Table: students

| student_id | name | age | department |
|---|---|---|---|
| 1 | Alice | 20 | IT |
| 2 | Bob | 22 | HR |
| 3 | Charlie | 21 | IT |
| 4 | Diana | 23 | Finance |
| 5 | Eve | 22 | HR |

## Questions

1. SELECT DISTINCT department
   FROM students;

Results →

| department |
|---|
| IT |
| HR |
| Finance |

2. SELECT department,
   AVG(age) AS avg_age
   FROM students
   GROUP BY department;

(because this is aggregated)

Results →

| department | avg_age |
|---|---|
| IT | 20.5 |
| HR | 22 |
| Finance | 23 |

3. SELECT COUNT(student_id) AS student_count,
   department
   FROM students
   GROUP BY department
   HAVING student_count > 1;

↳ filter data that is already aggregated.

Results →

| department | student_count |
|---|---|
| IT | 2 |
| HR | 2 |

4. SELECT *
   FROM students
   WHERE age BETWEEN 21 AND 23;

↳ inclusive statement

| student_id | name | age | department |
|---|---|---|---|
| 2 | Bob | 22 | HR |
| 3 | Charlie | 21 | IT |
| 4 | Diana | 23 | Finance |
| 5 | Eve | 22 | HR |

5. SELECT *
   FROM students
   WHERE department IN ('IT', 'HR')
   AND age > 21

| student_id | name | age | department |
|---|---|---|---|
| 2 | Bob | 22 | HR |
| 2 | Eve | 22 | HR |

**Table: courses**

| course_id | course_name | department | credits |
|---|---|---|---|
| 101 | SQL Basics | IT | 3 |
| 102 | Python | IT | 4 |
| 103 | Data Science | IT | 4 |
| 104 | Excel | Finance | 2 |
| 105 | Statistics | HR | 3 |

6.
```
SELECT department,
       SUM(credits) AS total_credits
FROM courses
Group By department
HAVING total_credit > 5
```

| department | total_credits |
|---|---|
| IT | 11 |

7.
```
SELECT *
FROM courses
WHERE credit != 4
```

| Course_id | course_name | department | Credit |
|---|---|---|---|
| 101 | SQL Basics | IT | 3 |
| 104 | Excel | Finance | 2 |
| 105 | Statistics | HR | 3 |

8.
```
SELECT Course_id,
       Course_name,
       Credits
FROM courses
ORDER BY Credits DESC
Limit 3;
```

| Course_id | course_name | Credits |
|---|---|---|
| 102 | Python | 4 |
| 103 | Data Science | 4 |
| 101 | SQL Basics | 3 |

**Table: enrollments**

| enrollment_id | student_id | course_id | grade |
|---|---|---|---|
| 1 | 1 | 101 | 85 |
| 2 | 2 | 102 | 78 |
| 3 | 3 | 103 | 90 |
| 4 | 4 | 104 | 88 |
| 5 | 5 | 105 | 82 |

9.
```
SELECT MAX(grade) AS max_grade,
       MIN(grade) AS min_grade,
       AVG(grade) AS avg_grade
FROM enrollments;
```

| max_grade | min_grade | avg_grade |
|---|---|---|
| 90 | 78 | 84.6 |

10.
```
SELECT COUNT(enrolment_id)
       AS enrolment_count
FROM enrollments
GROUP BY course
```

| Course_id | enrollment_count |
|---|---|
| 101 | 1 |
| 102 | 2 |
| 103 | 3 |
| 104 | 4 |
| 105 | 5 |

Table: salaries

| employee_id | name | department | salary | bonus |
|---|---|---|---|---|
| 1 | Tom | IT | 60000 | 5000 |
| 2 | Jerry | HR | 55000 | 4000 |
| 3 | Spike | Finance | 70000 | 6000 |
| 4 | Tyke | IT | 62000 | 5500 |
| 5 | Butch | HR | 54000 | 3500 |

11.
```
SELECT SUM(salary) AS total_salary,
       SUM(bonus) AS total_bonus,
       department
FROM salaries
GROUP BY department
```

| total_salary | total_bonus | department |
|---|---|---|
| 122 000 | 10 500 | IT |
| 109 000 | 7 500 | HR |
| 70000 | 6000 | Finance |

12.
```
SELECT department,
       AVG(salary) AS avg_salary
FROM salaries
GROUP BY department
Having avg_salary > 55 000
```

| department | avg-Salary |
|---|---|
| IT | 61 000 |

13.
```
SELECT employee_id,
       name,
       salary,
       bonus
       (salary + bonus) AS total_compensation
FROM salaries
WHERE (salary + bonus) > 60 000
```

| employee_id | name | salary | bonus | total_compensation |
|---|---|---|---|---|
| 3 | Spike | 70000 | 6 000 | 76 000 |
| 4 | Tyke | 62000 | 5 500 | 67 500 |

Table: projects

| project_id | project_name | department | budget |
|---|---|---|---|
| 1 | AI App | IT | 120000 |
| 2 | Payroll System | Finance | 80000 |
| 3 | Dashboard | IT | 150000 |
| 4 | Website | Marketing | 60000 |
| 5 | HR Portal | HR | 50000 |

14.
```
SELECT department,
       sum (budget) AS total_budget
       AVG (budget) AS avg-budget
FROM projects
GROUP BY department;
```

| department | total_budget | avg-budget |
|---|---|---|
| IT | 270 000 | 135 000 |
| Finance | 80 000 | 80 000 |
| Marketing | 60 000 | 60 000 |
| HR | 50 000 | 50 000 |

15.
```
SELECT *
FROM projects
WHERE budget BETWEEN 50 000 AND 120 000
  AND department NOT IN Marketing
```

| Prodget_id | project_name | department | budget |
|---|---|---|---|
| 1 | AI APP | IT | 120 000 |
| 2 | Payroll System | Finance | 80 000 |
| 3 | Dashboard | IT | 150 000 |