

Quality Assurance 2.0 -> Lessons 21-22. Бази даних та SQL.

Практика 5. Робота з базою даних SQL

Інструкції SQL

В SQL існує близько 40 інструкцій. Кожна з них звертається до СУБД за виконанням конкретної дії.

Відповідно до призначення, інструкції поділяються на види:

- інструкції обробки даних;
- інструкції визначення даних;
- інструкції управління доступом;
- інструкції управління транзакціями;
- програмні інструкції.

Інструкції з обробки та представлення даних SQL:

- **SELECT** Отримує дані з однієї або декількох таблиць;
- **INSERT** Додає нові рядки в таблицю;
- **DELETE** Видаляє рядки з таблиці;
- **UPDATE** Оновлює дані, які вже існують в таблиці;
- **CREATE TABLE** Додає нову таблицю в БД;
- **DROP TABLE** Видаляє таблицю з БД;
- **ALTER TABLE** Змінює структуру існуючої таблиці.

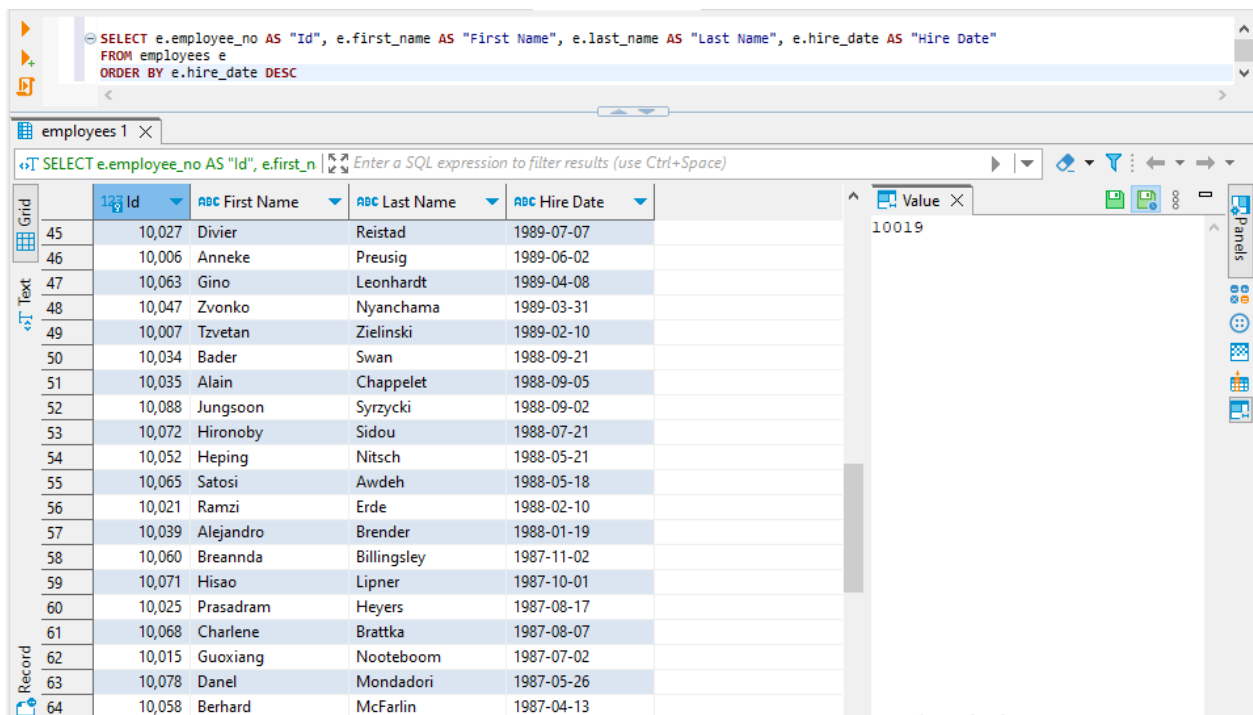
Кожна інструкція SQL починається з команди, яка описує дію, що виконується. Після команди йде одне або декілька речень. Речення описують дані, з якими працює інструкція, або містять інформацію про дію, яку виконує інструкція. Кожне речення також починається з ключового слова, наприклад WHERE (де), FROM (звідки), INTO (куди). Деякі речення в інструкціях є обов'язковими, інші – ні. Велика частина речень містить імена таблиць чи стовпців, деякі з них можуть містити додаткові ключові слова, константи та вирази.

