



Базы данных и SQL

Семинар 5.

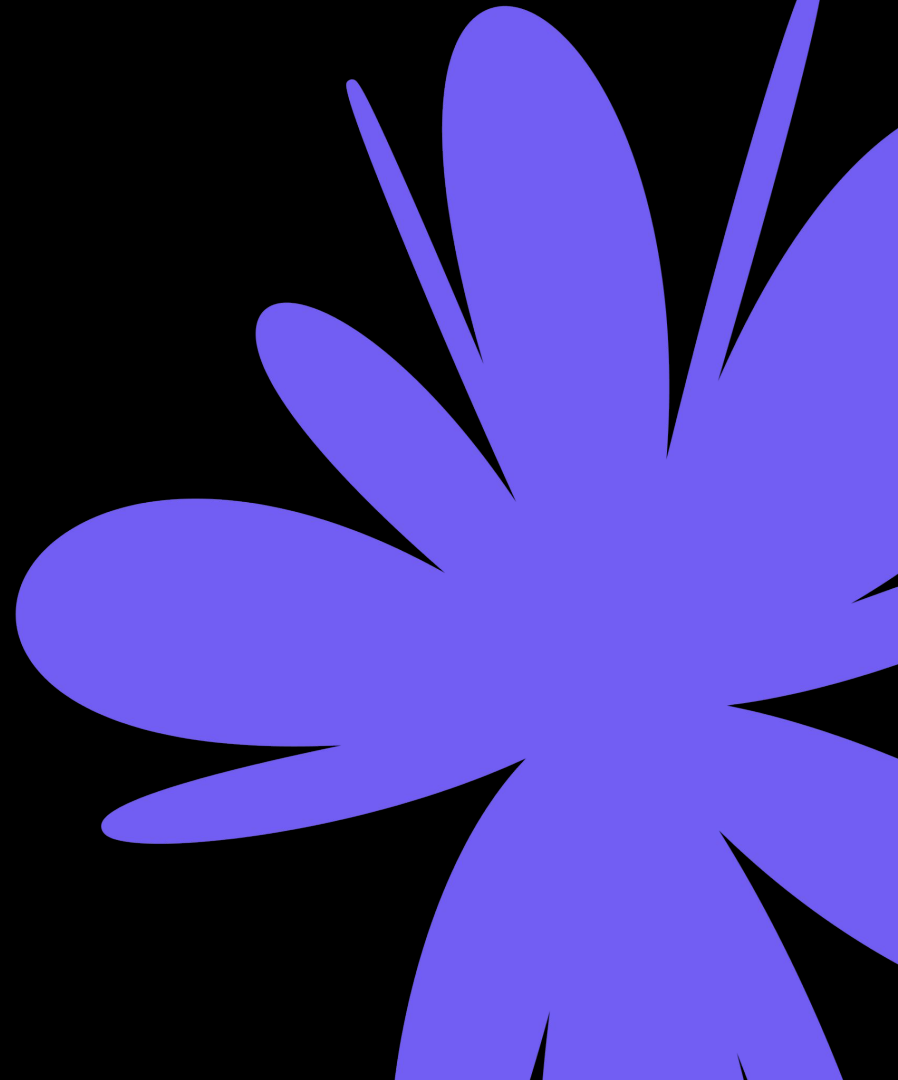
План на сегодня:

- Quiz!
- Оконные функции
- Представления
- Домашнее задание





Quiz!



Для создания новой виртуальной таблицы, которая базируется на результатах сделанного ранее SQL запроса, используется команда:

1. CREATE VIRTUAL TABLE
2. CREATE VIEW
3. ALTER VIEW



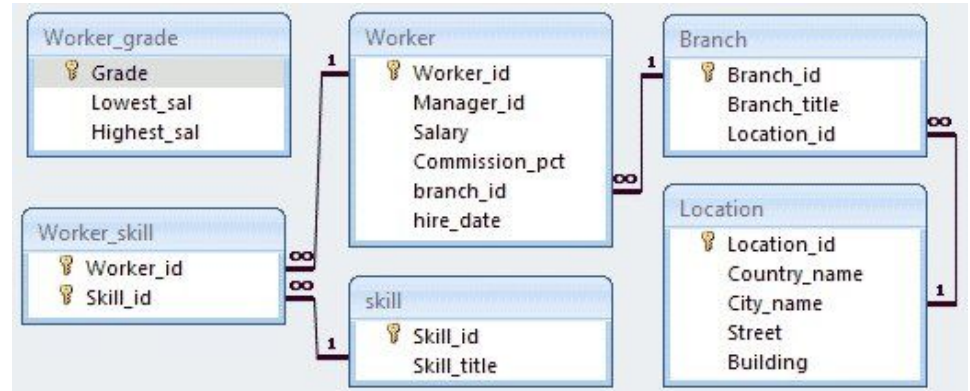
Для создания новой виртуальной таблицы, которая базируется на результатах сделанного ранее SQL запроса, используется команда:

1. CREATE VIRTUAL TABLE
2. CREATE VIEW
3. ALTER VIEW



Для создания представления, в которое должны попасть только имена сотрудников, работающих в отделе Research, используется запрос:

```
CREATE _____  
SELECT Worker_name FROM Worker w, Branch b  
WHERE w.Branch_id = b.Branch_id AND Branch_title LIKE  
'Research'
```

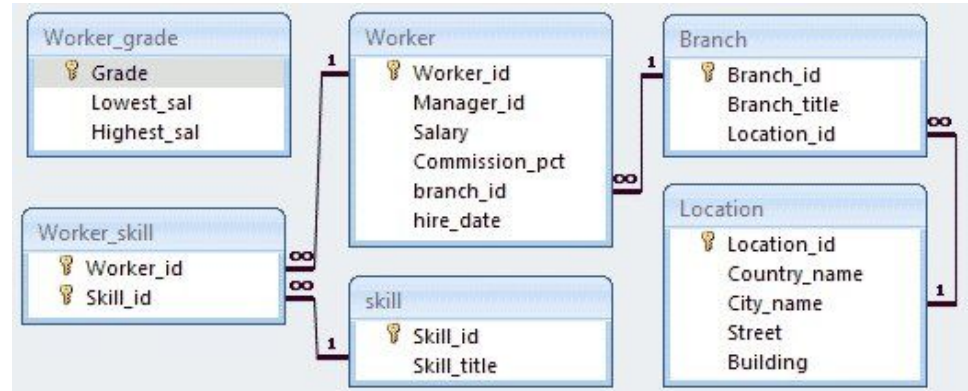


1. VIEW AS
2. view1 AS
3. VIEW view1
4. VIEW view1 AS SUBQUERY
5. VIEW view1 AS



Для создания представления, в которое должны попасть только имена сотрудников, работающих в отделе Research, используется запрос:

```
CREATE _____  
SELECT Worker_name FROM Worker w, Branch b  
WHERE w.Branch_id = b.Branch_id AND Branch_title LIKE  
'Research'
```

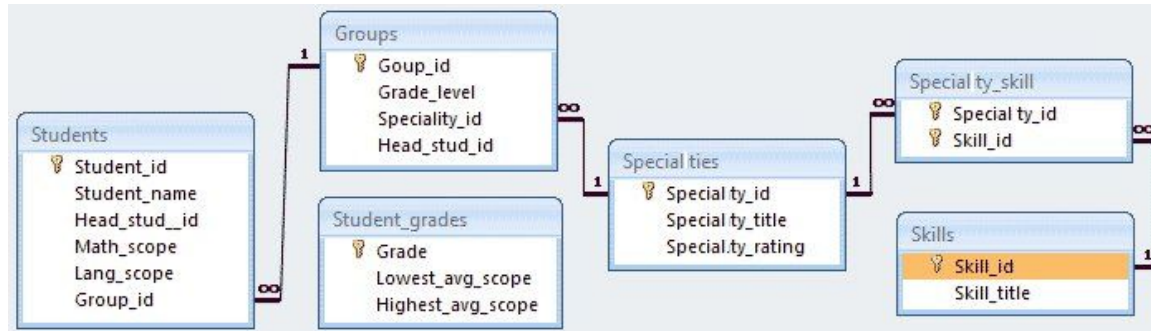


1. VIEW AS
2. view1 AS
3. VIEW view1
4. VIEW view1 AS SUBQUERY
5. **VIEW view1 AS**



Для создания представления, в которое должны попасть только имена студентов второго курса, используется запрос:

**CREATE VIEW view 1
AS.....**

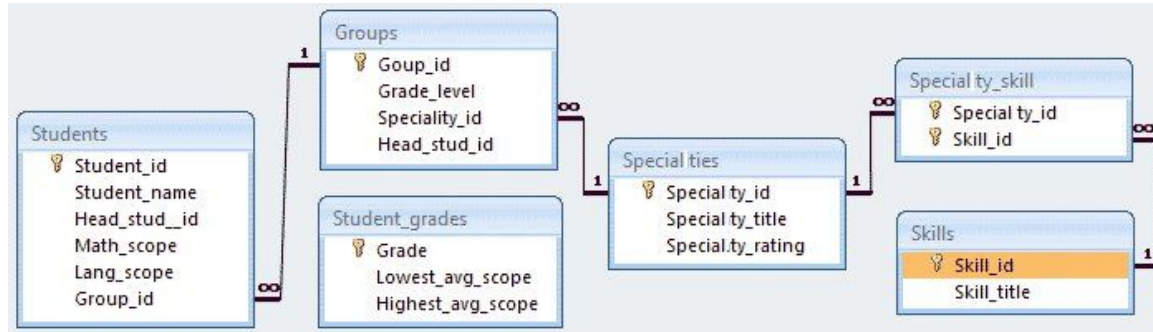


1. (SELECT Student_name FROM Students JOIN Groups ON Students.Group_id = Groups.Group_id) WITH CHECK OPTION Grade_level = 2
2. SELECT Student_name FROM Students, Groups WHERE Students.Group_id = Groups.Group_id AND Grade_level = 2
3. SELECT Student_name FROM Students JOIN Groups ON Students.Group_id = Groups.Group_id AND Grade_level = 2
4. WITH CHECK OPTION Grade_level=2 (SELECT Student_name FROM Students JOIN Groups ON Students.Group_id = Groups.Group_id)



Для создания представления, в которое должны попасть только имена студентов второго курса, используется запрос:

**CREATE VIEW view 1
AS.....**



1. (SELECT Student_name FROM Students JOIN Groups ON Students.Group_id = Groups.Group_id) WITH CHECK OPTION Grade_level = 2
2. SELECT Student_name FROM Students, Groups WHERE Students.Group_id = Groups.Group_id AND Grade_level = 2
3. SELECT Student_name FROM Students JOIN Groups ON Students.Group_id = Groups.Group_id AND Grade_level = 2
4. WITH CHECK OPTION Grade_level=2 (SELECT Student_name FROM Students JOIN Groups ON Students.Group_id = Groups.Group_id)



В чем заключается главное отличие оконных функций от функций агрегации с группировкой?

1. При использовании агрегирующих функций предложение GROUP BY сокращает количество строк в запросе с помощью их группировки, а при использовании оконных функций количество строк в запросе не уменьшается по сравнению с исходной таблицей.
2. Никакого различия нет
3. При использовании агрегирующих функций предложение GROUP BY НЕ сокращает количество строк в запросе с помощью их группировки, а при использовании оконных функций количество строк в запросе не уменьшается по сравнению с исходной таблицей.



В чем заключается главное отличие оконных функций от функций агрегации с группировкой?

1. При использовании агрегирующих функций предложение GROUP BY сокращает количество строк в запросе с помощью их группировки, а при использовании оконных функций количество строк в запросе не уменьшается по сравнению с исходной таблицей.
2. Никакого различия нет
3. При использовании агрегирующих функций предложение GROUP BY НЕ сокращает количество строк в запросе с помощью их группировки, а при использовании оконных функций количество строк в запросе не уменьшается по сравнению с исходной таблицей.



Оконные функции делятся на:

1. Агрегатные функции
2. Ранжирующие функции
3. Встроенные
4. Функции смещения



Оконные функции делятся на:

1. Агрегатные функции
2. Ранжирующие функции
3. Встроенные
4. Функции смещения

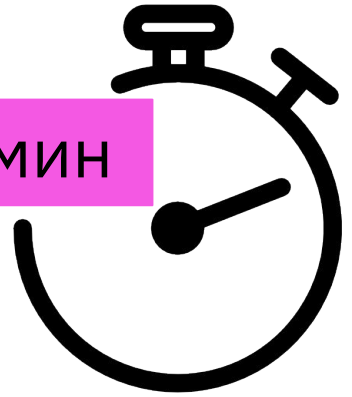


Оконные функции

Обычный запрос

Запрос с оконной функцией

20 мин



SELECT

Название функции (столбец для вычислений)

OVER (

PARTITION BY столбец для группировки

ORDER BY столбец для сортировки



Домашнее задание

Скрипт:

https://drive.google.com/file/d/1rkNr6GHSE9ulRs39BB4RC80aKMg1c7Jw/view?usp=share_link

1. Создайте представление, в которое попадут автомобили стоимостью до 25 000 долларов
2. Изменить в существующем представлении порог для стоимости: пусть цена будет до 30 000 долларов (используя оператор ALTER VIEW)
3. Создайте представление, в котором будут только автомобили марки “Шкода” и “Ауди”

```
mysql> SELECT * FROM Cars;
```

```
+----+-----+-----+
| Id | Name       | Cost  |
+----+-----+-----+
|  1 | Audi       | 52642 |
|  2 | Mercedes   | 57127 |
|  3 | Skoda      |  9000 |
|  4 | Volvo      | 29000 |
|  5 | Bentley    | 350000|
|  6 | Citroen    | 21000 |
|  7 | Hummer     | 41400 |
|  8 | Volkswagen | 21600 |
+----+-----+-----+
```



Рефлексия



Был урок полезен вам?




Узнали вы что-то новое?



Что было сложно?



Спасибо 
за внимание