

Урок 1. Инструменты для работы с базами данных



Давайте знакомиться!

Меркушов Михаил

Ведущий разработчик баз данных

Меркушов Михаил Сергеевич - более 3 лет занимаюсь разработкой баз данных и созданием программ. Оканчиваю Волгоградский технический университет по специальности "Информатика и вычислительная техника", профиль "Автоматизированное проектирование киберфизических систем". В данный момент занимаюсь разработкой приложения для людей с ограниченными возможностями для Бельгийского университета (компания mail инвестиционная технологическая корпорация)

Что будет на уроке сегодня

- Базы данных, СУБД кто есть кто
- Истории возникновения СУБД и языка SQL
- Основные операторы SQL
- Компоненты СУБД
- Клиентские приложения для работы с СУБД
- Работа с БД с помощью дизайнера
- Запросы выборки данных

База данных VS СУБД

База данных - некоторый набор постоянно хранимых данных

Решаемые задачи:

- Хранение данных
- Получение данных
- Обработка

СУБД (DBMS) — система управления базами данных

Наиболее используемые СУБД

398 systems in ranking, June 2022

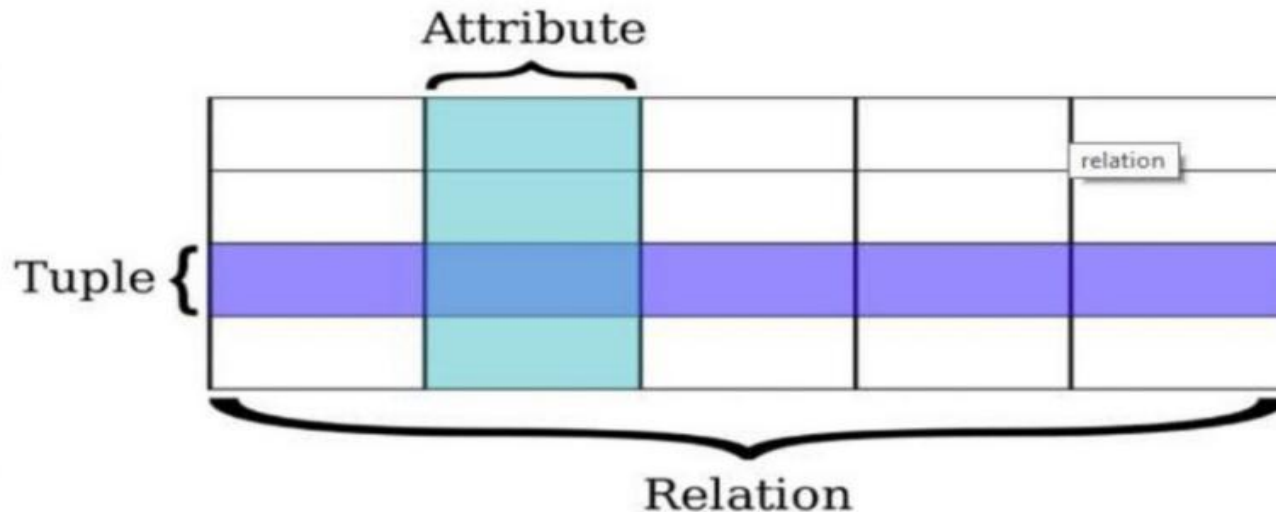
Rank			DBMS	Database Model	Score		
Jun 2022	May 2022	Jun 2021			Jun 2022	May 2022	Jun 2021
1.	1.	1.	Oracle +	Relational, Multi-model ⓘ	1287.74	+24.92	+16.80
2.	2.	2.	MySQL +	Relational, Multi-model ⓘ	1189.21	-12.89	-38.65
3.	3.	3.	Microsoft SQL Server +	Relational, Multi-model ⓘ	933.83	-7.37	-57.25
4.	4.	4.	PostgreSQL +	Relational, Multi-model ⓘ	620.84	+5.55	+52.32
5.	5.	5.	MongoDB +	Document, Multi-model ⓘ	480.73	+2.49	-7.49
6.	6.	↑ 7.	Redis +	Key-value, Multi-model ⓘ	175.31	-3.71	+10.06
7.	7.	↓ 6.	IBM Db2	Relational, Multi-model ⓘ	159.19	-1.14	-7.85
8.	8.	8.	Elasticsearch	Search engine, Multi-model ⓘ	156.00	-1.70	+1.29
9.	9.	↑ 10.	Microsoft Access	Relational	141.82	-1.62	+26.88
10.	10.	↓ 9.	SQLite +	Relational	135.44	+0.70	+4.90
11.	11.	11.	Cassandra +	Wide column	115.45	-2.56	+1.34
12.	12.	12.	MariaDB +	Relational, Multi-model ⓘ	111.58	+0.45	+14.79
13.	↑ 14.	↑ 26.	Snowflake +	Relational	96.42	+2.91	+61.67
14.	↓ 13.	↓ 13.	Splunk	Search engine	95.56	-0.79	+5.30
15.	15.	15.	Microsoft Azure SQL Database	Relational, Multi-model ⓘ	86.01	+0.68	+11.22
16.	16.	16.	Amazon DynamoDB +	Multi-model ⓘ	83.88	-0.58	+10.12
17.	17.	↓ 14.	Hive +	Relational	81.58	-0.03	+1.89
18.	18.	↓ 17.	Teradata +	Relational, Multi-model ⓘ	70.41	+2.02	+1.07
19.	19.	↓ 18.	Neo4j +	Graph	59.53	-0.61	+3.78
20.	20.	20.	Solr	Search engine, Multi-model ⓘ	56.61	-0.64	+4.52

