



# Базы данных и SQL

Семинар 3.



Вопросы?

Вопросы?



Вопросы?



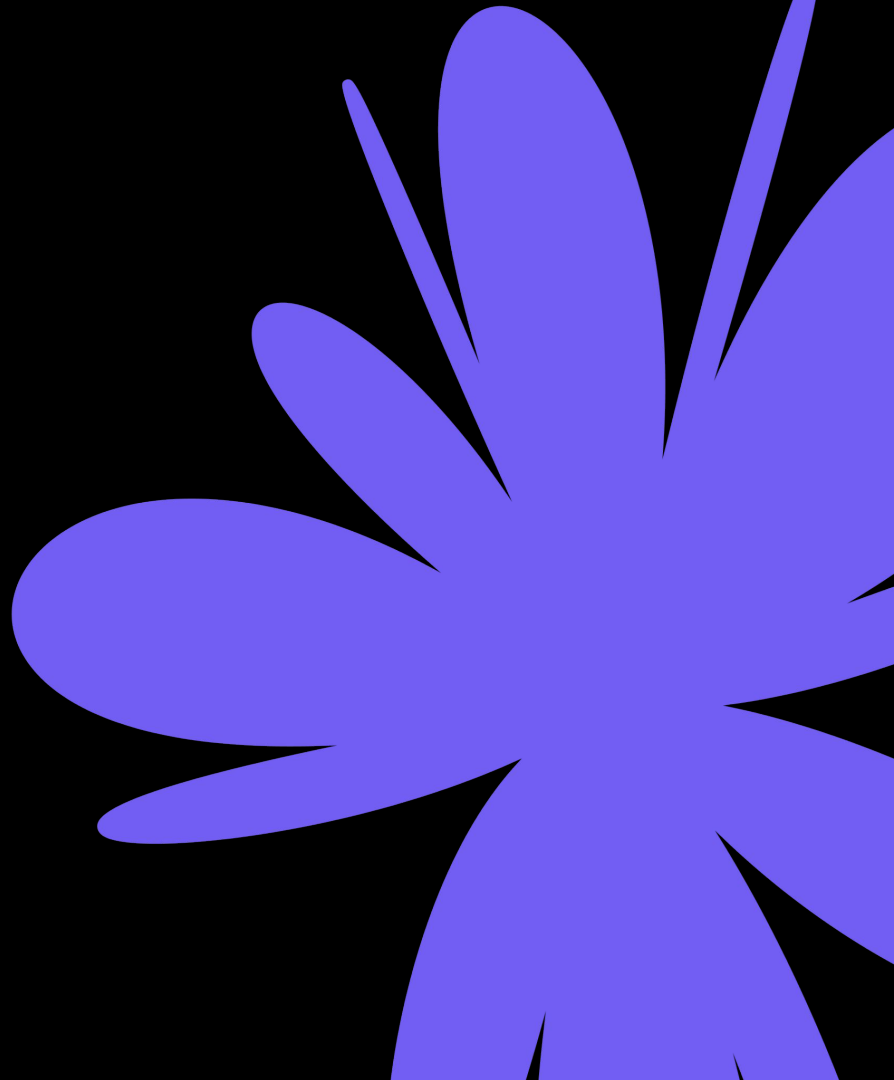
## План на сегодня:

- Quiz!
- Задания на top, order, distinct
- Задания на group by
- Задания на where + having
- Домашнее задание





# Quiz!



# Что такое агрегирующие функции?

1. функции, которые фильтруют значения
2. функции, которые сортируют значения
3. функции, которые работают с набором данных, превращая их в одно итоговое значение
4. функции, которые суммируют все значения



# Что такое агрегирующие функции?

1. функции, которые фильтруют значения
2. функции, которые сортируют значения
3. функции, которые работают с набором данных, превращая их в одно итоговое значение
4. функции, которые суммируют все значения



**Для подсчета количества записей в таблице  
«Persons» используется команда:**

1. `COUNT ROW IN Persons`
2. `SELECT COUNT(*) FROM Persons`
3. `SELECT ROWS FROM Persons`
4. `SELECT SUM(*) FROM Persons`



