Display **NULL** when no data is available in some of the columns. Limit only the first 5 rows.

## Example

country_name	highest_peak_elevation	longest_river_length
China	8848	6300
India	8848	3180
Nepal	8848	2948
Pakistan	8611	3180
Argentina	6962	4880