

# DB-UNIVERSITY-3 + BONUS

## Todo

Dopo aver testato le vostre query con `phpMyAdmin`, riportatele in un file `txt` o `md` e caricatelo nella vostra repo.

## Query

### Group by

1. Contare quanti iscritti ci sono stati ogni anno

```
SELECT YEAR(enrolment_date), COUNT(*)  
FROM students  
GROUP BY YEAR(enrolment_date);
```

2. Contare gli insegnanti che hanno l'ufficio nello stesso edificio

```
SELECT office_address, COUNT(*)  
FROM teachers  
GROUP BY office_address;
```

3. Calcolare la media dei voti di ogni appello d'esame  
(dell'esame vogliamo solo l' `id` )

```
SELECT exam_id, FLOOR(AVG(vote)) 'avg_vote'  
FROM exam_student  
GROUP BY exam_id;
```

#### 4. Contare quanti corsi di laurea ci sono per ogni dipartimento

```
SELECT department_id, COUNT(*)
FROM `degrees`
GROUP BY department_id;
```

## Join

#### 1. Selezionare tutti gli studenti iscritti al Corso di Laurea in Economia

```
SELECT students.name, students.surname, degrees.name
FROM students
    JOIN degrees
        ON students.degree_id = degrees.id
WHERE degrees.name LIKE 'Corso di Laurea in Economia';
```

#### 2. Selezionare tutti i Corsi di Laurea Magistrale del Dipartimento di Neuroscienze

```
SELECT *
FROM degrees
    JOIN departments
        ON degrees.department_id = departments.id
WHERE degrees.level = 'magistrale'
AND departments.name = 'Dipartimento di Neuroscienze';
```

#### 3. Selezionare tutti i corsi in cui insegna Fulvio Amato (id=44)

```
SELECT *
FROM courses
    JOIN course_teacher
```

```
        ON courses.id = course_teacher.teacher_id  
WHERE course_teacher.teacher_id = 44;
```

4. Selezionare tutti gli studenti con i dati relativi al corso di laurea a cui sono iscritti e il relativo dipartimento, in ordine alfabetico per cognome e nome

```
SELECT students.name, students.surname, degrees.name  
'corso_laurea', departments.name 'nome_dipartimento'  
FROM students  
    JOIN degrees  
        ON students.degree_id = degrees.id  
    JOIN departments  
        ON degrees.department_id = departments.id  
ORDER BY students.surname, students.name;
```

5. Selezionare tutti i corsi di laurea con i relativi corsi e insegnanti

```
SELECT degrees.name 'corso_laurea', courses.name  
'nome_corso', teachers.name 'nome_insegnante',  
teachers.surname 'cognome_insegnante'  
FROM degrees  
    JOIN courses  
        ON courses.degree_id = degrees.id  
    JOIN course_teacher  
        ON course_teacher.course_id = courses.id  
    JOIN teachers  
        ON course_teacher.teacher_id = teachers.id;
```

6. Selezionare tutti i docenti che insegnano nel Dipartimento di Matematica (54)

```
SELECT DISTINCT teachers*
FROM teachers
  JOIN course_teacher
    ON teachers.id = course_teacher.teacher_id
  JOIN courses
    ON course_teacher.course_id = courses.id
  JOIN degrees
    ON courses.degree_id = degrees.id
  JOIN departments
    ON degrees.department_id = departments.id
WHERE departments.name LIKE 'Dipartimento di
Matematica';
```

## Bonus

7. Selezionare per ogni studente il numero di tentativi sostenuti per ogni esame, stampando anche il voto massimo. Successivamente, filtrare i tentativi con voto minimo 18.

```
SELECT exam_student.exam_id,
       exam_student.student_id,
       COUNT(*) 'tentativi',
       MAX(exam_student.vote) 'voto_massimo'
FROM exam_student
WHERE exam_student.vote >= 18
GROUP BY exam_student.exam_id, exam_student.student_id;
```