**Для подсчета количества записей в таблице  
«Persons» используется команда:**

1. COUNT ROW IN Persons
2. **SELECT COUNT(\*) FROM Persons**
3. SELECT ROWS FROM Persons
4. SELECT SUM(\*) FROM Persons





# Какая агрегатная функция используется для расчета суммы?

1. SUM
2. AVG
3. COUNT



# Какая агрегатная функция используется для расчета суммы?

1. SUM
2. AVG
3. COUNT



## **Запрос для выборки первых 14 записей из таблицы «Users» имеет вид:**

1. `SELECT * FROM Users LIMIT 14`
2. `SELECT * LIMIT 14 FROM Users`
3. `SELECT * FROM USERS`



## Запрос для выборки первых 14 записей из таблицы «Users» имеет вид:

1. `SELECT * FROM Users LIMIT 14`
2. `SELECT * LIMIT 14 FROM Users`
3. `SELECT * FROM USERS`



## Что покажет следующий запрос?

```
1 select DISTINCT seller_id  
2 order by seller_id  
3 from Orders;
```

1. Уникальные ID продавцов, отсортированные по возрастанию
2. Уникальные ID продавцов, отсортированные по убыванию
3. Ничего, запрос составлен неверно, ORDER BY всегда ставится в конце запроса
4. Неотсортированные никак уникальные ID продавцов



## Что покажет следующий запрос?

```
1 select DISTINCT seller_id  
2 order by seller_id  
3 from Orders;
```

1. Уникальные ID продавцов, отсортированные по возрастанию
2. Уникальные ID продавцов, отсортированные по убыванию
3. Ничего, запрос составлен неверно, ORDER BY всегда ставится в конце запроса
4. Неотсортированные никак уникальные ID продавцов



## Что покажет следующий запрос:

```
1 select seller_id, count(*)  
2 from Orders  
3 GROUP BY seller_id  
4 HAVING seller_id IN (2,4,6);
```

1. количество заказов сгруппированное по продавцам 2, 4 и 6
2. количество продавцов, у которых 2, 4 или 6 товаров
3. ничего, запрос составлен неверно, HAVING указывается до группировки
4. ничего, запрос составлен неверно, для указания условия должно быть использовано WHERE



## Что покажет следующий запрос:

```
1 select seller_id, count(*)  
2 from Orders  
3 GROUP BY seller_id  
4 HAVING seller_id IN (2,4,6);
```

1. количество заказов сгруппированное по продавцам 2, 4 и 6
2. количество продавцов, у которых 2, 4 или 6 товаров
3. ничего, запрос составлен неверно, HAVING указывается до группировки
4. ничего, запрос составлен неверно, для указания условия должно быть использовано WHERE





# ORDER BY. Табличка для работы

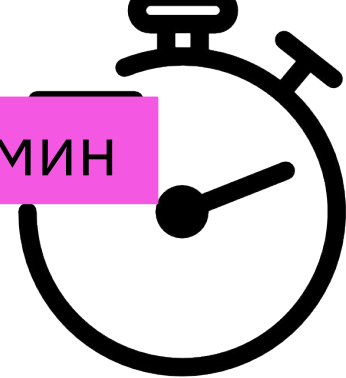
**SELECT** поле1, поле2, ...

