

# Бази даних та SQL

Агрегатні функції  
Практика SQL

# Бази даних та SQL

## План уроку

- Агрегатні функції SQL
  - COUNT
  - MAX
  - MIN
  - SUM
  - AVG
- Групування
  - GROUP BY
  - HAVING
- Командна Практика

## Агрегатні функції SQL

# Бази даних та SQL

## Агрегатні функції SQL

Агрегатні функції SQL діють по відношенню до значень стовпця з метою отримання єдиного результуючого значення

Функції агрегації даних:

**AVG** – обчислює середнє значення

**COUNT** – обчислює кількість рядків у запиті

**SUM** – обчислює суму значень

**MIN** – обчислює найменше значення

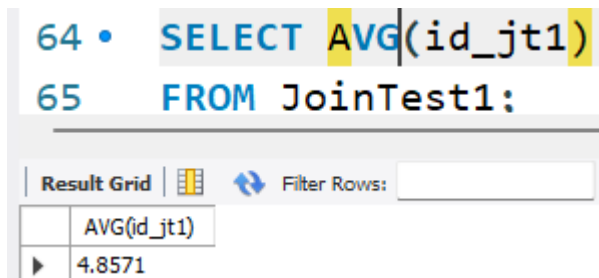
**MAX** – обчислює найбільше значення

# Бази даних та SQL

## ФУНКЦІЇ AVG() і COUNT()

Функція **AVG()** повертає середнє значення числового стовпця

```
SELECT AVG(column_name)  
FROM table_name
```



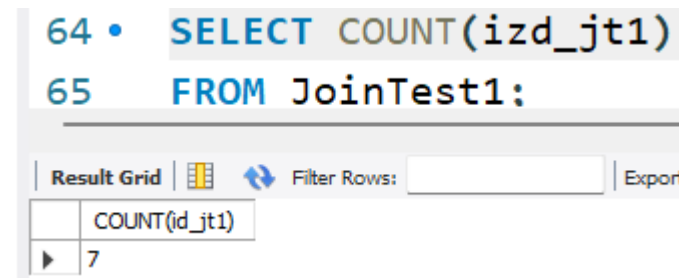
64 • SELECT AVG(id\_jt1)  
65 FROM JoinTest1;

Result Grid | Filter Rows:

AVG(id_jt1)
4.8571

Функція **COUNT()** повертає кількість рядків, які відповідають заданому критерію

```
SELECT COUNT(column_name)  
FROM table_name
```



64 • SELECT COUNT(izd\_jt1)  
65 FROM JoinTest1;

Result Grid | Filter Rows:  Export

COUNT(id_jt1)
7

# Бази даних та SQL

## ФУНКЦІЇ SUM(), MIN() і MAX()

Функція **SUM()** повертає загальну суму числового стовпця.

```
SELECT SUM(column_name)
FROM table_name
```

```
64 • SELECT SUM(id_jt1)
65 FROM JoinTest1;
```

Result Grid	Filter Rows:
SUM(id_jt1)	
▶ 34	

Функція **MIN()** повертає найменше значення вибраного стовпця

```
SELECT MIN(column_name)
FROM table_name
```

```
64 • SELECT MIN(id_jt1)
65 FROM JoinTest1;
```

Result Grid	Filter Rows:
MIN(id_jt1)	
▶ 1	

Функція **MAX()** повертає найбільше значення вибраного стовпця

```
SELECT MAX(column_name)
FROM table_name
```

```
64 • SELECT MAX(id_jt1)
65 FROM JoinTest1;
```

Result Grid	Filter Rows:
MAX(id_jt1)	
▶ 10	



## Групування результатів у SQL

# Бази даних та SQL

## GROUP BY

Інструкція **GROUP BY** групує рядки з однаковими значеннями в підсумкові рядки, як-от «знайти кількість клієнтів у кожній країні».

Оператор **GROUP BY** часто використовується з агрегатними функціями (COUNT(), MAX(), MIN(), SUM(), AVG()), щоб групувати набір результатів за одним або кількома стовпцями.

```
SELECT column_name(s)
FROM table_name
GROUP BY column_name(s)
```

```
72 • SELECT mark, SUM(speed)
73 FROM carsshop.cars
74 WHERE speed IN (225, 240, 250)
75 GROUP BY mark;
```

Result Grid			Filter Rows:	Export:	Wrap Cell Co
	mark	SUM(speed)			
▶	BMW	1125			
	Audi	250			





# Бази даних та SQL

## HAVING

Інструкція **HAVING** було додано до SQL, оскільки ключове слово WHERE не можна використовувати з агрегатними функціями.

**HAVING** – це **WHERE** для агрегатних функцій

```
SELECT column_name(s)  
FROM table_name  
GROUP BY column_name(s)  
HAVING condition
```

```
77 • SELECT mark, avg(speed) AS AVERAGE_SPEED  
78 FROM carsshop.cars group by mark  
79 HAVING avg(speed) > 230;
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
	mark	AVERAGE_SPEED	
▶	Audi	250.0000	
	Porsche	285.0000	



# Бази даних та SQL

## Командна Практика

# Бази даних та SQL

## Командна Практика

### Структура практики:

- Task 1: Data Definition (DDL)
- Task 2: Data Manipulation (DML)
- Task 3: JOINS and Retrieval
- Task 4: Aggregations and Grouping Functions

### Умови:

1. Працюємо у командах по 2 людини
2. Час на виконання - 1 година
3. Кожна команда отримує один з 3-х варіантів
4. Робимо завдання у MySQL Workbench
5. Після завершення розбираємо завдання різних команд разом

# Manual QA

Що ми сьогодні вивчили

# Бази даних та SQL

## План уроку

- Агрегатні функції SQL
  - COUNT
  - MAX
  - MIN
  - SUM
  - AVG
- Групування
  - GROUP BY
  - HAVING
- Командна Практика

# Manual QA

## Підсумки

Що одне, найголовніше, ви дізнались сьогодні?

# Manual QA

Дякую за увагу! До нових зустрічей!

# Інформаційний відеосервіс для розробників програмного забезпечення