Завдання

Встановити [DBeaver](#) та імпортувати [файл з Базою даних](#).

1. Обрати працівників з таблиці `employees` і відсортувати їх по даті найму (`hire_date`) за спаданням. Обрати айді (`employee_no`), ім'я (`first_name`), прізвище (`last_name`) та дату найму (`hire_date`).

```
SELECT e.employee_no AS "Id", e.first_name AS "First Name", e.last_name AS "Last Name", e.hire_date AS "Hire Date"  
FROM employees e  
ORDER BY e.hire_date DESC
```



The screenshot shows the DBeaver SQL editor interface. The SQL editor at the top contains the query: `SELECT e.employee_no AS "Id", e.first_name AS "First Name", e.last_name AS "Last Name", e.hire_date AS "Hire Date" FROM employees e ORDER BY e.hire_date DESC`. Below the editor, the results are displayed in a grid view for a table named 'employees 1'. The grid shows 20 rows of data, sorted by hire date in descending order. The columns are: Id, First Name, Last Name, and Hire Date. A 'Value' window is open on the right, showing the value '10019'.

	Id	First Name	Last Name	Hire Date
45	10,027	Divier	Reistad	1989-07-07
46	10,006	Anneke	Preusig	1989-06-02
47	10,063	Gino	Leonhardt	1989-04-08
48	10,047	Zvonko	Nyanchama	1989-03-31
49	10,007	Tzvetan	Zielinski	1989-02-10
50	10,034	Bader	Swan	1988-09-21
51	10,035	Alain	Chappelet	1988-09-05
52	10,088	Jungsoon	Syrzycki	1988-09-02
53	10,072	Hironoby	Sidou	1988-07-21
54	10,052	Heping	Nitsch	1988-05-21
55	10,065	Satosi	Awdeh	1988-05-18
56	10,021	Ramzi	Erde	1988-02-10
57	10,039	Alejandro	Brender	1988-01-19
58	10,060	Breannnda	Billingsley	1987-11-02
59	10,071	Hisao	Lipner	1987-10-01
60	10,025	Prasadram	Heyers	1987-08-17
61	10,068	Charlene	Brattka	1987-08-07
62	10,015	Guoxiang	Nooteboom	1987-07-02
63	10,078	Danel	Mondadori	1987-05-26
64	10,058	Berhard	McFarlin	1987-04-13

2. Обрати працівників з таблиці `employees`, які народилися після 1960 року (включно). Обрати ім'я (`first_name`), прізвище (`last_name`) та дату народження (`birth_date`).

```
SELECT e.first_name AS "First Name", e.last_name AS "Last Name", e.birth_date AS "Birth Date"  
FROM employees e  
WHERE e.birth_date >= "1960-01-01"
```

The screenshot shows a database management tool interface. At the top, the SQL query is entered in a text area:

```
SELECT e.first_name AS "First Name", e.last_name AS "Last Name", e.birth_date AS "Birth Date"  
FROM employees e  
WHERE e.birth_date >= "1960-01-01"
```

Below the query, the results are displayed in a grid view. The grid has three columns: "First Name", "Last Name", and "Birth Date". The results are sorted by birth date in ascending order. A tooltip is visible over the first row, showing the value "Bezalel".