Реляционные базы данных (от англ. Relation – связь) — базы данных, в которых данные распределены по отдельным, но связанным между собой таблицам.

Кортеж (tuple) — это множество пар {имя атрибута, значение}.

(например {Фамилия, Петров}, {Телефон, +7921-123-56-69}) — фактически это строка таблицы, где имена атрибутов — это столбцы таблицы.

Отношение (relation) - это множество кортежей, соответствующих одной схеме. (На прикладном уровне это соотносится с таблицей.)



Небольшой пример на кортежи:

В таблице “Сотрудники” имеется кортеж данных:

{ [Идентиф_код], '2931123455' }

{ [Фамилия_и_инициалы], 'Петренко П.П.' }

Идентиф_код	Фамилия_и_инициалы
2931123455	Петренко П.П.

Первичный ключ (Primary key) – поле(или набор полей) позволяющее однозначно идентифицировать запись в БД. Если ключ состоит из нескольких полей его называют **составным**.

Суррогатный ключ - автоматически сгенерированное поле, никак не связанное с информационным содержанием записи.

Естественный ключ — ключ состоящий из информационных полей таблицы.

<i>Студенты</i>				
Фамилия	Имя	Год рождения	Паспорт серия	Паспорт номер
Иванов	Петр	1992	0111	121245
<u>Пупкин</u>	Федор	1995	1102	457879
Смирнов	Иван	1986	0013	787952
Смирнов	Степан	1997	0013	784593
Петрова	Ирина	1996	1802	596485

Теперь ваша очередь!

Вопрос

какие ключи можно выделить в этой таблице?
подумайте 1 минуту и разберем варианты

<i>Студенты</i>				
Фамилия	Имя	Год рождения	Паспорт серия	Паспорт номер
Иванов	Петр	1992	0111	121245
<u>Пупкин</u>	Федор	1995	1102	457879
Смирнов	Иван	1986	0013	787952
Смирнов	Степан	1997	0013	784593
Петрова	Ирина	1996	1802	596485

Потенциальный ключ – подмножество атрибутов таблицы, удовлетворяющее требованиям уникальности и минимальности.

Естественный ключ (Паспорт серия + номер), недостатки:

- Может изменяться
- Может отсутствовать
- Широкий — два текстовых поля

Студенты				
Фамилия	Имя	Год рождения	Паспорт серия	Паспорт номер
Иванов	Петр	1992	0111	121245
Пупкин	Федор	1995	1102	457879
Смирнов	Иван	1986	0013	787952
Смирнов	Степан	1997	0013	784593
Петрова	Ирина	1996	1802	596485

Связи между таблицами — использование ключей

Студенты

Фамилия	Имя	Год рождения	Паспорт серия	Паспорт номер
Иванов	Петр	1992	0111	121245
Пупкин	Федор	1995	1102	457879
Смирнов	Иван	1986	0013	787952
Смирнов	Степан	1997	0013	784593
Петрова	Ирина	1996	1802	596485

Телефоны студентов

Студент	Паспорт серия	Студент	Паспорт номер	Телефон	Использование
	0111		121245	89211285696	основной
	1102		457879	89651238956	основной
	0013		787952	65428967	основной
	0013		787952	49626548596	дополнительный

Суррогатный ключ

Студенты

Студент id	Фамилия	Имя	Год рождения	Паспорт серия	Паспорт номер
1	Иванов	Петр	1992	0111	121245
2	Пупкин	Федор	1995	1102	457879
3	Смирнов	Иван	1986	0013	787952
4	Смирнов	Степан	1997	0013	784593
5	Петрова	Ирина	1996	1802	596485

