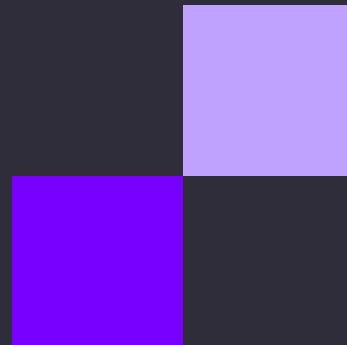




онлайн  
университет



ВЫСШАЯ ШКОЛА  
ЭКОНОМИКИ



# SQL для начинающих

Занятие 2



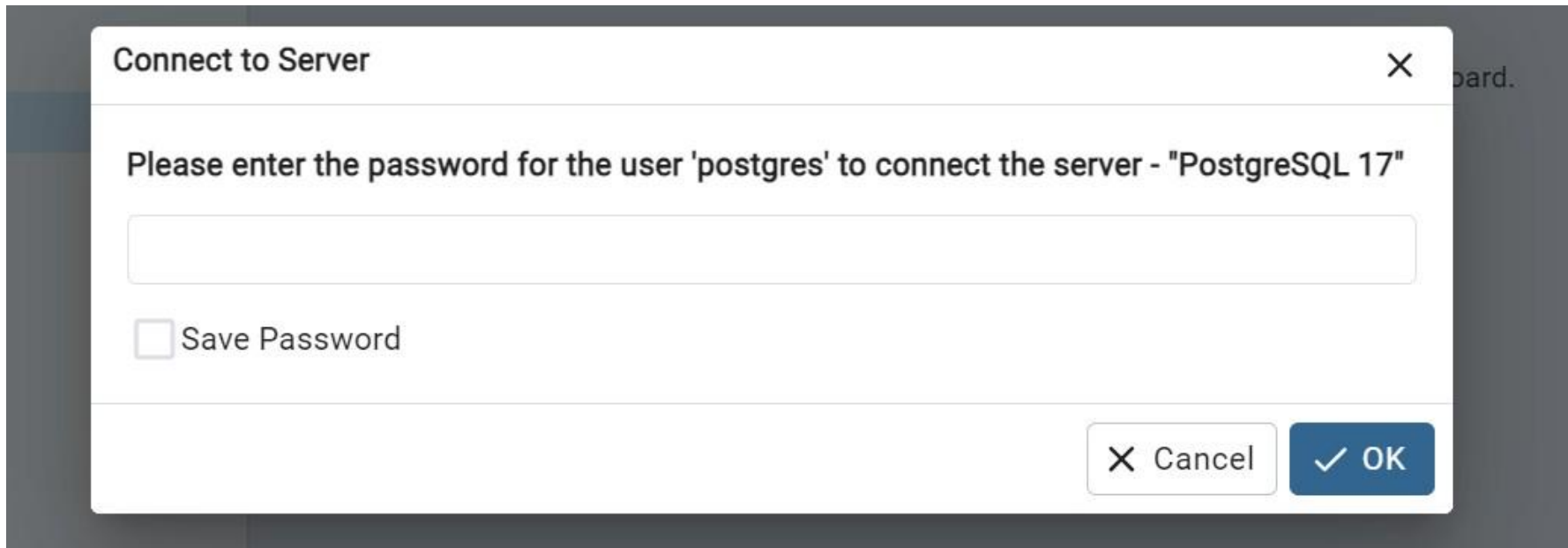
# Где работаем?

Для упрощения администрирования на сервере postgresql в базовый комплект установки входит такой инструмент как **pgAdmin**. Он представляет графический клиент для работы с сервером, через который мы в удобном виде можем создавать, удалять, изменять базы данных и управлять ими.