	First Name	Last Name	Birth Date
1	Bezalel	Simmell	1964-06-02
2	Duangkaew	Piveteau	1963-06-01
3	Patricio	Bridgland	1960-10-04
4	Eberhardt	Terkki	1963-06-07
5	Kazuhito	Cappelletti	1961-05-02
6	Ramzi	Erde	1960-02-20
7	Divier	Reistad	1962-07-10
8	Domenick	Tempesti	1963-11-26
9	Jeong	Reistad	1960-08-09
10	Bader	Swan	1962-12-29
11	Pradeep	Makrucki	1963-07-22
12	Huan	Lortz	1960-07-20
13	Yishay	Tzvieli	1960-09-19
14	Mingsen	Casley	1961-09-21
15	Lucien	Rosenbaum	1960-07-23
16	Florian	Syrotiuk	1963-07-11
17	Basil	Tramer	1961-04-24
18	Heping	Nitsch	1961-02-26
19	Brendon	Bernini	1961-09-01

- Оберіть працівників з таблиці `employees`, ім'я яких починається на «Ма». Оберіть ім'я (`first_name`), прізвище (`last_name`) та дату народження (`birth_date`).

```
SELECT e.first_name AS "First Name", e.last_name AS "Last Name", e.birth_date AS "Birth Date"
FROM employees e
WHERE e.first_name LIKE "Ma%"
```

employees 1

SELECT e.first_name AS "First Name", e.last_name AS "Last Name", e.birth_date AS "Birth Date"
FROM employees e
WHERE e.first_name LIKE "Ma%"

Grid

	ABC First Name	ABC Last Name	ABC Birth Date
1	Mary	Sluis	1953-11-07
2	Mayuko	Warwick	1952-12-24
3	Magy	Stamatiou	1956-02-26
4	Mayumi	Schueller	1957-04-04
5	Margareta	Bierman	1960-09-06

Value X
Mary

- Обрати айді працівників з таблиці `employeeTerritories` які проживають у Києві, Дніпрі та Львові. Обрати айді (`employee_no`) та місто (`city`).

```
SELECT et.employee_no AS "Id", et.city AS "City" FROM employeeTerritories et
WHERE et.city IN ("Kyiv", "Dnipro", "Lviv")
```

employeeTerritories 1

SELECT et.employee_no AS "Id", et.city AS "City" FROM employeeTerritories et
WHERE et.city IN ("Kyiv", "Dnipro", "Lviv")

Grid

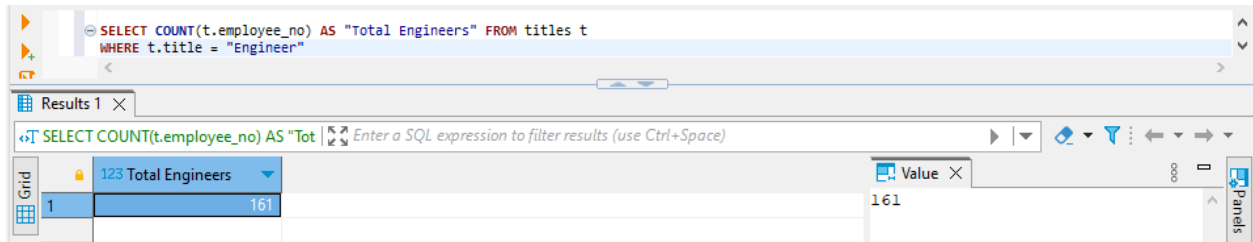
	123 Id	123 City
1	10,002	Dnipro
2	10,004	Dnipro
3	10,005	Kyiv
4	10,006	Lviv
5	10,007	Kyiv
6	10,009	Dnipro
7	10,014	Dnipro
8	10,016	Kyiv
9	10,018	Dnipro
10	10,020	Lviv
11	10,022	Dnipro
12	10,025	Dnipro
13	10,029	Dnipro
14	10,032	Lviv
15	10,033	Dnipro
16	10,040	Kyiv
17	10,042	Lviv
18	10,050	Dnipro
19	10,054	Dnipro
20	10,057	Lviv

Value X
Edit value:
10002
Dictionary (employees): (Define Description)
Type part of dictionary value to search

Value	Description
10002	Bezalel
10003	Parto
10004	Chirstian
10005	Kyoichi
10006	Anneke
10007	Tzvetan
10008	Saniya
10009	Sumant
10010	Duangkaew
10011	Mary
10012	Patricio
10013	Eberhardt
10014	Berni
10015	Guoxianq

5. Порахувати кількість працівників з таблиці titles які займають позицію (title) - "Engineer". Результуючій колонці присвоїти назву "total engineers".