Язык SQL, история

1970-е — компанией IBM был разработан прототип СУБД System R, язык запросов (Structured Query Language) SQL

1986 — первый международный стандарт, SQL-86 ещё называют ANSI/ISO

1992 — стандарт SQL-92 (SQL2)

1999 — стандарт SQL3:1999 (SQL3)

2003 — стандарт SQL 2003

2006 — стандарт SQL 2003

2008 — стандарт SQL 2008

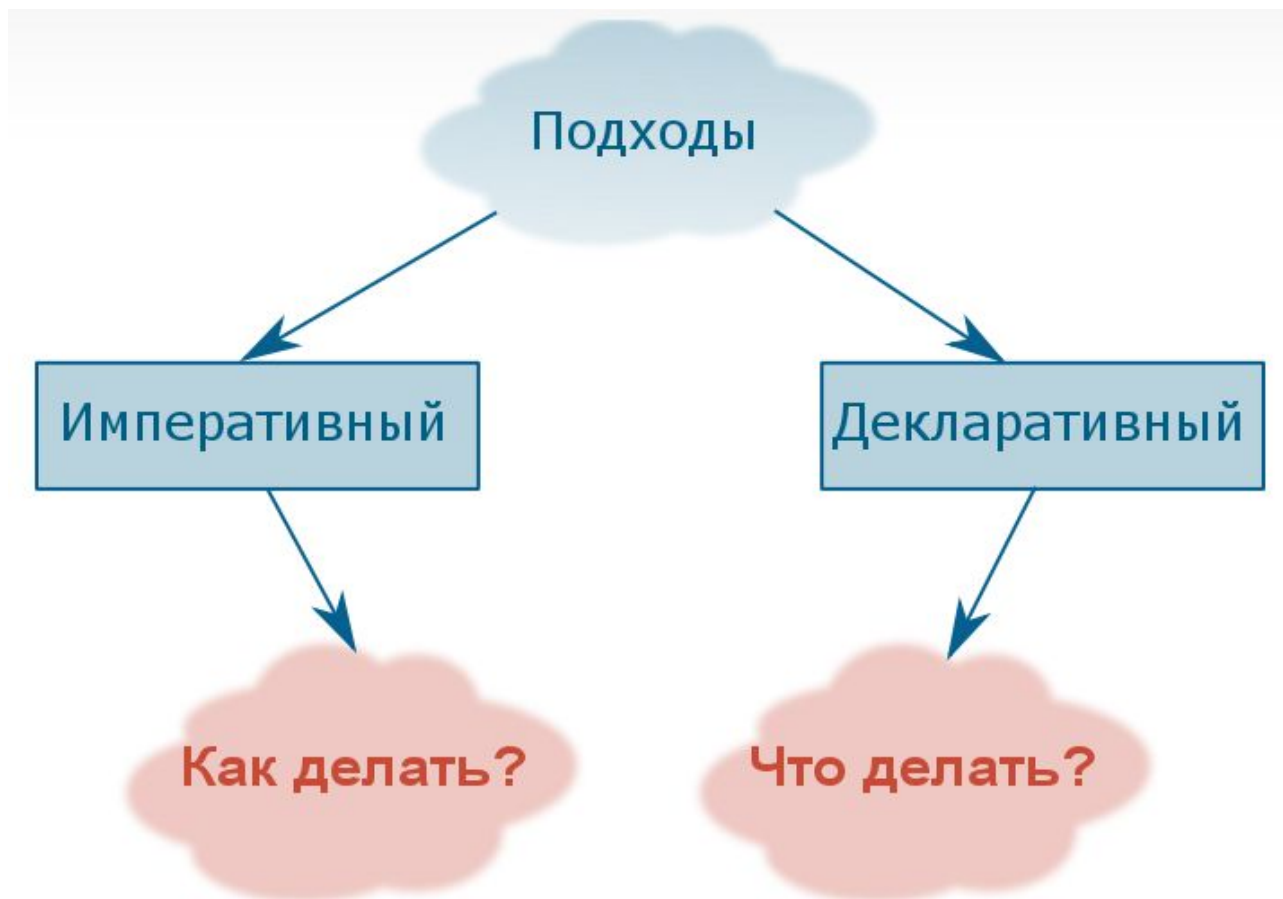
2011 — стандарт SQL 2011

2016 — стандарт SQL 2016

Декларативное программирование

парадигма программирования, в которой задаётся спецификация решения задачи, то есть описывается ожидаемый результат, а не способ его получения