# Открываем приложение

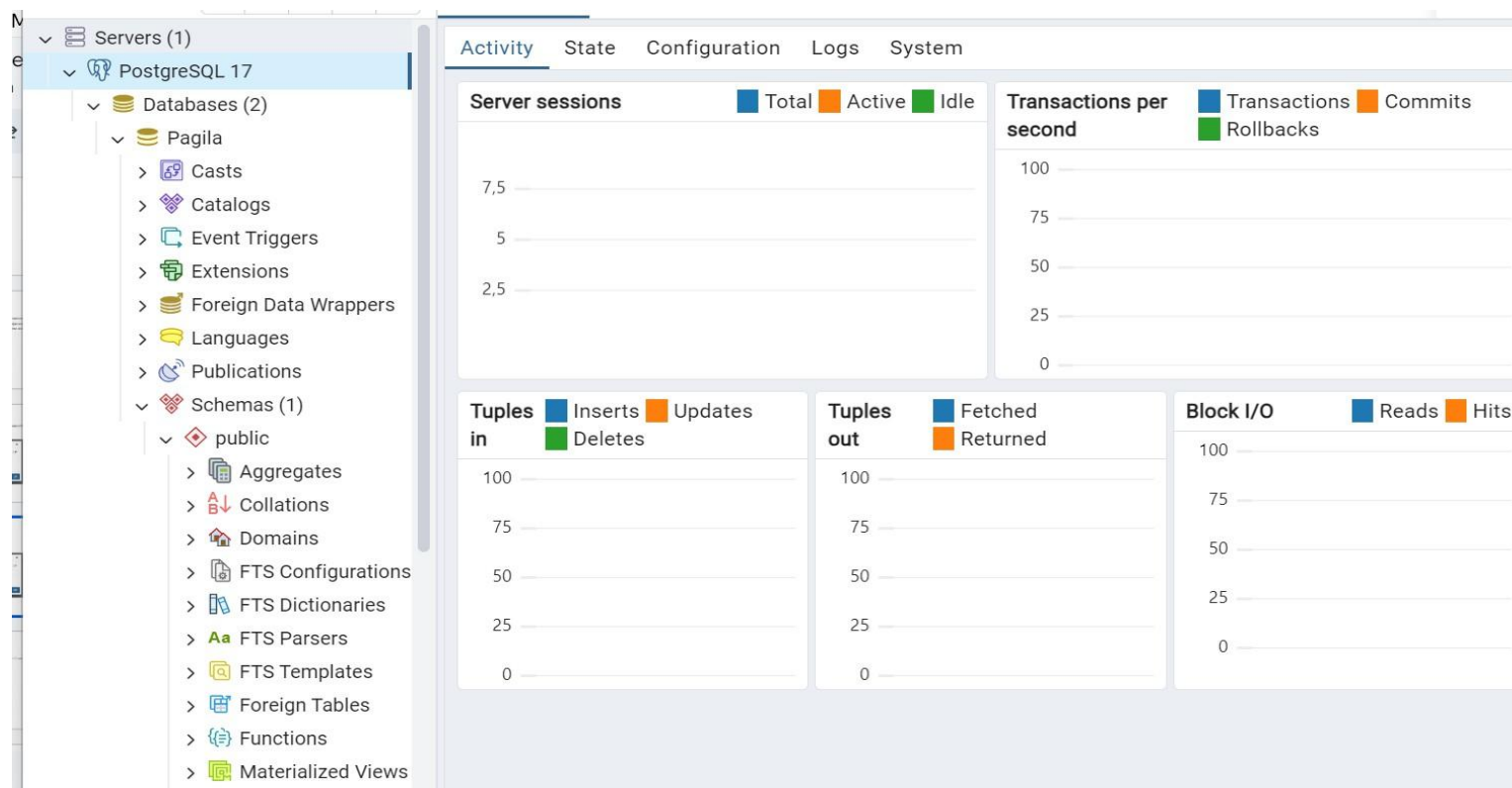
При подключении запросит пароль,  
который вы задавали при установке:



The image shows a 'Connect to Server' dialog box with a close button (X) in the top right corner. The main text inside the dialog reads: 'Please enter the password for the user 'postgres' to connect the server - "PostgreSQL 17"'. Below this text is a large, empty rectangular input field for the password. Underneath the input field is a checkbox labeled 'Save Password'. At the bottom right of the dialog, there are two buttons: 'Cancel' with a close icon (X) and 'OK' with a checkmark icon (✓).

