



# Базы данных и SQL

Семинар 2.



Вопросы?

Вопросы?



Вопросы?



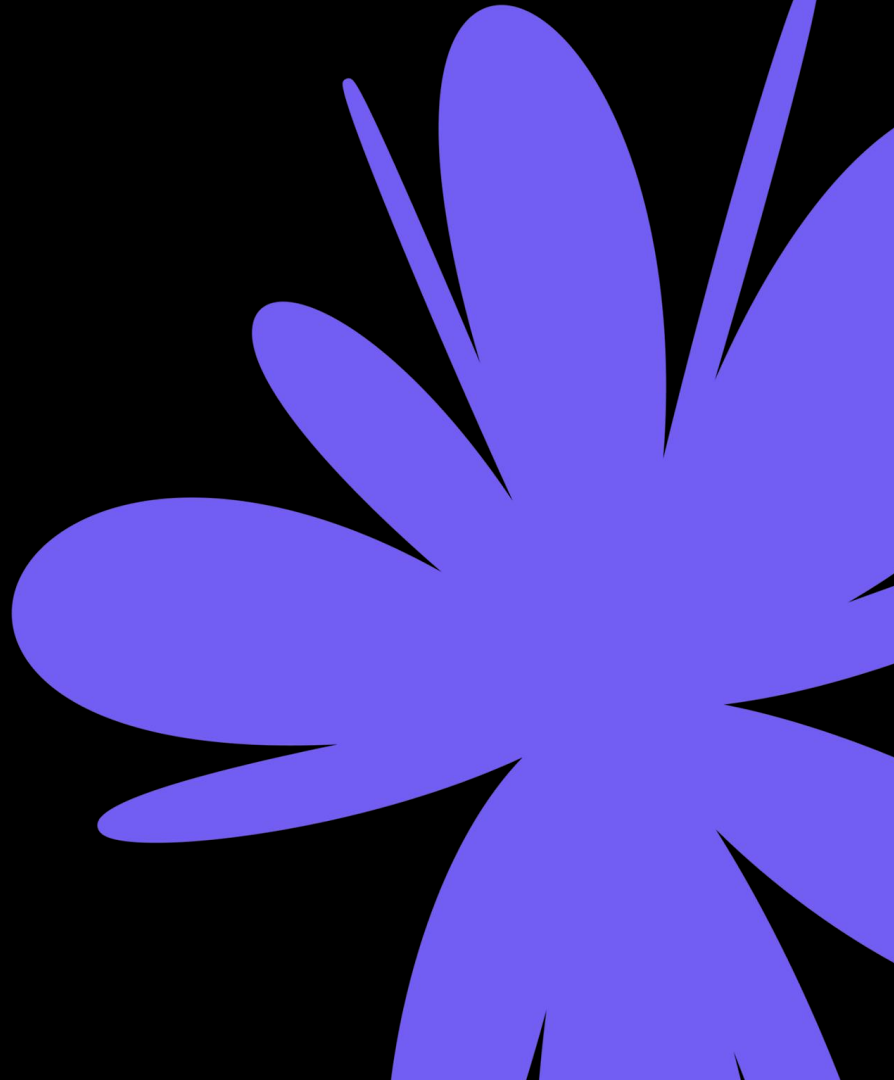
# План на сегодня:

- Викторина
- Работа с таблицами: создание и заполнение
- Манипуляции с таблицами
- Перерыв
- Использование оператора CASE и функции IF()
- Домашнее задание





# Викторина



## **Первичный ключ задается с помощью команды...**

1. PRIMARY KEY
2. FOREIGN KEY



## Первичный ключ задается с помощью команды...

1. PRIMARY KEY
2. FOREIGN KEY



## Выберите обозначения комментариев в MySQL:

1. - -
2. #
3. //
4. /\* Многострочный комментарий \*/



## Выберите обозначения комментариев в MySQL:

1. - -

2. #

3. //

4. /\* Многострочный комментарий \*/





# Оператор AND

1. объединяет два выражения, если оба этих выражения одновременно истинны
2. объединяет два выражения, если хотя бы одно выражение истинно
3. объединяет два выражения, если выражение в этой операции ложно, то общее условие истинно.