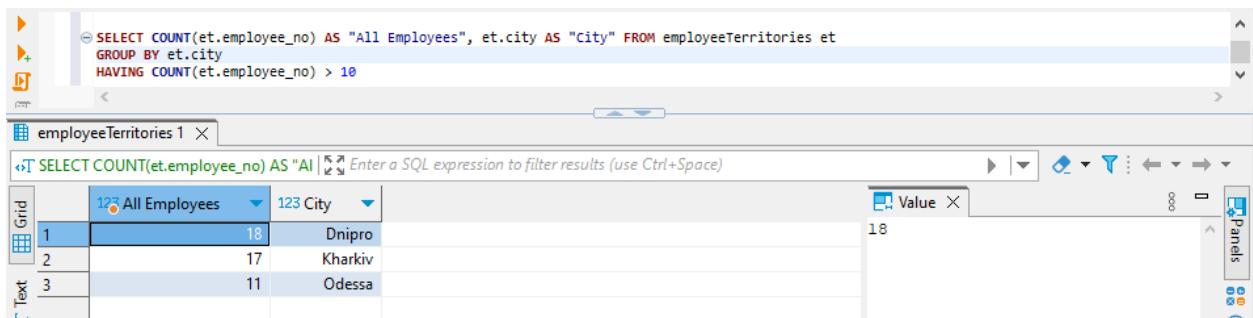
```
SELECT COUNT(t.employee_no) AS "Total Engineers" FROM titles t
WHERE t.title = "Engineer"
```



Grid	123 Total Engineers	Value
1	161	161

6. Вивести кількість працівників для кожного міста з таблиці employeeTerritories. Включаєте лише міста, у яких більше ніж 10 працівників. Вивести кількість працівників і назву міста.

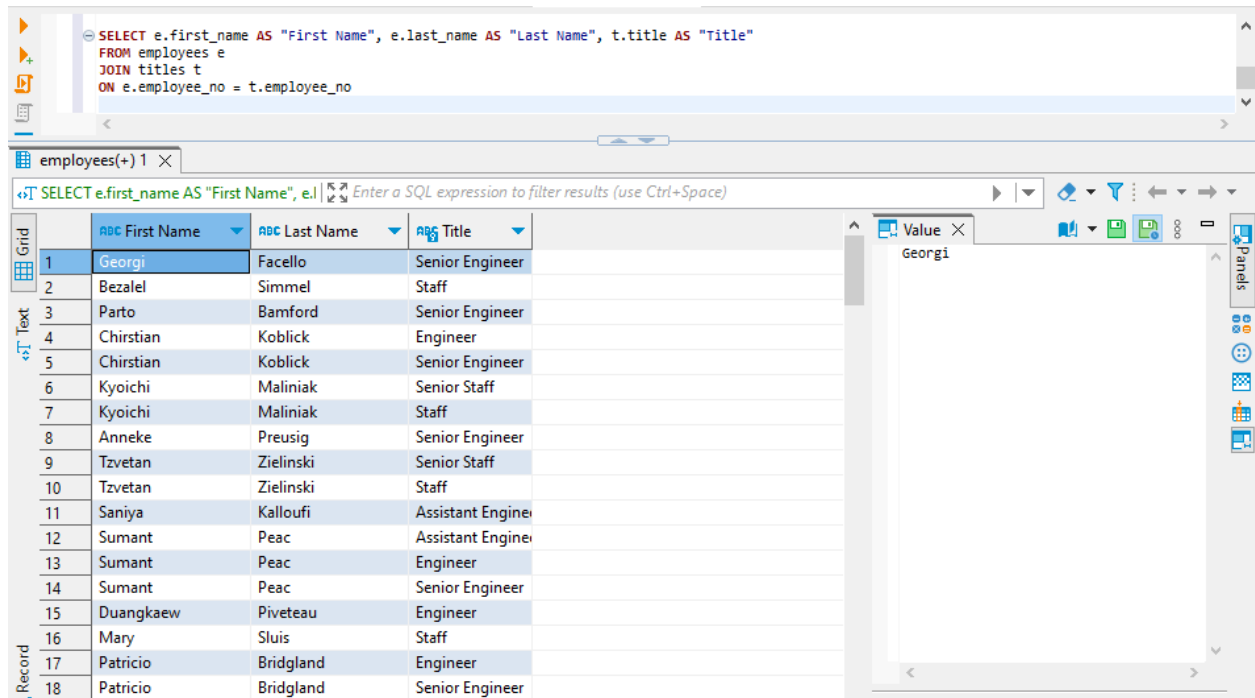
```
SELECT COUNT(et.employee_no) AS "All Employees", et.city AS "City" FROM employeeTerritories et
GROUP BY et.city
HAVING COUNT(et.employee_no) > 10
```



