Exercise 2:

1.) SELECT DISTINCT department
FROM students;

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

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

4.) SELECT student_id,
name,
age,
department
FROM students
WHERE age BETWEEN 21 AND 23;

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

6.) SELECT department,
SUM(credits) AS total_credits
FROM courses
GROUP BY department
HAVING SUM(credits) > 5;

7.) SELECT course_id,
course_name,
department,
credits
FROM courses
WHERE credits != 4;

8.) SELECT course_id, course_name, credits
FROM courses
ORDER BY credits DESC
LIMIT 3;

9.) SELECT MAX(grade) AS max_grade
MIN(grade) AS min_grade
AVG(grade) AS avg_grade
FROM enrollment;

10.) SELECT course_id,
COUNT(enrollment_id) AS enrollment_count
FROM enrollments;
GROUP BY course_id

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

12) SELECT department,
AVG(salary) AS avg_salary
FROM salaries
GROUP BY department
HAVING AVG(salary) > 55000;

13.) SELECT employee_id,
name,
salary,
bonus,
(salary + bonus) AS total_compensation
FROM salaries
WHERE salary + bonus > 60000;

14.) SELECT department,
SUM(budget) AS total_budget
AVG(budget) AS avg_budget
FROM projects
GROUP BY department
HAVING AVG(budget) > 70000;

15.)
```
SELECT project_id,
       project_name,
       department,
       budget
FROM projects
WHERE budget BETWEEN 50000 AND 120000
  AND department != 'Marketing';
```