# Оператор AND

1. объединяет два выражения, если оба этих выражения одновременно истинны
2. объединяет два выражения, если хотя бы одно выражение истинно
3. объединяет два выражения, если выражение в этой операции ложно, то общее условие истинно.



# Какой оператор SQL нужно ставить, чтобы создать таблицу Persons?

```
1  ##
2  (
3  PersonID int,
4  LastName varchar(255),
5  FirstName varchar(255),
6  Address varchar(255),
7  City varchar(255)
8 );
```

1. CREATE TABLE Person
2. CREATE Persons
3. CREATE TABLE Persons
4. ADD TABLE Persons



# Какой оператор SQL нужно ставить, чтобы создать таблицу Persons?

```
1  ##
2  (
3  PersonID int,
4  LastName varchar(255),
5  FirstName varchar(255),
6  Address varchar(255),
7  City varchar(255)
8 );
```

1. CREATE TABLE Person
2. CREATE Persons
3. CREATE TABLE Persons
4. ADD TABLE Persons



# Что покажет следующий запрос:

```
1 select *  
2 from Orders  
3 where date between '2017-01-01' and '2017-12-31'
```

1. Все данные по заказам, совершенным за 2017 год, за исключением 01 января 2017 года
2. Все данные по заказам, совершенным за 2017 год, за исключением 31 декабря 2017 года
3. Все данные по заказам, совершенным за 2017 год
4. Ничего, запрос составлен неверно



## Что покажет следующий запрос:

```
1 select *  
2 from Orders  
3 where date between '2017-01-01' and '2017-12-31'
```

1. Все данные по заказам, совершенным за 2017 год, за исключением 01 января 2017 года
2. Все данные по заказам, совершенным за 2017 год, за исключением 31 декабря 2017 года
3. Все данные по заказам, совершенным за 2017 год
4. Ничего, запрос составлен неверно



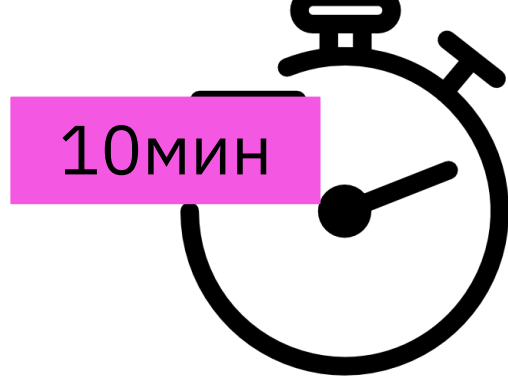
# Задача 1. Создать сущность с подборкой фильмов (movies).

В таблице имеются следующие атрибуты:

1. id -- уникальный идентификатор фильма,
2. title -- название фильма
3. title\_eng -- название фильма на английском языке
4. year\_movie -- год выхода
5. count\_min -- длительность фильма в минутах
6. storyline -- сюжетная линия, небольшое описание фильма

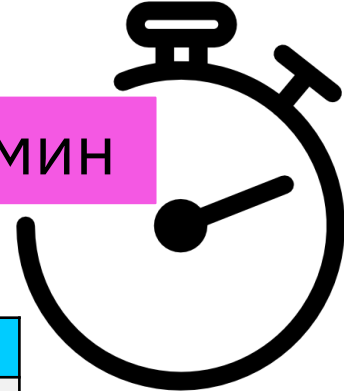
Все поля (кроме title\_eng, count\_min и storyline) обязательны для заполнения.

Поле id : первичный ключ, который заполняется автоматически.