SQL — декларативный язык



Диалекты SQL

- **PL/SQL (Procedural Language SQL)**

Oracle

- **T/SQL (Transact SQL)**

MSSQL

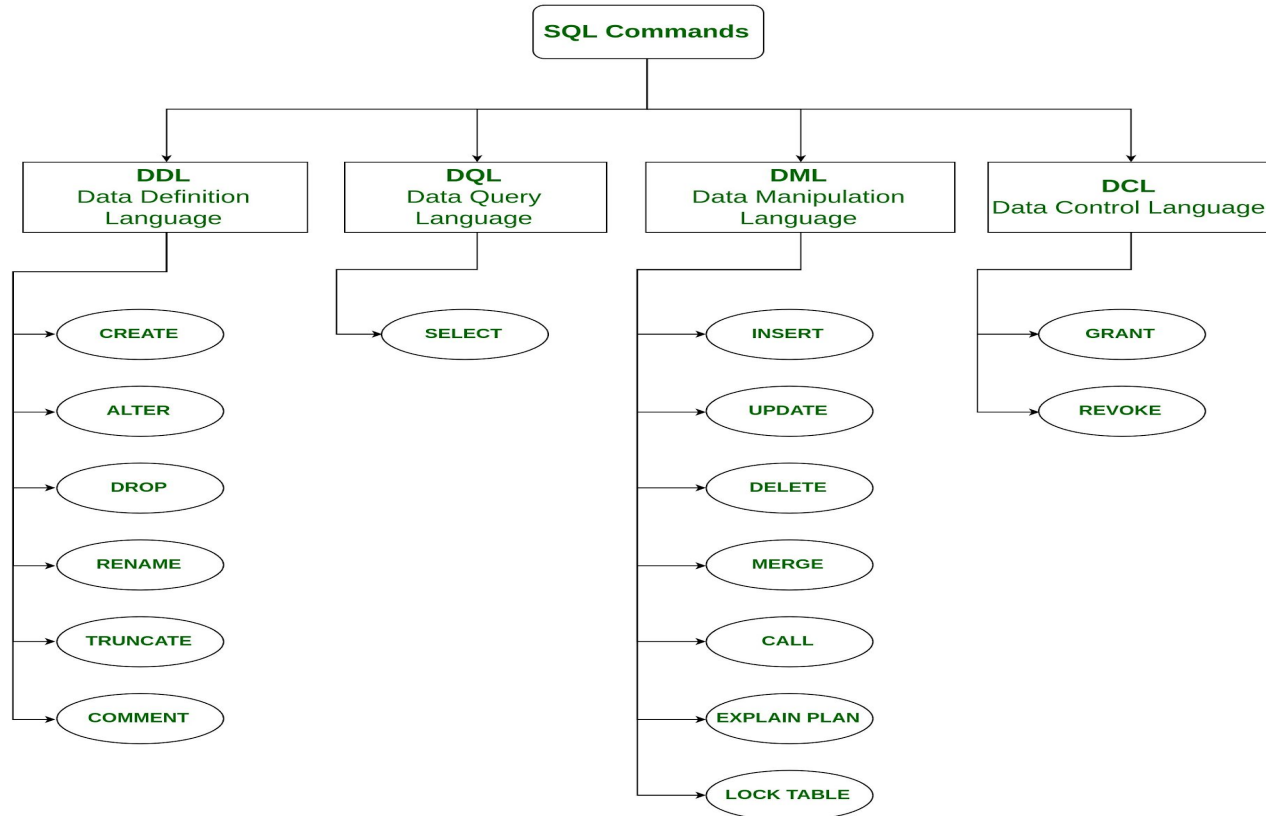
- **PL/pgSQL (Procedural Language PostGres SQL)**

Postgresql

- **Множество других**

- в каждой СУБД, свой диалект

Types of SQL Commands



Основные операторы SQL

Data Definition Language (DDL) – это группа операторов определения данных

- **create**

создание новых объектов базы данных

- **alter**

изменение существующих объектов

- **drop**

удаление объектов

Основные операторы SQL

Data Manipulation Language (DML) – это группа операторов для манипуляции с данными

- **insert**

добавление новых данных

- **update**

изменение данных

- **delete**

удаление данных

- **select**

- выборка данных

Основные операторы SQL

Data Control Language (DCL) – группа операторов определения доступа к данным

- **grant**
предоставить права на объект базы
- **revoke**
отозвать у пользователя права на объект
- **deny**
запрет на объект, имеет более высокий приоритет чем grant

Основные операторы SQL

Transaction Control Language (TCL) – группа операторов для управления транзакциями