**FROM** имя\_таблицы

**WHERE** условие

**ORDER BY** поле1, поле2, ... **ASC(DESC)**;

10 мин



+ Параметры

<div><div>←</div><div>→</div></div>				<div>▼</div>	id	name	surname	specialty	seniority	salary	age
<div><input type="checkbox"/></div>	<div></div>	<div></div>	<div></div>	<div><input checked="" type="checkbox"/></div>	1	Вася	Васькин	начальник	40	100000	60
<div><input type="checkbox"/></div>	<div></div>	<div></div>	<div></div>	<div><input checked="" type="checkbox"/></div>	2	Петя	Петькин	начальник	8	70000	30
<div><input type="checkbox"/></div>	<div></div>	<div></div>	<div></div>	<div><input checked="" type="checkbox"/></div>	3	Катя	Каткина	инженер	2	70000	25
<div><input type="checkbox"/></div>	<div></div>	<div></div>	<div></div>	<div><input checked="" type="checkbox"/></div>	4	Саша	Сашкин	инженер	12	50000	35
<div><input type="checkbox"/></div>	<div></div>	<div></div>	<div></div>	<div><input checked="" type="checkbox"/></div>	5	Иван	Иванов	рабочий	40	30000	59
<div><input type="checkbox"/></div>	<div></div>	<div></div>	<div></div>	<div><input checked="" type="checkbox"/></div>	6	Петр	Петров	рабочий	20	25000	40
<div><input type="checkbox"/></div>	<div></div>	<div></div>	<div></div>	<div><input checked="" type="checkbox"/></div>	7	Сидор	Сидоров	рабочий	10	20000	35
<div><input type="checkbox"/></div>	<div></div>	<div></div>	<div></div>	<div><input checked="" type="checkbox"/></div>	8	Антон	Антонов	рабочий	8	19000	28
<div><input type="checkbox"/></div>	<div></div>	<div></div>	<div></div>	<div><input checked="" type="checkbox"/></div>	9	Юра	Юркин	рабочий	5	15000	25
<div><input type="checkbox"/></div>	<div></div>	<div></div>	<div></div>	<div><input checked="" type="checkbox"/></div>	10	Максим	Воронин	рабочий	2	11000	22
<div><input type="checkbox"/></div>	<div></div>	<div></div>	<div></div>	<div><input checked="" type="checkbox"/></div>	11	Юра	Галкин	рабочий	3	12000	24
<div><input type="checkbox"/></div>	<div></div>	<div></div>	<div></div>	<div><input checked="" type="checkbox"/></div>	12	Люся	Люськина	уборщик	10	10000	49

↑

☐ Отметить все

С отмеченными:

☒



# Операторы сортировки

10 мин

1. Выведите все записи, отсортированные по полю "age" по возрастанию
2. Выведите все записи, отсортированные по полю "name"
3. Выведите записи полей "name", "surname", "age", отсортированные по полю "name" в алфавитном порядке по убыванию
4. Выполните сортировку по полям "name" и "age" по убыванию



# Операторы сортировки. DISTINCT, LIMIT

## DISTINCT:

**SELECT DISTINCT** поле1, поле2, ...

**FROM** имя\_таблицы;

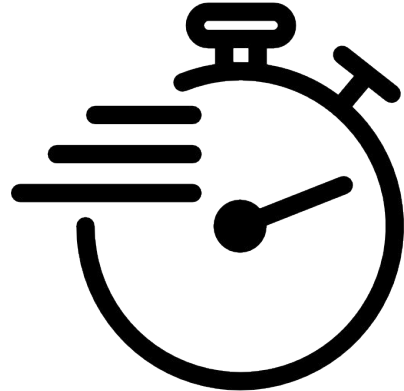
## LIMIT:

**LIMIT**[смещение\_относительно\_начала,  
количество\_извлекаемых\_строк

10 мин



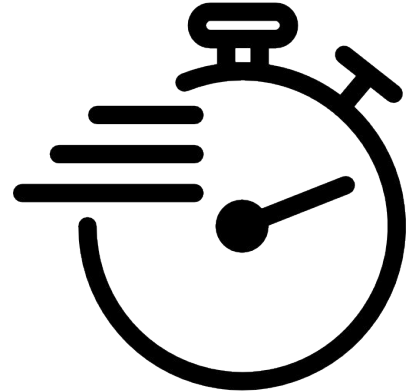
# Операторы сортировки. DISTINCT, LIMIT



1. Выведите уникальные (неповторяющиеся) значения полей "name"
2. Выведите первые две первые записи из таблицы
3. Пропустите первые 4 строки ("id" = 1, "id" = 2,"id" = 3,"id" = 4) и извлеките следующие 3 строки ("id" = 5, "id" = 6, "id" = 7)
- 4\*.** Пропустите две последние строки (где id=12, id=11) и извлекаются следующие за ними 3 строки (где id=10, id=9, id=8)