## Задача 2. Заполните табличку тестовыми данными, используя оператор INSERT INTO. Пример:

10 мин



| id | title                            | title_eng          | year_movie | count_min | storyline   |
|----|----------------------------------|--------------------|------------|-----------|---|
| 1  | Игры разума                      | A Beautiful Mind   | 2001       | 135       | От всемирной известности до греховных глубин — все это познал на своей шкуре Джон Форбс Нэш-младший. Математический гений, он на заре своей карьеры сделал титаническую работу в области теории игр, которая перевернула этот раздел математики и практически принесла ему международную известность. Однако буквально в то же время заносчивый и пользующийся успехом у женщин Нэш получает удар судьбы, который переворачивает уже его собственную жизнь. |
| 2  | Форрест Гамп                     | Forrest Gump       | 1994       | 142       | Сидя на автобусной остановке, Форрест Гамп — не очень умный, но добрый и открытый парень — рассказывает случайным встречным историю своей необыкновенной жизни. С самого малолетства парень страдал от заболевания ног, соседские мальчишки дразнили его, но в один прекрасный день Форрест открыл в себе невероятные способности к бегу. Подруга детства Джинни всегда его поддерживала и защищала, но вскоре дороги их разошлись.                         |
| 3  | Иван Васильевич меняет профессию |                    | 1998       | 128       | Инженер-изобретатель Тимофеев сконструировал машину времени, которая соединила его квартиру с далеким шестнадцатым веком - точнее, с палатами государя Ивана Грозного. Туда-то и попадают тезка царя пенсионер-общественник Иван Васильевич Бунша и квартирный вор Жорж Милославский. На их место в двадцатом веке «переселяется» великий государь. Поломка машины приводит ко множеству неожиданных и забавных событий...                                  |
| 4  | Назад в будущее                  | Back to the Future | 1985       | 116       | Подросток Марти с помощью машины времени, сооруженной его другом-профессором доком Брауном, попадает из 80-х в далекие 50-е. Там он встречается со своими будущими родителями, ещё подростками, и другом-профессором, совсем молодым.   |
| 5  | Криминальное чтиво               | Pulp Fiction       | 1994       | 154       |   |





# Операции с таблицами. Задачи

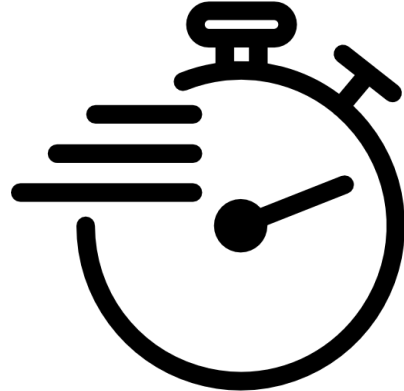
10 мин



1. Переименовать сущность **movies** в **cinema**.
2. Добавить сущности **cinema** новый атрибут **status\_active** и атрибут **genre\_id** после атрибута **title\_eng**.
3. Удалить атрибут **status\_active** сущности **cinema**.
4. Удалить сущность **actors** из базы данных
5. Добавить внешний ключ на атрибут **genre\_id** сущности **cinema** и направить его на атрибут **id** сущности **genres**.
6. Очистить сущность **genres** от данных и обнулить автоинкрементное поле.



# Операции с таблицами. Решения



1. Переименовать сущность **movies** в **cinema**:

```
RENAME TABLE movies TO cinema;
```

2. Добавить сущности **cinema** новый атрибут **status\_active** (тип **BIT**) и атрибут **genre\_id** после атрибута **title\_eng**:

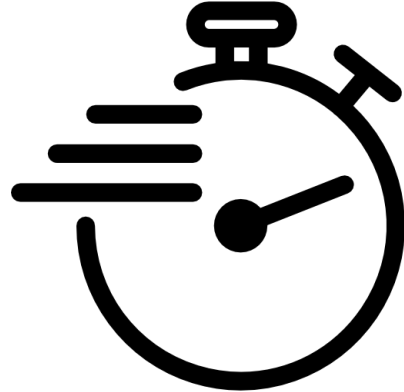
```
ALTER TABLE cinema  
ADD COLUMN active BIT DEFAULT b'1',  
ADD genre_id BIGINT UNSIGNED AFTER title_eng;
```