Транзакция - это набор команд или инструкций которые выполняются как единый блок

- **begin transacton**

определяет начало транзакции

- **commit transaction**

применяет транзакцию, конечная точка

- **rollback transaction**

откатывает все изменения транзакции

- **save transaction**

- устанавливает промежуточную точку сохранения внутри транзакции

Основные компоненты СУБД

- **Storage engine**

подсистема постоянного хранения данных

- **Query parser**

парсер и транслятор запросов

- **Query optimizer**

оптимизатор запросов

- **Query executor**

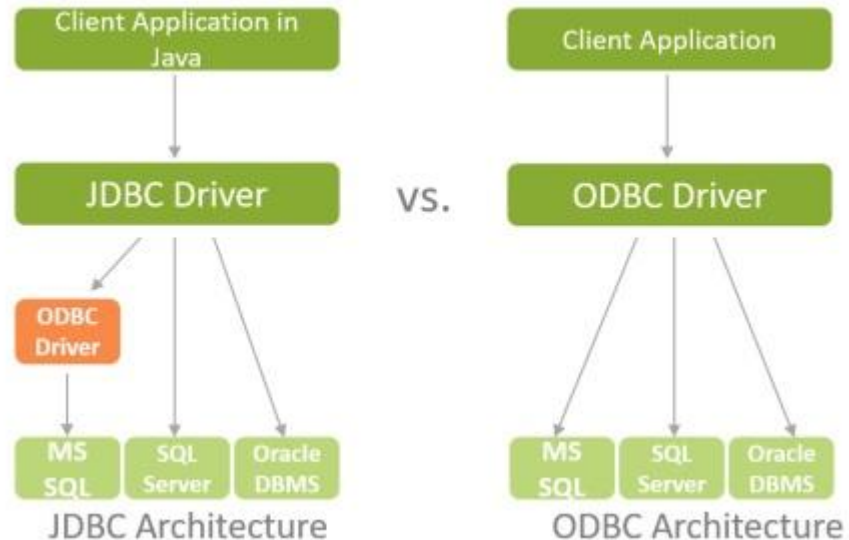
- подсистема выполнения

- **Buffer cache**

- подсистемы кэширования данных

Подключение к СУБД, компоненты доступа

- **ODBC (Open Database Connectivity)**
программный интерфейс (API) доступа к базам данных
- **JDBC (Java DataBase Connectivity)**
программный интерфейс (API) доступа к базам данных в среде Java
- **Другие провайдеры**



Клиентские приложения для работы с СУБД

- **Dbeaver**

open-source, поддержка более 20ти различных СУБД

- **Azure Data Studio**

Microsoft, поддержка различных СУБД за счет плагинов

- **dbForge**

компания Devart, требуется покупка лицензии

- **DataGrip**

- компания JetBrains, требуется покупка лицензии

- **MSMS**

- компания Microsoft, только для MSSQL

- **PGAdmin**

- open-source, только для Postgres

Online sql песочницы

• sqlfiddle.com

• db-fiddle.com

The screenshot shows the SQL Fiddle web application interface. The browser address bar displays the URL `sqlfiddle.com/#!15/9e41c/1`. The application header includes the text "SQL Fiddle", the database version "PostgreSQL 9.3", and buttons for "View Sample Fiddle", "Clear", "Text to DDL", "Donate", and "About".

The main interface is divided into two panels for SQL queries. The left panel contains the following SQL code:

```
1 create table if not exists Students
2 (
3   id SERIAL primary key
4   ,first_name text
5   ,second_name text
6 );
7
8 insert into Students(first_name,second_name)
9 values('Смирнов','Петр');
10
```

The right panel contains the following SQL query:

```
1 select * from Students
```

Below the query panels are buttons for "Build Schema", "Edit Fullscreen", "Browser", and "Run SQL". The "Run SQL" button has been executed, resulting in a table of data.

id	first_name	second_name
1	Смирнов	Петр

At the bottom of the interface, a green status bar displays the message: "Record Count: 1; Execution Time: 3ms". It also includes links for "View Execution Plan" and "link".