Grid	123 All Employees	123 City	Value
1	18	Dnipro	18
2	17	Kharkiv	
3	11	Odessa	

7. Обрати ім'я та прізвища працівників, їх позиції у компанії. Вивести ім'я (first_name), прізвище (last_name) та позицію (title) використовуючи таблиці employees і titles.

```
SELECT e.first_name AS "First Name", e.last_name AS "Last Name", t.title AS "Title"
FROM employees e
JOIN titles t
ON e.employee_no = t.employee_no
```



The screenshot shows a database management tool interface. At the top, a SQL query is entered in a text area:

```
SELECT e.first_name AS "First Name", e.last_name AS "Last Name", t.title AS "Title"
FROM employees e
JOIN titles t
ON e.employee_no = t.employee_no
```

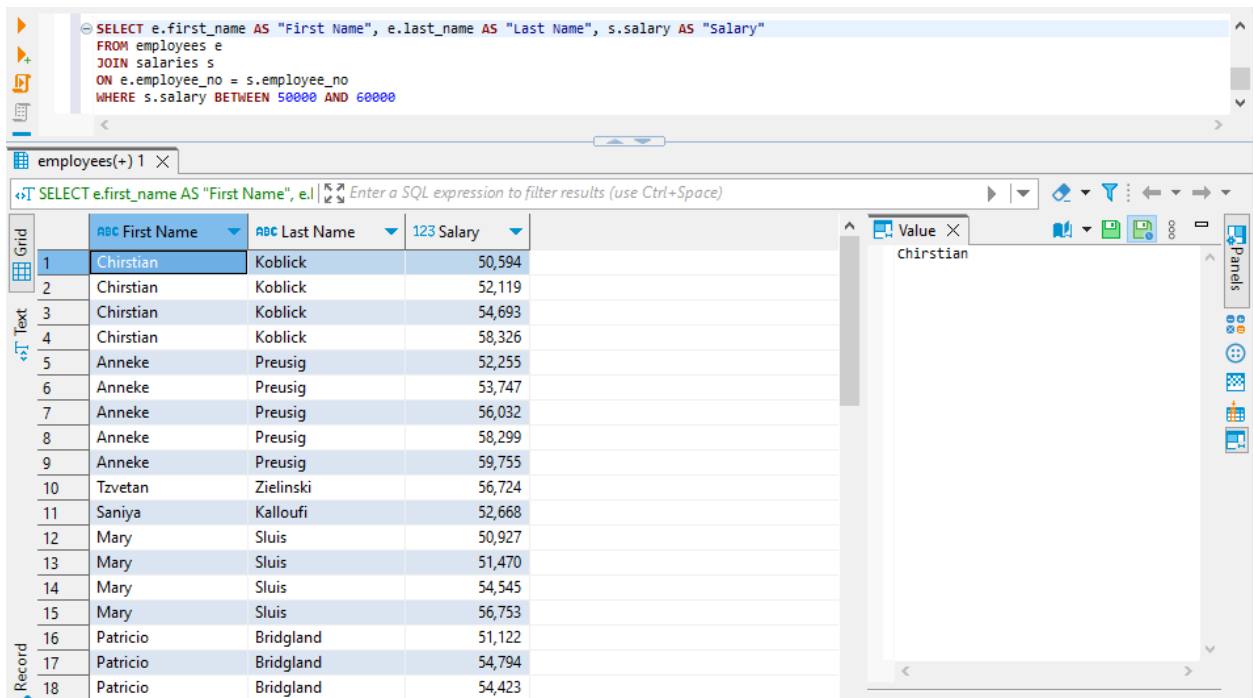
Below the query, a tab labeled "employees(+) 1 X" is active. The main area displays a grid of query results. The grid has four columns: "First Name", "Last Name", and "Title". The first row is highlighted in blue and contains the values "Georgi", "Facello", and "Senior Engineer".