3. Удалить атрибут **status\_active** сущности **cinema**:

```
ALTER TABLE cinema  
DROP COLUMN status_active;
```



# Операции с таблицами. Решения



4. Удалить сущность **actors** из базы данных:

**DROP TABLE** actors;

5. Добавить внешний ключ на атрибут **genre\_id** сущности **cinema** и направить его на атрибут **id** сущности **genres**:

**ALTER TABLE** cinema

**ADD FOREIGN KEY**(genre\_id) **REFERENCES** genres(id);

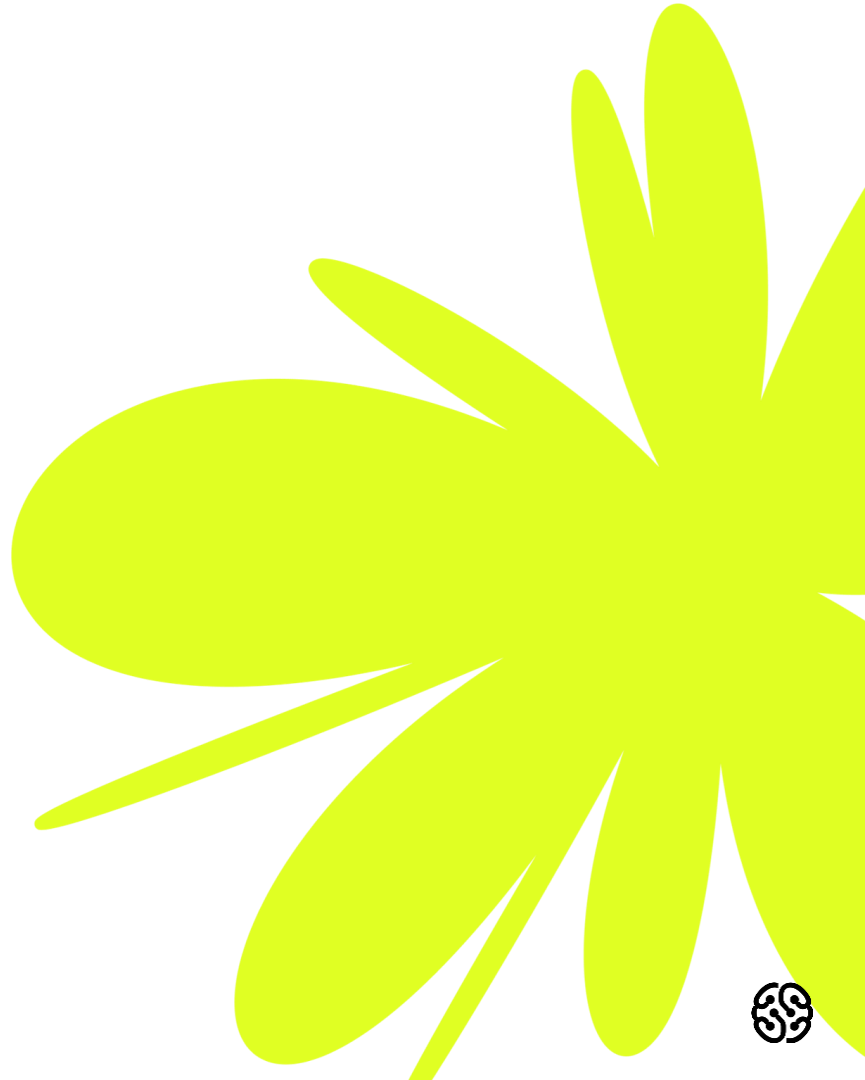
6. Очистить сущность **genres** от данных и обнулить автоинкрементное поле:

**TRUNCATE TABLE** genres ;



**Ваши вопросы?**

**Перерыв**



**Задача 3.** Выведите id, название фильма и категорию фильма, согласно следующего перечня:

Д- Детская, П – Подростковая,  
В – Взрослая, Не указана

10 мин



| Номер | Название фильма                  | Категория    |
|-------|----------------------------------|--------------|
| 1     | Игры разума                      | Подростковая |
| 2     | Форрест Гамп                     | Не указана   |
| 3     | Иван Васильевич меняет профессию | Не указана   |
| 4     | Назад в будущее                  | Детская      |
| 5     | Криминальное чтиво               | Взрослая     |