Заполнение и просмотр данных, используя графический интерфейс

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

Filter objects

- mysampledб
- sakila
- sys
- users**
 - Tables
 - student**
 - Columns
 - Indexes
 - Foreign Keys
 - Triggers
 - Views
 - Stored Procedures
 - Functions
 - world

student x users.student student - Table

Limit to 1000 rows

```
1 • SELECT * FROM users.student;
```

Result Grid

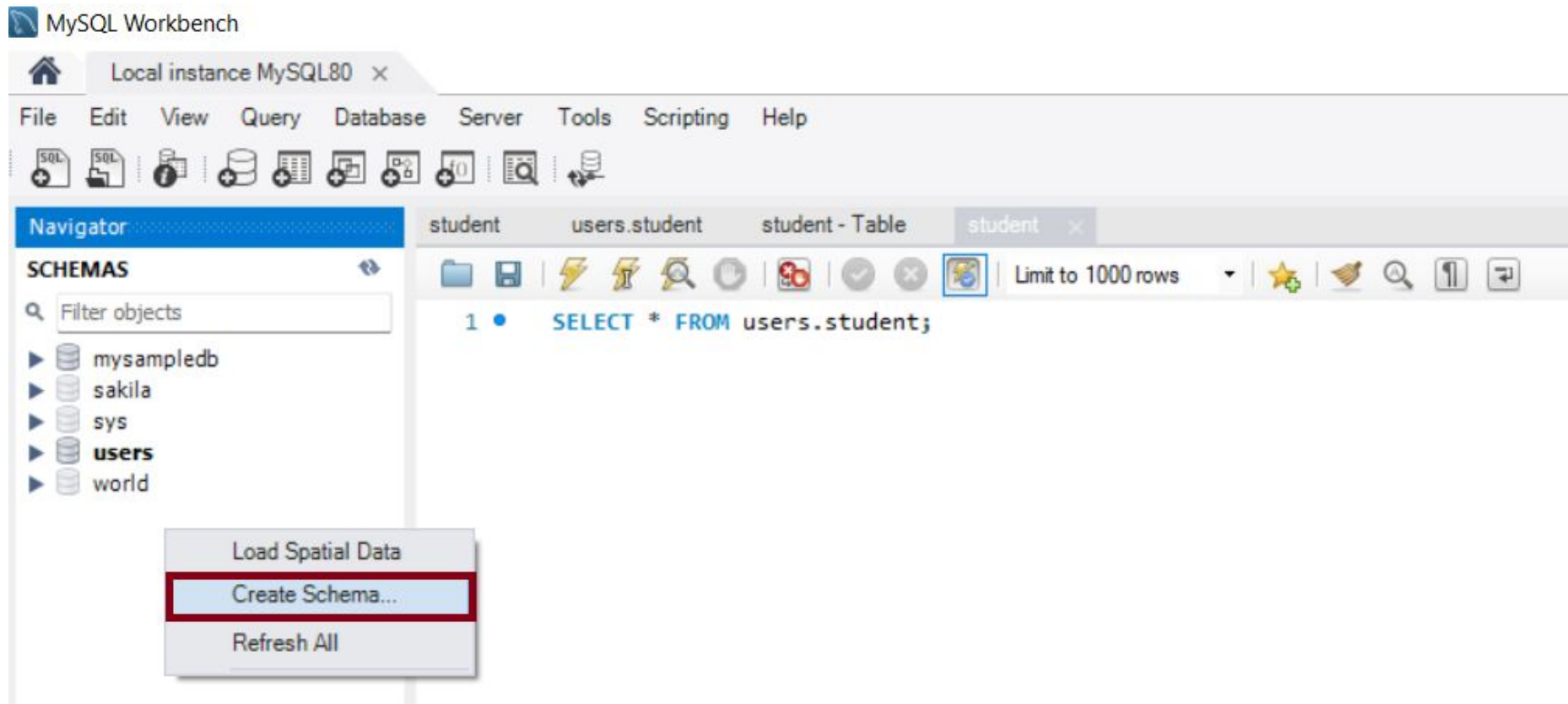
Filter Rows:

Edit: Export/Import: Wrap Cell Content:

