

SQL CASE STUDY 2

HUMAN RESOURCES

SQL CASE STUDY

DATA IN MOTION
HUMAN RESOURCES



DATA IN MOTION

DATASETS

id	name	manager_id
Filter	Filter	Filter
1	HR	1
2	IT	2
3	Sales	3

Departments

id	name	hire_date ▾ ¹	job_title	department_id
Filter	Filter	Filter	Filter	Filter
1	John Doe	2018-06-20	HR Manager	1
2	Jane Smith	2019-07-15	IT Manager	2
3	Alice Johnson	2020-01-10	Sales Manager	3
4	Bob Miller	2021-04-30	HR Associate	1
5	Charlie Brown	2022-10-01	IT Associate	2
6	Dave Davis	2023-03-15	Sales Associate	3

Employees

Projects

id	name	start_date	end_date	department_id
Filter	Filter	Filter	Filter	Filter
1	HR Project 1	2023-01-01	2023-06-30	1
2	IT Project 1	2023-02-01	2023-07-31	2
3	Sales Project 1	2023-03-01	2023-08-31	3

QUESTION 1

Find the longest ongoing project for each department.

```
SELECT
    departments.id,
    departments.name AS department_name,
    projects.name,
    projects.start_date,
    projects.end_date,
    JULIANDAY(projects.end_date) - JULIANDAY(projects.start_date) AS project_length

FROM projects
JOIN departments
ON projects.department_id = departments.id

GROUP BY
    departments.id
ORDER BY
    project_length DESC
```

id	department_name	name	start_date	end_date	project_length
3	Sales	Sales Project 1	2023-03-01	2023-08-31	183.0
2	IT	IT Project 1	2023-02-01	2023-07-31	180.0
1	HR	HR Project 1	2023-01-01	2023-06-30	180.0

QUESTION 2

Find all employees who are not managers.

```
SELECT
    employees.id,
    employees.name

FROM employees
JOIN departments
ON employees.department_id = departments.id

WHERE
    employees.id != departments.manager_id
```

id	name
4	Bob Miller
5	Charlie Brown
6	Dave Davis

QUESTION 3

Find all employees who have been hired after the start of a project in their department.

```
SELECT
    employees.id,
    employees.name AS employee_name,
    employees.hire_date,
    departments.name AS department_name,
    projects.name AS projet_name,
    projects.start_date AS project_start_date
FROM employees
JOIN projects
JOIN departments
ON employees.department_id = projects.department_id
AND employees.department_id = departments.id
WHERE
    employees.hire_date > projects.start_date
```

id	employee_name	hire_date	department_name	projet_name	project_start_date
6	Dave Davis	2023-03-15	Sales	Sales Project 1	2023-03-01

QUESTION 4

Rank employees within each department based on their hire date (earliest hire gets the highest rank).

```
SELECT
    employees.name AS employee_name,
    employees.hire_date,
    departments.name AS department_name,
    DENSE_RANK() OVER (
        PARTITION BY departments.name
        ORDER BY hire_date
    ) rank
FROM employees
JOIN departments
ON employees.department_id = departments.id
```

employee_name	hire_date	department_name	rank
John Doe	2018-06-20	HR	1
Bob Miller	2021-04-30	HR	2
Jane Smith	2019-07-15	IT	1
Charlie Brown	2022-10-01	IT	2
Alice Johnson	2020-01-10	Sales	1
Dave Davis	2023-03-15	Sales	2

QUESTION 5

Find the duration between the hire date of each employee and the hire date of the next employee hired in the same department.

```
SELECT
    e.id,
    e.name AS employee_name,
    e.hire_date,
    e2.name AS next_employee_hierd,
    MIN(e2.hire_date) AS next_hire_date,
    JULIANDAY(MIN(e2.hire_date)) - JULIANDAY(e.hire_date) AS duration,
    d.name AS department_name
FROM
    employees e
JOIN employees e2 ON e.department_id = e2.department_id AND e.hire_date < e2.hire_date
JOIN departments d ON e.department_id = d.id
GROUP BY
    e.id,
    employee_name,
    e.hire_date,
    department_name
ORDER BY
    department_name,
    e.hire_date
```

id	employee_name	hire_date	next_employee_hierd	next_hire_date	duration	department_name
1	John Doe	2018-06-20	Bob Miller	2021-04-30	1045.0	HR
2	Jane Smith	2019-07-15	Charlie Brown	2022-10-01	1174.0	IT
3	Alice Johnson	2020-01-10	Dave Davis	2023-03-15	1160.0	Sales

THANK YOU

Name: Jermaine Herman Sangiwa

Email: jermaine@sangiwa.com

LinkedIn: <https://www.linkedin.com/in/jermainesangiwa>