## Задача 3. Решение.

```
SELECT
    id,
    title,
    CASE age_category
        WHEN 'Д' THEN 'Детская'
        WHEN 'П' THEN 'Подростковая'
        WHEN 'В' THEN 'Взрослая'
        ELSE 'Не указана'
    END AS 'Категория'
FROM cinema;
```



## Задача 4. Выведите id, название фильма, продолжительность, тип в зависимости от продолжительности (с использованием CASE).

До 50 минут - Короткометражный фильм

От 50 минут до 100 минут - Среднеметражный фильм

Более 100 минут - Полнометражный фильм

Иначе - Не определено

| Номер | Название фильма                  | Продолжительность | Тип                    |
|-------|----------------------------------|-------------------|------------------------|
| 1     | Игры разума                      | 135               | Полнометражный фильм   |
| 2     | Форрест Гамп                     | 88                | Среднеметражный фильм  |
| 3     | Иван Васильевич меняет профессию |                   | Не определено          |
| 4     | Назад в будущее                  | 34                | Короткометражный фильм |
| 5     | Криминальное чтиво               | 154               | Полнометражный фильм   |

10 мин



## Задача 4. Решение.

**SELECT**

id **AS** 'Номер фильма',  
title **AS** 'Название фильма',  
count\_min **AS** 'Продолжительность',

**CASE**

**WHEN** count\_min < 50 **THEN** 'Короткометражный фильм'

**WHEN** count\_min **between** 50 **AND** 100

**THEN** 'Среднеметражный фильм'

**WHEN** count\_min > 100 **THEN** 'Полнометражный фильм'

**ELSE** 'Не определено'

**END AS** 'Тип'

**FROM** cinema;





## Задача 5. Выведите id, название фильма, продолжительность, тип в зависимости от продолжительности (с использованием IF).

До 50 минут - Короткометражный фильм

От 50 минут до 100 минут - Среднеметражный фильм

Более 100 минут - Полнометражный фильм

Иначе - Не определено

| Номер | Название фильма                  | Продолжительность | Тип                    |
|-------|----------------------------------|-------------------|------------------------|
| 1     | Игры разума                      | 135               | Полнометражный фильм   |
| 2     | Форрест Гамп                     | 88                | Среднеметражный фильм  |
| 3     | Иван Васильевич меняет профессию |                   | Не определено          |
| 4     | Назад в будущее                  | 34                | Короткометражный фильм |
| 5     | Криминальное чтиво               | 154               | Полнометражный фильм   |

