Employee Project Tracker: SQL Queries and Outputs

This document provides SQL queries and their expected outputs for an Employee Project Tracker database, with tables for Employees, Projects, and Assignments. The queries extract insights such as top earners, overloaded employees, and cross-department projects, as well as an extra challenge query.

Sample Data

Employees Table:

employee_id	name	department	salary
1	Alice	HR	70000
2	Bob	IT	90000
3	Charlie	HR	65000
4	David	Marketing	85000
5	Eve	IT	92000

Projects Table:

project_id	project_name	department
101	HR System	HR
102	Marketing Revamp	Marketing
103	IT Infrastructure	IT

Assignments Table:

assignment_id	employee_id	project_id	hours_worked
1	1	101	120

| 2 | 2 | 103 | 150 |

| 3 | 3 | 101 | 90 |

| 4 | 4 | 102 | 130 |

| 5 | 2 | 101 | 60 |

| 6 | 5 | 103 | 180 |

| 7 | 2 | 102 | 50 |

| 8 | 5 | 102 | 40 |

1. Top Earners by Department

SELECT department, name, salary FROM Employees e WHERE salary = (SELECT MAX(salary) FROM Employees WHERE department = e.department);

department | name | salary

HR | Alice | 70000

IT | Eve | 92000

Marketing | David | 85000

2. Overloaded Employees

SELECT e.employee_id, e.name, COUNT(a.project_id) AS project_count FROM Employees e JOIN Assignments a ON e.employee_id = a.employee_id GROUP BY e.employee_id, e.name HAVING COUNT(a.project_id) > 3;

employee_id | name | project_count

No Results

3. Department Project Hours

SELECT p.department, SUM(a.hours_worked) AS total_hours FROM Projects p JOIN Assignments a ON p.project_id = a.project_id JOIN Employees e ON e.employee_id = a.employee_id GROUP BY p.department;

department | total_hours

HR | 270

IT | 390

Marketing | 220

4. Underpaid but Overworked

SELECT e.employee_id, e.name, e.department, e.salary, SUM(a.hours_worked) AS total_hours FROM Employees e JOIN Assignments a ON e.employee_id = a.employee_id GROUP BY e.employee_id, e.name, e.department, e.salary HAVING e.salary < (SELECT AVG(salary) FROM Employees WHERE department = e.department) AND SUM(a.hours_worked) > 100;

employee_id | name | department | salary | total_hours

1 | Alice | HR | 70000 | 120

5. Cross-Department Projects

SELECT p.project_id, p.project_name FROM Projects p JOIN Assignments a ON p.project_id = a.project_id JOIN Employees e ON e.employee_id = a.employee_id GROUP BY p.project_id, p.project_name HAVING COUNT(DISTINCT e.department) > 1;

project_id | project_name

102 | Marketing Revamp

Extra Challenge: Average Hours per Project

SELECT e.employee_id, e.name, AVG(a.hours_worked) AS avg_hours_per_project FROM Employees e JOIN Assignments a ON e.employee_id = a.employee_id GROUP BY e.employee_id, e.name HAVING COUNT(a.project_id) >= 2;

employee_id | name | avg_hours_per_project

2 | Bob | 86.67

5 | Eve | 110