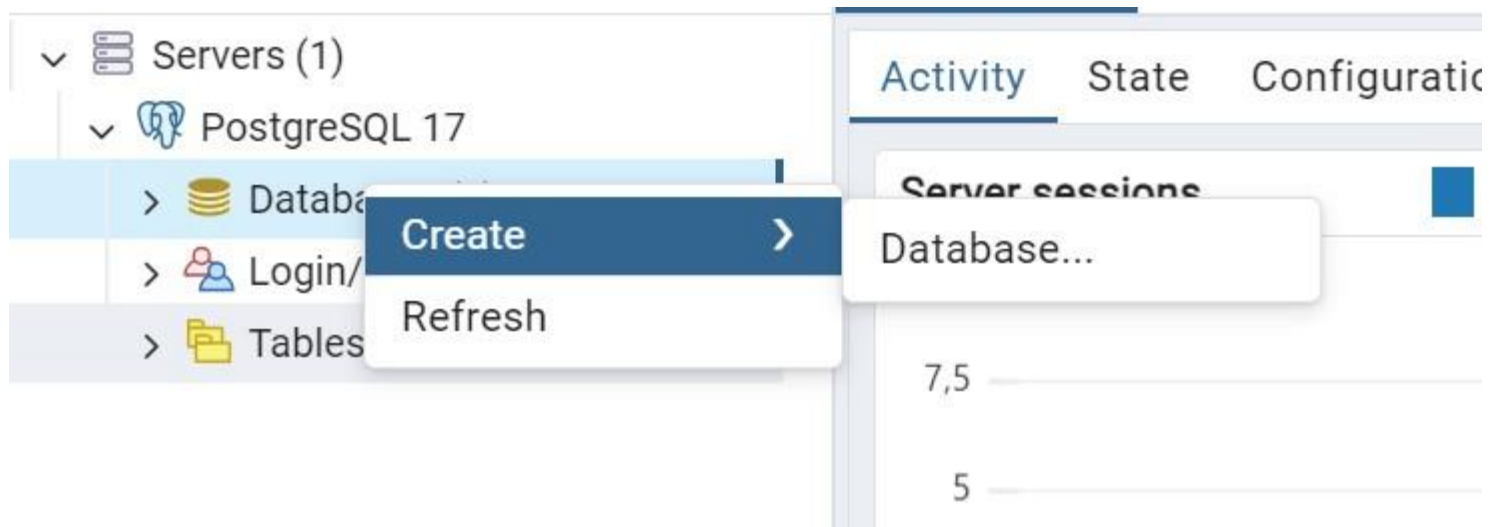
# Сервер

После успешного логина нам откроется  
содержимое сервера:

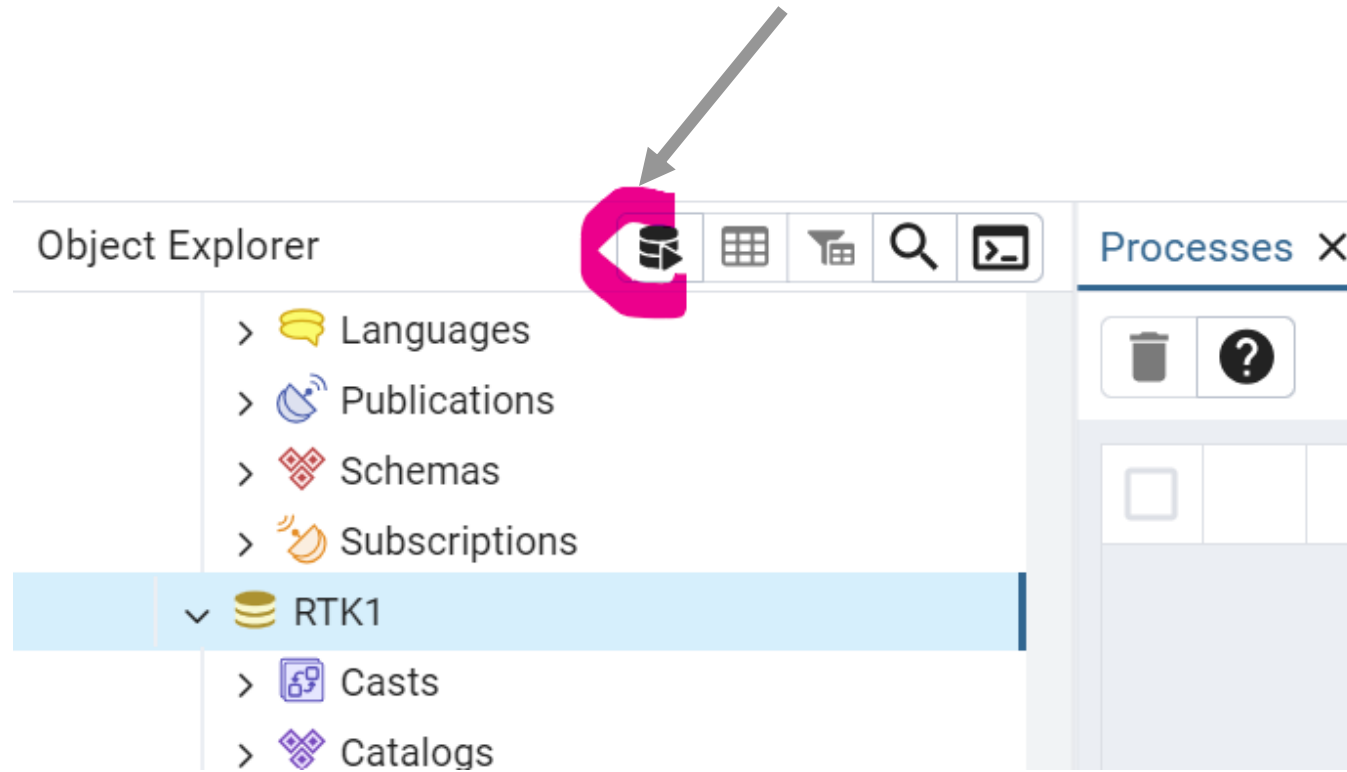


# Создание базы данных

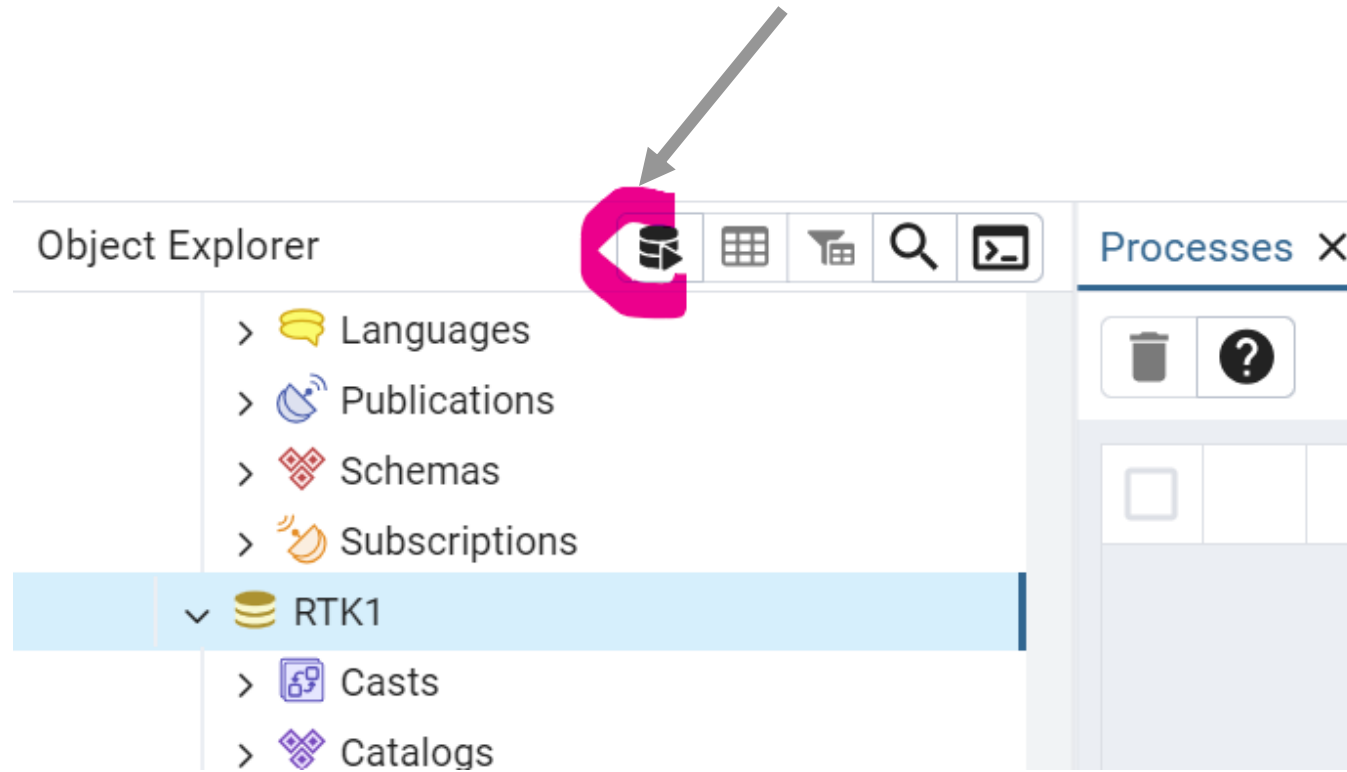
Теперь создадим свою базу данных. Для этого нажмем правой кнопкой мыши на узел **Databases**. И далее в контекстном меню выберем **Create->Database...**



# Создание БД с помощью запроса



# Создание БД с помощью запроса



# Описание структуры данных

## Таблица rooms

Хранит информацию о номерах отеля.

Поле	Тип данных	Описание
room_number	SERIAL PRIMARY KEY	Уникальный номер комнаты (Primary Key)
type_name	TEXT	Название типа номера
price_per_night	INTEGER	Цена за ночь в рублях
room_size_sqm	INTEGER	Площадь номера в квадратных метрах
max_occupancy	INTEGER	Максимальное количество гостей
has_minibar	BOOLEAN	Наличие минибара (True/False)
payment_option	TEXT	Вариант оплаты



# Описание структуры данных

## Таблица clients

Содержит информацию о клиентах отеля.

Поле	Тип данных	Описание
id	SERIAL PRIMARY KEY	Уникальный идентификатор клиента (Primary Key)
first_name	TEXT	Имя клиента
last_name	TEXT	Фамилия клиента
address	TEXT	Адрес клиента
email	TEXT	Адрес электронной почты клиента
phone_number	VARCHAR	Номер телефона клиента

# Описание структуры данных

## Таблица bookings

Содержит информацию о бронированиях номеров отеля.

Поле	Тип данных	Описание
booking_id	SERIAL PRIMARY KEY	Уникальный идентификатор бронирования (Primary Key)
renter_id	INTEGER REFERENCES clients(id)	Идентификатор клиента (внешний ключ)
room_number	INTEGER REFERENCES rooms(room_number)	Номер комнаты (внешний ключ)
check_in_date	TIMESTAMP	Дата заезда
check_out_date	TIMESTAMP	Дата заезда

# Описание структуры данных

## Таблица payments

Хранит информацию об оплатах за бронирования.

