# SQL – Data Manipulation Language

# The SELECT Statement

Students[sid, sname, email, age, group]

Teachers[tid, tname, no\_subjects]

Exams[sid, tid, grade]

Q1. Find the students that aged 21.

SELECT \*

FROM Students S

WHERE S.age=21

Q2. Find the students that aged 21, keep their name and email. Eliminate the duplicates

SELECT DISTINCT S.sname, S.email

FROM Students S

WHERE S.age=21

Range variable = an alias/abv/nickname for the table

Q3. Find the “10” grades (student name, tid)

SELECT S.sname, E.tid

FROM Students S, Exams E

WHERE E.grade = 10 AND S.sid = E.sid

SELECT Students.sname, Exams.tid

FROM Students, Exams

WHERE Exams.grade = 10 AND Students.sid = Exams.sid

LIKE Expression and Arithmetic expression

Q4. Find the students which have their name starting with ‘I’ and ending with ‘U’ and their name should have at least 5 characters (age, 2\*age, 18 - age)

SELECT S.sname, S.age, S.age \* 2 AS age2, age\_18 = S.age - 18

FROM Students S

WHERE S.sname LIKE ‘I\_\_\_%U’

Result set:

|  |  |  |  |
| --- | --- | --- | --- |
| sname | age | age2 | age\_18 |
| Ionescu | 23 | 46 | 5 |
|  | .. |  |  |

\_ - one character, any character

% - 0 or more characters

Set operations

UNION, INTERSECT, EXCEPT

UNION – eliminate the duplicates

UNION ALL – doesn’t eliminate the duplicates

Q5. Find the students ids with the age above 18 and a grade in the ‘t1’ teacher class.

SELECT S.sid

FROM Students S

WHERE S.age > 18

**INTERSECT**

SELECT E.sid

FROM Exams E

WHERE E.tid = ‘t1’

Q6. Find the students ids with the age above 18 or a grade in the ‘t1’ teacher class.

SELECT S.sid

FROM Students S

WHERE S.age > 18

**UNION**

SELECT E.sid

FROM Exams E

WHERE E.tid = ‘t1’

Q7. Find the students that have a grade at a teacher whose name is starting with A, but no grade at a teacher ‘t2’.

SELECT E.sid

FROM Exams E, Teachers T

WHERE E.tid = T.tid and T.tname LIKE ‘A%’

**EXCEPT**

SELECT E.sid

FROM Exams E

WHERE E.tid = ‘t2’

Nested queries – a query may have another subquery

* WHERE, HAVING, FROM

Q8. Find names of the students that are not graded by teacher t3.

SELECT S.sname

FROM Students S

WHERE S.sid NOT IN (SELECT E.sid

FROM Exams E

WHERE E.tid = ‘t3’)

SELECT S.sname

FROM Students S

WHERE NOT EXISTS

(SELECT \*

FROM Exams E

WHERE E.tid = ‘t3’ and E.sid = S.sid)

|  |  |  |  |
| --- | --- | --- | --- |
| S1 |  |  |  |
| S2 |  |  |  |
| S3 |  |  |  |

SELECT \*

FROM (SELECT E.sid

FROM Exams E

WHERE E.tid = ‘t3’)

ANY/ALL keywords

Q9. Find students who are older than at least one student named ‘Gigel’

SELECT \*

FROM Students S

WHERE S.age > ANY (

SELECT S2.age 23, 34, 18

FROM Students S2

WHERE S2.sname = ‘Gigel’

)

Q10. Find students who are older than all the students named ‘Gigel’

SELECT \*

FROM Students S

WHERE S.age > ALL (

SELECT S2.age 23, 34, 18

FROM Students S2

WHERE S2.sname = ‘Gigel’

)

JOIN Operators

Students

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| sid | sname | email | age | Group |
| S1 | Ana | a@cs.ro | 19 | 927 |
| S2 | Ada | [a1@cs.ro](mailto:a1@cs.ro) | 22 | 927 |
| S3 | Maria | [m@cs.ro](mailto:m@cs.ro) | 23 | 921 |
| S4 | Mihai | m1@cs.ro | 22 | 923 |

Teachers

|  |  |  |
| --- | --- | --- |
| tid | tname | No\_subjects |
| T1 | Name1 | 4 |
| T2 | Name2 | 5 |
| T3 | Name3 | 3 |

Exams

|  |  |  |
| --- | --- | --- |
| tid | sid | Grade |
| T1 | S1 | 8 |
| T1 | S3 | 9 |
| T2 | S5 | 10 |

INNER JOIN  
SELECT S.sname, T.tname

FROM Students S INNER JOIN Exams E ON S.sid = E.sid INNER JOIN Teachers T ON E.tid = T.tid

SELECT S.sname, T.tname

FROM Students S INNER JOIN Exams E ON S.sid = E.sid

Equivalent

SELECT S.sname, T.tname

FROM Students S, Exams E

WHERE S.sid = E.sid

|  |  |
| --- | --- |
| sname | Tname |
| Ana | Name1 |
| Maria | Name1 |
|  |  |
|  |  |

LEFT OUTER JOIN

SELECT S.sname, T.tname

FROM Students S LEFT OUTER JOIN Exams E ON S.sid = E.sid LEFT OUTER JOIN Teachers T ON E.tid = T.tid

|  |  |
| --- | --- |
| sname | tname |
| Ana | Name 1 |
| Ada | NULL |
| Maria | Name 1 |
| Mihai | NULL |
|  |  |
|  |  |

RIGHT OUTER JOIN

SELECT S.sname, T.tname

FROM Students S RIGHT OUTER JOIN Exams E ON S.sid = E.sid RIGHT OUTER JOIN Teachers T ON E.tid = T.tid

|  |  |
| --- | --- |
| sname | tname |
| Ana | Name 1 |
| Maria | Name 1 |
| NULL | Name 2 |
| NULL | Name 3 |
|  |  |
|  |  |

FULL OUTER JOIN (LEFT + RIGHT)

SELECT S.sname, T.tname

FROM Students S FULL OUTER JOIN Exams E ON S.sid = E.sid FULLOUTER JOIN Teachers T ON E.tid = T.tid

|  |  |
| --- | --- |
| sname | tname |
| Ana | Name 1 |
| Ada | NULL |
| Maria | Name 1 |
| Mihai | NULL |
| NULL | Name2 |
| NULL | Name 3 |

Aggregation operators:

* COUNT(\*)
* COUNT ([DISTINCT] A)
* SUM ([DISTINCT] A)
* AVG ([DISTINCT] A)
* MAX (A)
* MIN(A)

, where A is an attribute name in the table

Q15. Find the number of students

SELECT COUNT(\*)

FROM Students

Q16. Find the average age and the minimum age of students in the group 927.

SELECT AVG(S.age) AS avg\_age, MIN(S.age)

FROM Students S

WHERE S.group = 927

Q17. Find the number of groups that have at least one student name ‘Ion’

SELECT COUNT(DISTINCT S.group)

FROM Students S

WHERE S.sname = ‘Ion’

Q18. Find the name and age of the oldest student.

SELECT S.sname, S.age

FROM Student S

WHERE S.age = ANY (

SELECT MAX(S2.age)

FROM Students S2

)

Q19. For each teacher which has a no\_subjects above 2, find the no of grades and the avg grade

SELECT T.tid, COUNT(\*) as no\_grades, AVG(E.grade) as avg\_grade

FROM Teachers T, Exams E

WHERE T.tid = E.tid AND T.no\_subjects>2

GROUP BY T.tid

[HAVING ] condition