10 мин



## Задача 5. Решение.

**SELECT**

id **AS** 'Номер фильма',

title **AS** 'Название фильма',

count\_min **AS** 'Продолжительность',

**IF** (count\_min < 50, 'Короткометражный фильм',

**IF** (count\_min **between** 50 **AND** 100, 'Среднеметражный фильм',

**IF** (count\_min > 100, 'Полнометражный фильм', 'Не определено')

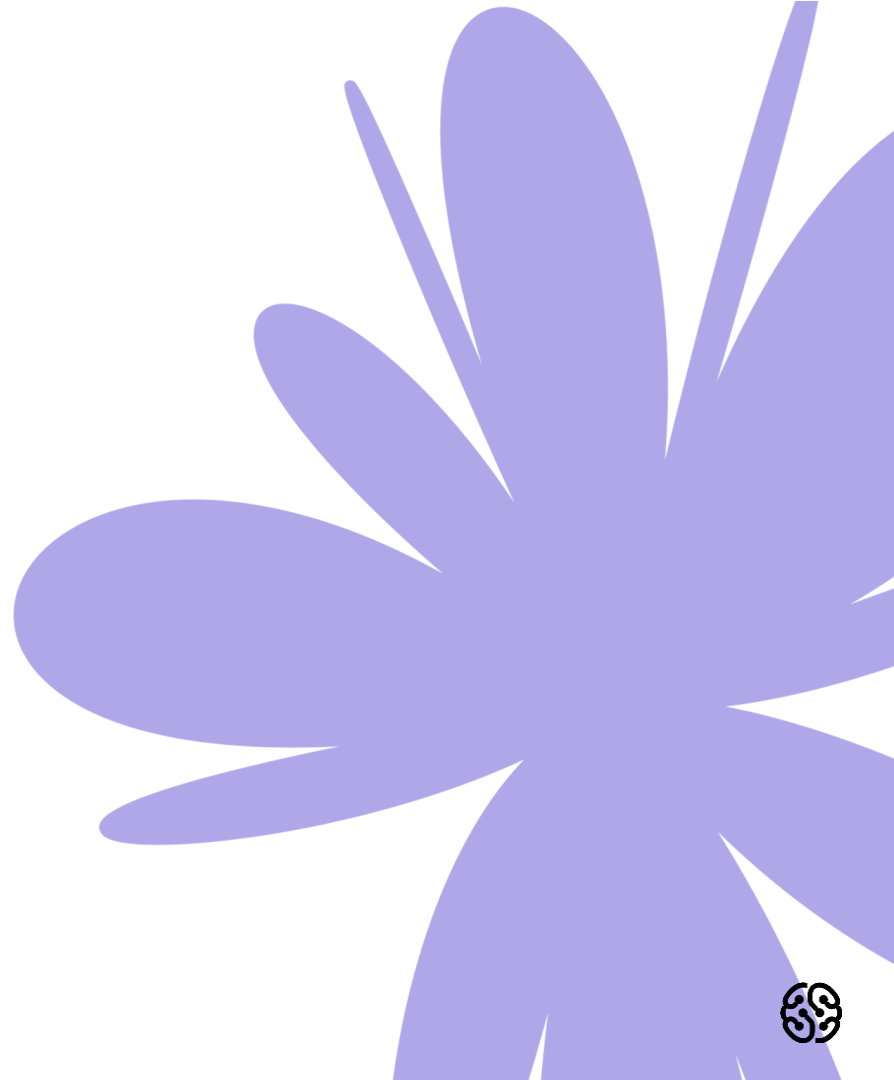
    )

) **AS** 'Тип'

**FROM** cinema;



**Ваши вопросы?**



## Домашнее задание

1. Используя операторы языка SQL, создайте таблицу “sales”. Заполните ее данными. Справа располагается рисунок к первому заданию.

| id | order_date | count_product |
|----|------------|---------------|
| 1  | 2022-01-01 | 156           |
| 2  | 2022-01-02 | 180           |
| 3  | 2022-01-03 | 21            |
| 4  | 2022-01-04 | 124           |
| 5  | 2022-01-05 | 341           |

2. Для данных таблицы “sales” укажите тип заказа в зависимости от кол-ва :  
меньше 100 - Маленький заказ  
от 100 до 300 - Средний заказ  
больше 300 - Большой заказ

| id заказа | Тип заказа      |
|-----------|-----------------|
| 1         | Средний заказ   |
| 2         | Средний заказ   |
| 3         | Маленький заказ |
| 4         | Средний заказ   |
| 5         | Большой заказ   |



3. Создайте таблицу “orders”, заполните ее значениями

| id | employee_id | amount | order_status |
|----|-------------|--------|--------------|
| 1  | e03         | 15.00  | OPEN         |
| 2  | e01         | 25.50  | OPEN         |
| 3  | e05         | 100.70 | CLOSED       |
| 4  | e02         | 22.18  | OPEN         |
| 5  | e04         | 9.50   | CANCELLED    |

Выберите все заказы. В зависимости от поля order\_status выведите столбец full\_order\_status: OPEN – «Order is in open state» ; CLOSED - «Order is closed»; CANCELLED - «Order is cancelled»

4. Чем 0 отличается от NULL?

Напишите ответ в комментарии к домашнему заданию на платформе



# 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'.