



**WEB** разработка

**SQL**

**заявки за напреднали**

# Съдържание

- **Aggregate** Functions
- **JOINs**

# Agregate Functions

# Agregate functions

Агрегиращите SQL функции връщат един резултат изчислен от много записи

Често се ползва и GROUP BY за групиране на записи по определена колона

# AVG

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

**или**

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

# COUNT

```
SELECT COUNT(*) FROM table_name;
```

**резултат** - връща броя на записите в съответната таблица

+ Options

```
COUNT('hotel_name')
```

10

# MAX

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

**результат** – максималната стойност от избраната колона

+ Options

**MAX(`rooms\_quantity`)**

1256

# MIN

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

**резултат – най-малката стойност от избраната колона**



# SUM

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

**резултат** – връща сумата от записите в съответната колона

подходящо е да се комбинира с GROUP BY по стойност в друга колона /  
например по id на град, за да сумираме броя на стаите в хотелите в определен град/

# Subqueries

```
SELECT ProductName, Price FROM Products  
WHERE Price > (SELECT AVG(Price) FROM Products);
```

# ORDER BY

```
SELECT ProductName, Price FROM Products ORDER BY Price DESC
```

По подразбиране в резултата, записите са подредени **ASC** /възходящо/.

# LIMIT

```
SELECT ProductName, Price FROM Products WHERE product_id = 22 LIMIT 1
```

/Преустановява търсенето след откриване на първия запис, отговарящ на условието/

# ORDER BY ... LIMIT

```
SELECT ProductName, Price FROM Products ORDER BY product_id LIMIT 2
```

Връща най-старите 2 резултата

# WHERE ... ORDER BY ... LIMIT

```
SELECT ProductName, Price FROM Products ORDER BY product_id DESC  
LIMIT 5
```

Връща най-новите 5 резултата

# JOINS

# JOINS

Предназначение –

свързваме таблици в БД чрез **FK** и **PK** /Internal relation/

Така имаме информация от няколко таблици и можем да изпълним **CRUD** с тази информация



# JOINS - 2

## **INNER JOIN** or

**JOIN:** Returns all rows when there is at least one match in BOTH tables

**LEFT JOIN:** Return all rows from the left table, and the matched rows from the right table

**RIGHT JOIN:** Return all rows from the right table, and the matched rows from the left table

**FULL JOIN:** Return all rows when there is a match in ONE of the tables

**JOIN / INNER JOIN**

# JOIN / INNER JOIN

```
SELECT column_name(s)
FROM table1
INNER JOIN table2
ON table1.column_name=table2.column_name;
```

**Или**

```
SELECT column_name(s)
FROM table1
JOIN table2
ON table1.column_name=table2.column_name;
```

# JOIN / INNER JOIN - 2

Като резултат виждаме само хотелите, за които има пълна информация от двете таблици

Hotel 1	vratsa	description 1	100
Hotel 2	Burgas	description 2	450

За да виждаме и другите хотели използваме ...

**LEFT JOIN/RIGHT JOIN**

# LEFT JOIN

```
SELECT column_name(s)
FROM table1
LEFT JOIN table2
ON table1.column_name=table2.column_name;
```

**или**

```
SELECT column_name(s)
FROM table1
LEFT OUTER JOIN table2
ON table1.column_name=table2.column_name;
```

# LEFT JOIN - 2

```
SELECT p.product_id, m.manufacturer_id,  
m.name FROM `product` p  
LEFT JOIN manufacturer m ON  
p.manufacturer_id=m.manufacturer_id  
WHERE 1
```

*Виждаме и продуктите без  
данни за производител.*

+ Настройки

product_id	manufacturer_id	name
28	5	HTC
29	6	Palm
30	9	Canon
31	NULL	NULL
32	8	Apple
33	NULL	NULL
34	8	Apple
35	NULL	NULL
36	8	Apple
40	8	Apple
41	8	Apple
42	8	Apple
43	8	Apple
44	8	Apple
45	8	Apple
46	10	Sony
47	7	Hewlett-Packard
48	8	Apple
49	NULL	NULL

# RIGHT JOIN - 1

```
SELECT p.product_id, m.manufacturer_id,  
m.name FROM `product` p  
RIGHT JOIN manufacturer m ON  
p.manufacturer_id=m.manufacturer_id  
WHERE 1
```

*Виждаме само данни за продуктите за  
наличните производители*

+ Настройки

product_id	manufacturer_id	name
28	5	HTC
29	6	Palm
30	9	Canon
31	NULL	NULL
32	8	Apple
33	NULL	NULL
34	8	Apple
35	NULL	NULL
36	8	Apple
40	8	Apple
41	8	Apple
42	8	Apple
43	8	Apple
44	8	Apple
45	8	Apple
46	10	Sony
47	7	Hewlett-Packard
48	8	Apple
49	NULL	NULL



# RIGHT JOIN - 2

*Добавяме е нов производител -  
без данни за продукт.*

```
SELECT p.product_id, m.manufacturer_id,  
m.name FROM `product` p  
RIGHT JOIN manufacturer m ON  
p.manufacturer_id=m.manufacturer_id  
WHERE 1
```

☐ Показване на всички

Брой редове:

25



Филтри

+ Настройки

product_id	manufacturer_id	name
28	5	HTC
29	6	Palm
30	9	Canon
32	8	Apple
34	8	Apple
36	8	Apple
40	8	Apple
41	8	Apple
42	8	Apple
43	8	Apple
44	8	Apple
45	8	Apple
46	10	Sony
47	7	Hewlett-Packard
48	8	Apple
NULL	11	New Man