Поле	Тип данных	Описание
payment_id	SERIAL PRIMARY KEY	Уникальный идентификатор платежа (Primary Key)
booking_id	INTEGER REFERENCES bookings(booking_id)	Идентификатор бронирования (внешний ключ)
amount_paid	DECIMAL	Сумма оплаты
payment_date	DATE	Дата платежа
payment_method	VARCHAR	Метод оплаты

# Описание структуры данных

## Таблица ratings

Хранит информацию об оценках, данной клиентами за проживание в номерах отеля.

Поле	Тип данных	Описание
rating_id	SERIAL PRIMARY KEY	Уникальный идентификатор оценки (Primary Key)
booking_id	INTEGER REFERENCES bookings(booking_id)	Идентификатор бронирования (внешний ключ)
rating_value	INTEGER	Значение оценки (от 1 до 5 звёзд)

# SELECT

Ключевое слово **SELECT** лежит в основе любого SQL- запроса к БД, в результате которого мы хотим получить какие-то данные. С помощью запроса **SELECT** можно выполнять выборку данных из таблицы либо производить вычисления.

**SELECT 2 + 3**



	?column? integer
1	5

# SELECT

Ключевое слово **SELECT** лежит в основе любого SQL- запроса к БД, в результате которого мы хотим получить какие-то данные. С помощью запроса **SELECT** можно выполнять выборку данных из таблицы либо производить вычисления.

Для численных типов данных в SQL используются традиционные арифметические операторы:

```
SELECT 2 + 3
```

## АРИФМЕТИЧЕСКИЕ ОПЕРАТОРЫ

+ Сложение

- Вычитание

/ Деление

\* Умножение

% Остаток от деления

# SELECT + FROM

**SELECT** (названия столбцов через запятую) **FROM** (название таблицы)

Если же нужно данные сразу из всех столбцов, то вместо названий столбцов мы просто введем знак \*:

```
SELECT * FROM <имя_таблицы>
```

# SELECT + FROM: практика

Получим данные обо всех номерах нашей гостиницы:

```
SELECT *  
FROM rooms
```



# Немного о правописании

1. Операторы следует писать в верхнем регистре, а ключевые слова в нижнем.
2. Каждое ключевое слово пишется с новой строки.

<https://www.sqlstyle.guide/ru/>

# SELECT + FROM

Чтобы упорядочить поля или выбрать только часть из них, их следует перечислить через запятую:

```
SELECT room_number, type_name, has_minibar  
FROM rooms
```

# SELECT + FROM

Можно добавлять вычисления:

```
SELECT payment_id, amount_paid/80  
FROM payments
```

# SELECT + FROM

Выборка может содержать дубликаты строк

```
SELECT type_name  
FROM rooms
```


	type_name text
1	Люкс
2	Стандартный
3	Стандартный
4	Стандартный

# SELECT + FROM

Выборка может содержать дубликаты строк

```
SELECT DISTINCT type_name  
FROM rooms
```

В таком случае, если мы хотим просто вывести все варианты рейтинга - мы можем воспользоваться оператором DISTINCT, который убирает дубликаты из результирующей выборки

	type_name text	
1	Президентский	
2	Люкс	
3	Стандартный	

# Псевдоним (alias)

По умолчанию при выводе столбцы называются так же, как в исходной таблице. Для переименования нужно в SELECT после названия столбца из таблицы написать as:

```
SELECT room_number AS r  
FROM rooms
```



	r 
1	1
2	2
3	3
4	4

# ORDER BY

Отсортировать выборку можно, добавив предложение ORDER BY в текст запроса.