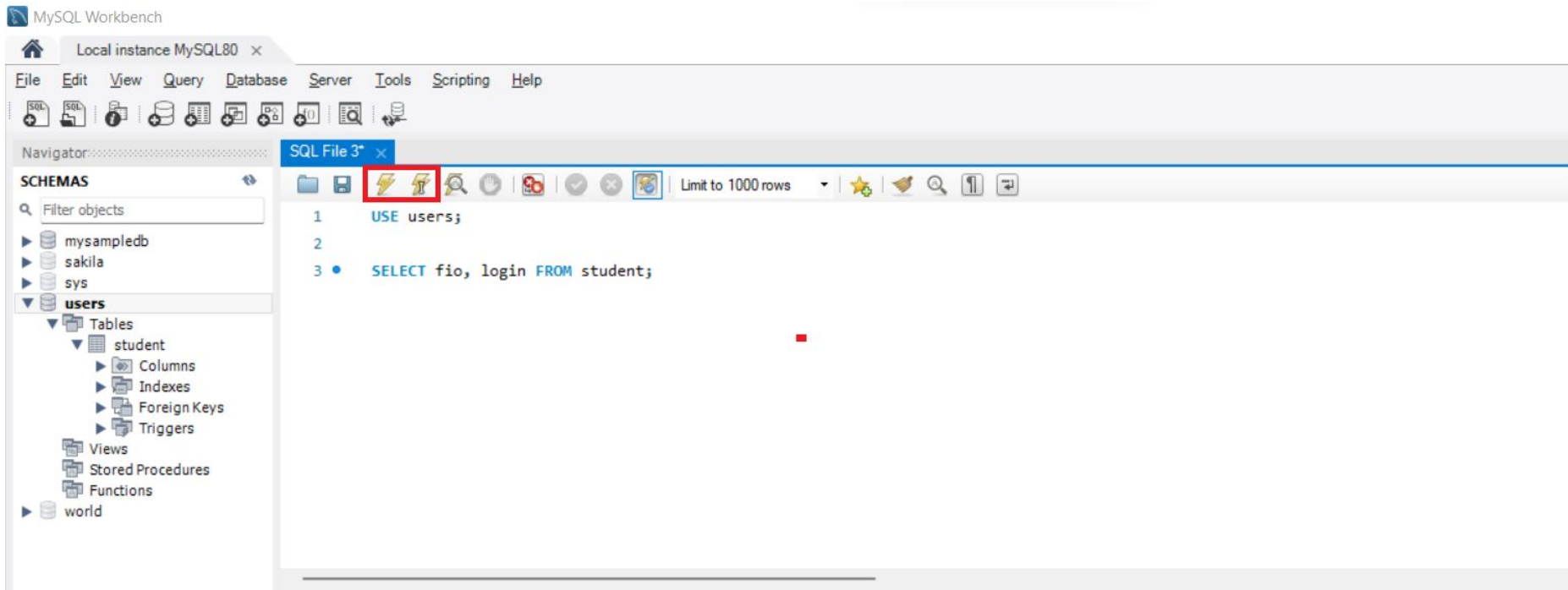
	id	fio	login	password	email
▶	1	Петров	test	123	petr@gmail.com
	2	Сидоров	test2	234	sid@gmail.com
*	NULL	NULL	NULL	NULL	NULL