# Агрегатные функции. Таблица для работы



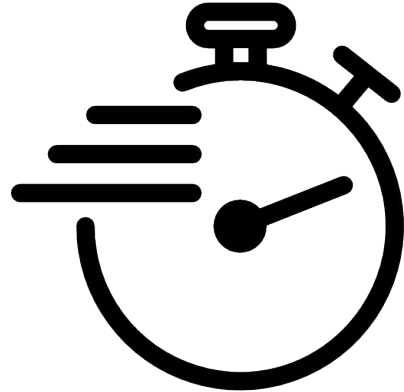
```
mysql> SELECT * FROM employee_tbl;
```

id	name	work_date	daily_typing_pages
1	John	2007-01-24	250
2	Ram	2007-05-27	220
3	Jack	2007-05-06	170
3	Jack	2007-04-06	100
4	Jill	2007-04-06	220
5	Zara	2007-06-06	300
5	Zara	2007-02-06	350

```
7 rows in set (0.00 sec)
```



# Агрегатные функции



1. Рассчитайте общее количество всех страниц `daily_typing_pages`
2. Выведите общее количество напечатанных страниц каждым человеком (с помощью предложения `GROUP BY`)
3. Посчитайте количество записей в таблице
4. Выведите количество имен, которые являются уникальными
5. Найдите среднее арифметическое по количеству ежедневных страниц для набора (`daily_typing_pages`)



**Ваши вопросы?**

**Перерыв**



# GROUP BY

**SELECT** \* **FROM** имя\_таблицы  
**WHERE** условие  
**GROUP BY** поле\_для\_группировки

20 мин



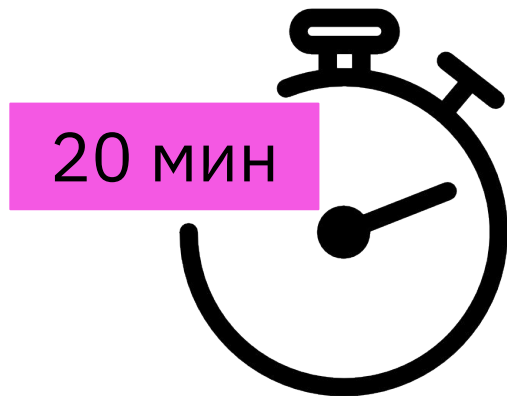
id	name	age	salary
1	Дима	23	100
2	Петя	23	200
3	Вася	23	300
4	Коля	24	1000
5	Иван	24	2000





# GROUP BY

1. Сгруппируйте поля по возрасту (будет 3 группы - 23 года, 24 года и 25 лет). Для каждой группы найдите суммарную зарплату
2. Сгруппируйте поля по возрасту (будет 3 группы - 23 года, 24 года и 25 лет). Найдите максимальную заработную плату внутри группы
3. Сгруппируйте поля по возрасту (будет 3 группы - 23 года, 24 года и 25 лет). Найдите минимальную заработную плату внутри группы



# HAVING

GROUP BY поле HAVING условие

## Задания:

1. Выведите только те строки, в которых суммарная зарплата больше или равна 1000
2. Выведите только те группы, в которых количество строк меньше или равно двум
3. Выведите только те группы, в которых количество строк меньше или равно двум. Для решения используйте оператор “BETWEEN”
- 4.\* Выведите только те группы, в которых количество строк меньше или равно двум. Для решения используйте оператор “IN”



20 мин



**Ваши вопросы?**



# Домашнее задание

ТАБЛИЦА 1: ПРОДАВЦЫ (SALESPEOPLE)

snum	sname	city	comm
1001	Peel	London	.12
1002	Serres	San Jose	.13
1004	Motika	London	.11
1007	Rifkin	Barcelona	.15
1003	Axelrod	New York	.10