Давайте отсортируем клиентов по фамилии в алфавитном порядке:

```
SELECT last_name, first_name, email  
FROM clients  
ORDER BY last_name
```

	last_name text		first_name text		email text
1	Абрамов		Моисей		ogromov@ex
2	Абрамов		Ираклий		sobolevanan
3	Абрамов		Иннокентий		danilovaagaf
4	Абрамов		Станислав		ribakovapolir

# ORDER BY

В обратном порядке также можно сортировать. Для этого используем ключевое слово DESC (от слов descending order)

```
SELECT last_name, first_name, email  
FROM clients  
ORDER BY last_name DESC, first_name
```

	last_name text	first_name text	email text
1	Якушева	Агата	djas
2	Якушева	Алла	mak
3	Якушева	Анастасия	erst
4	Якушева	Дарья	filar



# ORDER BY

В обратном порядке также можно сортировать. Для этого используем ключевое слово DESC (от слов descending order)

```
SELECT last_name, first_name, email  
FROM clients  
ORDER BY last_name DESC, first_name
```

	last_name text	first_name text	email text
1	Якушева	Агата	djas
2	Якушева	Алла	mak
3	Якушева	Анастасия	erst
4	Якушева	Дарья	filar

Порядок сортировки (DESC или ASC) применяется только для той колонки, для которой он указан

# ORDER BY

Для упрощения работы с ORDER BY можно использовать не названия столбцов, а номер столбца в выгружаемых данных. Например:

```
SELECT last_name, first_name, email  
FROM clients  
ORDER BY 2
```

	last_name text	first_name text
1	Волков	Август
2	Кузьмин	Август
3	Овчинников	Август
4	Зайцев	Август

# LIMIT и OFFSET

Иногда нам нужны не все данные по каким-то условиям, а лишь их часть, например, когда нужно посмотреть, что и в каком формате содержится в таблице. С этим могут помочь ключевые слова **LIMIT** и **OFFSET**.

Допустим, нам нужно вывести 5 строк таблицы. Нам поможет слово **LIMIT**.

```
SELECT last_name, first_name, email  
FROM clients  
ORDER BY 2  
LIMIT 5
```

# LIMIT и OFFSET

Можем посмотреть по дешевизне номера со 2 по 6

```
SELECT *  
FROM rooms  
ORDER BY price_per_night  
LIMIT 5  
OFFSET 1
```

# LIMIT и OFFSET

Важно, что их порядок не важен!

```
SELECT *  
FROM rooms  
ORDER BY price_per_night  
OFFSET 1  
LIMIT 5
```

# Задача 1

Выведите клиентов по алфавиту по фамилии в порядке убывания. Выведите только 10 человек.

# Задача 1

```
SELECT *  
FROM clients  
ORDER BY last_name DESC  
LIMIT 10
```

## Задача 2

Выведите 5 самых ранних бронирований по  
check\_in\_date



## Задача 2

```
SELECT *  
FROM bookings  
ORDER BY check_in_date  
LIMIT 10
```

## Задача 3

Выведите все уникальные типы оплат,  
отсортированные по алфавиту

## Задача 3

```
SELECT DISTINCT payment_method  
FROM payments  
ORDER BY payment_method
```

## Задача 4

Выведите только номера комнат и их цену за ночь.  
Переименуйте столбец `price_per_night` в `price`.

## Задача 4

Выведите только номера комнат и их цену за ночь.  
Переименуйте столбец price\_per\_night в price.

```
SELECT room_number, price_per_night AS price  
FROM rooms;
```

## Задача 5

Выведите все типы номеров вместе с ценой за ночь и признаком мини-бара.

Отсортируйте по типу номера, затем по цене по возрастанию.

## Задача 5

Выведите все типы номеров вместе с ценой за ночь и признаком мини-бара.

Отсортируйте по типу номера, затем по цене по возрастанию.

```
SELECT type_name, price_per_night, has_minibar  
FROM rooms  
ORDER BY type_name, price_per_night;
```

**Спасибо за внимание!**