Administration Schemas Information

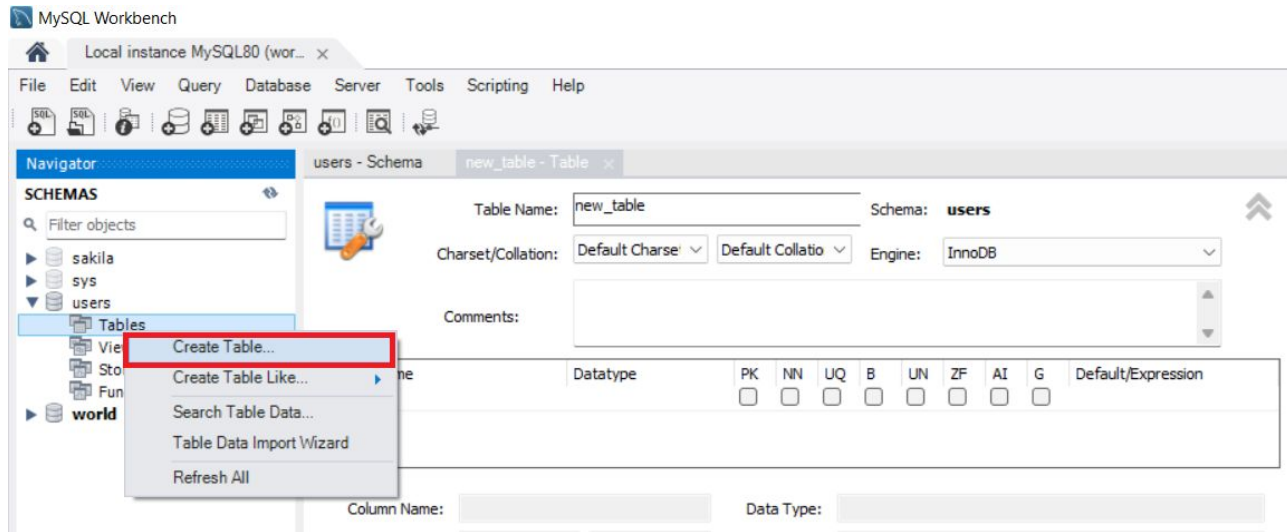
Создание схемы в MySQL Workbench



Выполнение запросов в MySQL



Создание таблицы в MySQL Workbench

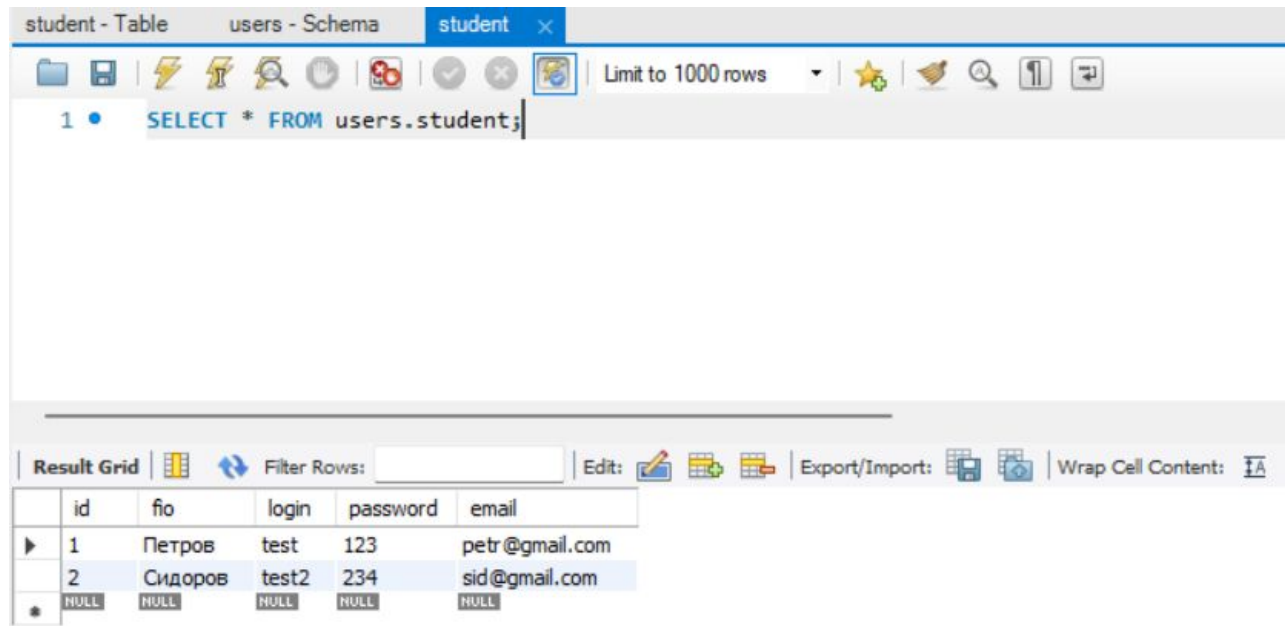


Запрос выборки данных с простыми условиями

```
1 SELECT поле1, поле2, поле3 from имя_таблицы
2   [WHERE предложение]
3   [GROUP BY {имя_столбца}]
4   [HAVING условие_where]
5   [ORDER BY {имя_столбца} {ASC | DESC}, ...] [LIMIT{OFFSET M}{LIMIT N}]
```

Пример вывода всех пользователей через SELECT (оператор “*”)

```
1 SELECT * FROM student;
```



The screenshot shows a database management interface with a query editor and a results grid. The query editor contains the SQL statement: `SELECT * FROM users.student;`. The results grid displays the data returned by the query, which includes two rows of user information. The first row has values: id=1, fio=Петров, login=test, password=123, email=petr@gmail.com. The second row has values: id=2, fio=Сидоров, login=test2, password=234, email=sid@gmail.com. Below the results grid, there is a row of NULL values.

	id	fio	login	password	email
▶	1	Петров	test	123	petr@gmail.com
	2	Сидоров	test2	234	sid@gmail.com
*	NULL	NULL	NULL	NULL	NULL

Вывод ограниченного числа столбцов — нужно явно перечислить столбцы

```
1 SELECT fio, login FROM student;
```

Result Grid			Filter Rows:	
	fio	login		
▶	Петров	test		
	Сидоров	test2		

Запросы SELECT с применением фильтров

Result Grid					
		Filter Rows:		Edit:	
	id	fio	login	password	email
▶	1	Петров	test	123	petr@gmail.com
	2	Сидоров	test2	234	sid@gmail.com
•	NULL	NULL	NULL	NULL	NULL

SQL File 3* x

Limit to 1000 rows

1 **USE** users;

2

3 • **SELECT** * **FROM** student **WHERE** login = "test2";

Result Grid | | Filter Rows: | Edit: | | Export/Import: | Wrap Cell Content:

	id	fio	login	password	email
▶	2	Сидоров	test2	234	sid@gmail.com
•	NULL	NULL	NULL	NULL	NULL