ТАБЛИЦА 2: ЗАКАЗЧИКИ (CUSTOMERS)

cnum	cname	city	rating	snum
2001	Hoffman	London	100	1001
2002	Giovanni	Rome	200	1003
2003	Liu	SanJose	200	1002
2004	Grass	Berlin	300	1002
2006	Clemens	London	100	1001
2008	Cisneros	SanJose	300	1007
2007	Pereira	Rome	100	1004



# Домашнее задание

ТАБЛИЦА 3: ЗАКАЗЫ (ORDERS)

onum	amt	odate	cnum	snum
3001	18.69	10/03/1990	2008	1007
3003	767.19	10/03/1990	2001	1001
3002	1900.10	10/03/1990	2007	1004
3005	5160.45	10/03/1990	2003	1002
3006	1098.16	10/03/1990	2008	1007
3009	1713.23	10/04/1990	2002	1003
3007	75.75	10/04/1990	2004	1002
3008	4723.00	10/05/1990	2006	1001
3010	1309.95	10/06/1990	2004	1002
3011	9891.88	10/06/1990	2006	1001



## Домашнее задание

1. Напишите запрос, который вывел бы таблицу со столбцами в следующем порядке: city, sname, snum, comm. (к первой или второй таблице, используя SELECT)
2. Напишите команду SELECT, которая вывела бы оценку(rating), сопровождаемую именем каждого заказчика в городе San Jose. (“заказчики”)
3. Напишите запрос, который вывел бы значения snum всех продавцов из таблицы заказов без каких бы то ни было повторений. (уникальные значения в “snum” “Продавцы”)
- 4\*. Напишите запрос, который бы выбирал заказчиков, чьи имена начинаются с буквы G. Используется оператор "LIKE": (“заказчики”) <https://dev.mysql.com/doc/refman/8.0/en/string-comparison-functions.html>
5. Напишите запрос, который может дать вам все заказы со значениями суммы выше чем \$1,000. (“Заказы”, “amt” - сумма)
6. Напишите запрос который выбрал бы наименьшую сумму заказа.  
(Из поля “amt” - сумма в таблице “Заказы” выбрать наименьшее значение)
7. Напишите запрос к таблице “Заказчики”, который может показать всех заказчиков, у которых рейтинг больше 100 и они находятся не в Риме.



# Домашнее задание

Таблица для работы (из классной работы)

+ Параметры

				id	name	surname	specialty	seniority	salary	age
<input type="checkbox"/>				1	Вася	Васькин	начальник	40	100000	60
<input type="checkbox"/>				2	Петя	Петькин	начальник	8	70000	30
<input type="checkbox"/>				3	Катя	Каткина	инженер	2	70000	25
<input type="checkbox"/>				4	Саша	Сашкин	инженер	12	50000	35
<input type="checkbox"/>				5	Иван	Иванов	рабочий	40	30000	59
<input type="checkbox"/>				6	Петр	Петров	рабочий	20	25000	40
<input type="checkbox"/>				7	Сидор	Сидоров	рабочий	10	20000	35
<input type="checkbox"/>				8	Антон	Антонов	рабочий	8	19000	28
<input type="checkbox"/>				9	Юра	Юркин	рабочий	5	15000	25
<input type="checkbox"/>				10	Максим	Воронин	рабочий	2	11000	22
<input type="checkbox"/>				11	Юра	Галкин	рабочий	3	12000	24
<input type="checkbox"/>				12	Люся	Люськина	уборщик	10	10000	49

↑ ☐ Отметить все С отмеченными:

1. Отсортируйте поле “зарплата” в порядке убывания и возрастания
2. \*\* Отсортируйте по возрастанию поле “Зарплата” и выведите 5 строк с наибольшей заработной платой (возможен подзапрос)
3. Выполните группировку всех сотрудников по специальности , суммарная зарплата которых превышает 100000



# Рефлексия



**Был урок полезен вам?**



**Узнали вы что-то новое?**



**Что было сложно?**





Спасибо  
за внимание

A yellow smiley face is drawn over the text. It has two vertical lines for eyes and a curved line for a mouth, positioned to the right of the word 'Спасибо' and below the word 'за'.