	First Name	Last Name	Title
1	Georgi	Facello	Senior Engineer
2	Bezalel	Simmel	Staff
3	Parto	Bamford	Senior Engineer
4	Chirstian	Koblick	Engineer
5	Chirstian	Koblick	Senior Engineer
6	Kyoichi	Maliniak	Senior Staff
7	Kyoichi	Maliniak	Staff
8	Anneke	Preusig	Senior Engineer
9	Tzvetan	Zielinski	Senior Staff
10	Tzvetan	Zielinski	Staff
11	Saniya	Kalloufi	Assistant Engineer
12	Sumant	Peac	Assistant Engineer
13	Sumant	Peac	Engineer
14	Sumant	Peac	Senior Engineer
15	Duangkaew	Piveteau	Engineer
16	Mary	Sluis	Staff
17	Patricio	Bridgland	Engineer
18	Patricio	Bridgland	Senior Engineer

On the right side of the grid, there is a "Value" pane showing the value "Georgi" for the first row. The bottom of the interface shows a "Record" bar with a "Text" button and a "Grid" button.

8. Обрати ім'я та прізвища працівників, зарплата яких буде від 50000 до 60000. Обрати ім'я (*first_name*), прізвище (*last_name*) та зарплату (*salary*), використовуючи таблиці *employees* і *salaries*.

```
SELECT e.first_name AS "First Name", e.last_name AS "Last Name", s.salary AS "Salary"  
FROM employees e  
JOIN salaries s  
ON e.employee_no = s.employee_no  
WHERE s.salary BETWEEN 50000 AND 60000
```



The screenshot shows a SQL IDE interface. The top pane contains the SQL query. The bottom pane shows the results in a grid view. The grid has columns for First Name, Last Name, and Salary. The results are filtered to show only employees with a salary between 50,000 and 60,000.

	First Name	Last Name	Salary
1	Christian	Koblick	50,594
2	Christian	Koblick	52,119
3	Christian	Koblick	54,693
4	Christian	Koblick	58,326
5	Anneke	Preusig	52,255
6	Anneke	Preusig	53,747
7	Anneke	Preusig	56,032
8	Anneke	Preusig	58,299
9	Anneke	Preusig	59,755
10	Tzvetan	Zielinski	56,724
11	Saniya	Kalloufi	52,668
12	Mary	Sluis	50,927
13	Mary	Sluis	51,470
14	Mary	Sluis	54,545
15	Mary	Sluis	56,753
16	Patricio	Bridgland	51,122
17	Patricio	Bridgland	54,794
18	Patricio	Bridgland	54,423