MTAT.03.105
Introduction to Databases

Lecture #7
Queries to a single table

Ljubov Jaanuska
(ljubov.jaanuska@ut.ee)
Lecture 6. Summary

• Foreign keys

• PCA update
Lecture 7. What will you learn

• Queries to a single table
SELECT [DISTINCT]
{<column_name, ...>}
FROM {<table_name, ...>}
[WHERE {<expression>}]  
[GROUP BY {<column_name, ...>}] 
[HAVING {<expression>}]  
[ORDER BY {<column_name, ...}>];

SELECT p_name
FROM player
WHERE
len(p_name)<6
GROUP BY p_name
HAVING
count(p_name)>1
ORDER BY p_name
SELECT

SELECT * FROM Club;

SELECT c_id, c_name FROM Club;

SELECT c_id FROM Club;

SELECT DISTINCT club_id FROM Player;

SELECT DISTINCT club_id, p_id FROM Player;
CONSTANTS

SELECT c_name, 0, ‘Tartu’, NULL, CURRENT DATE FROM Club;
SELECT c_id || ’, ’ || c_name FROM Club;

SELECT c_id || ’, ’ || c_name AS club_data FROM Club;

SELECT c_id || ’, ’ || c_name club_data, district_id FROM Club;
IF operator

IF ... THEN ... ELSE ... ENDIF

SELECT IF c_name='Laudnikud' THEN '!!!' ELSE c_name ENDIF, c_id FROM Club;

SELECT IF c_id=3 THEN 300
ELSE
  IF c_id=4 THEN 400
  ELSE c_id
  ENDIF
ENDIF, c_name FROM Club;
Functions

LEFT()
SUBSTRING()
YEAR()
...

SELECT LEFT (c_name, 2), SUBSTRING (c_name, 2, 4), YEAR (CURRENT DATE) FROM Club;
Aggregation functions

MIN()  
MAX()  
SUM()  
AVG()

SELECT MIN(c_name), MAX(c_name), SUM(district_id), AVG(c_id) FROM Club;
Once it is decided from **which columns** and **in what way** to present data in the result, think on **conditions WHERE**
Conditional expressions

IN          OR
BETWEEN     AND

SELECT * FROM Club WHERE c_id=54;
SELECT * FROM Club WHERE c_id=54 OR c_id=58;
SELECT * FROM Club WHERE c_id IN (54, 58)
SELECT * FROM Club WHERE c_id=54 AND c_id=58;
SELECT * FROM Club WHERE c_id BETWEEN 54 AND 58;
LIKE operator

LIKE

SELECT * FROM Club WHERE c_name LIKE 'v%';

SELECT * FROM Club WHERE c_name LIKE 'L_udnikud';

SELECT * FROM Club WHERE c_name LIKE 'a_[aeiou]%';

SELECT * FROM Club WHERE c_id LIKE '[5-6][1,7-9]';
COUNT() and GROUP BY

COUNT(*)
GROUP BY

SELECT COUNT(*) FROM Club;

SELECT COUNT(DISTINCT district_id) FROM Club;

SELECT district_id, COUNT(*) FROM Club;  -> shouldn’t work

SELECT district_id, COUNT(*) FROM Club GROUP BY district_id;
SELECT district_id, COUNT(*) FROM Club GROUP BY district_id HAVING COUNT(*) >1;
Ordering

ORDER BY

SELECT * FROM Club ORDER BY c_id  ASC;

SELECT c_id, district_id FROM Club ORDER BY district_id DESC, c_id;

SELECT c_name, district_id FROM Club ORDER BY 1 DESC, 2;
TOP ratings

TOP START AT

```
SELECT club_id, COUNT(*) AS s FROM Player GROUP BY club_id ORDER BY club_id DESC;

SELECT TOP 3 club_id, COUNT(*) AS s FROM Player GROUP BY club_id ORDER BY s DESC;

SELECT TOP 3 START AT 2 club_id, COUNT(*) AS s FROM Player GROUP BY club_id ORDER BY s DESC;
```
UNION

UNION ALL
UNION

SELECT c_id FROM Club
UNION ALL
SELECT district_id FROM Club;

SELECT c_id FROM Club
UNION
SELECT district_id FROM Club;
Example

```
SELECT LEFT(c_name, 1) AS c_abb, COUNT(*) nr FROM Club
GROUP BY c_abb ORDER BY LEFT(c_name, 1) DESC;
```
Tasks

1. When was the tournament venue in ‘Kambja’?
2. When was the last game?
3. Whose player id is 85? Show the result in one column in the form of ‘A. Tamm’.
4. Show ids of the games if the result of the game was 1:1.
5. How many players have the same surname ‘Hiis’?
6. Show player id, tournament id and number of games during the tournament if the player had more than 5 games during the tournament (separately for players with white and black chess pieces).
7. Show the names of the clubs and tournaments in one column.
8. Show first names of the players if the first name starts with ‘A’ or ‘H’ or ‘M’.
9. Show the first name and the surname of the player if the names begin with the same char (e.g. Anu Aru, Anu - first name, Aru - surname).
Tasks

10. Over all games, who are more successful – the ones who play with white chess pieces or black chess pieces? Show only the colour.
11. Show first names of the players whose first name is not longer than 4 chars. (HINT: look for the function LEN())
12. Show the tournament ids and the average duration of the games during each tournament (HINT: look for the function DATEDIFF())

PLEASE SUBMIT via Moodle by **28.03 23:55**
- .db file
- .log file
- .txt or .sql file with 12 queries only
- .txt file with 5 questions that can be solved by queries to one or several tables