Работа с БД

-- создание БД

CREATE DATABASE database\_name;

-- удаление БД

DROP DATABASE database\_name;

-- просомтр списока всех БД

с

Работа с таблицами

-- создание таблицы

CREATE TABLE table\_name (

column1 datatype,

column2 datatype,

column3 datatype,

....

);

-- удаление таблицы

DROP TABLE table\_name;

ALTER TABLE - изменение столбцов

-- добавление нового поля

ALTER TABLE table\_name

ADD column\_name datatype;

-- удаление поля

ALTER TABLE table\_name

DROP COLUMN column\_name;

-- изменение поля (тип данных)

ALTER TABLE table\_name

MODIFY COLUMN column\_name datatype;

-- изменение имени поля

ALTER TABLE table\_name

CHANGE old\_column\_name new\_col\_name datatype;

UPDATE - изменение данных

UPDATE table\_name

SET column1 = value1, column2 = value2, ...

WHERE condition;

set sql\_safe\_updates = 0;

DELETE - удаление данных

--удаление строки

DELETE FROM table\_name

WHERE condition;

-- удаление всего содержимого таблицы

DELETE FROM table\_name;

set sql\_safe\_updates = 0;

DELETE FROM table\_name;

set sql\_safe\_updates = 0;

DELETE FROM table\_name;

ALTER TABLE students

DROP CHECK students\_chk\_1;

*/\**

*\*\*Создание таблиц\*\**

*0) Создать БД University*

​

*1) создать таблицу Students*

​

*с полями:*

​

*- id целое число первичный ключ автоинкремент*

​

*- name строка 128 не null*

​

*- age целое число*

​

*2) создать таблицу Teachers*

​

*с полями:*

​

*- id целое число первичный ключ автоинкремент*

​

*- name строка 128 не null*

​

*- age целое число*

​

*3) создать таблицу Competencies*

​

*с полями:*

​

*- id целое число первичный ключ автоинкремент*

​

*- title строка 128 не null*

​

*4) создать таблицу Teachers2Competencies*

​

*с полями:*

​

*- id целое число первичный ключ автоинкремент*

​

*- teacher\_id целое число*

​

*- competencies\_id целое число*

​

*5) создать таблицу Courses*

​

*- id целое число первичный ключ автоинкремент*

​

*- teacher\_id целое число*

​

*- title строка 128 не null*

​

*- headman\_id целое число*

​

*6) создать таблицу Students2Courses*

​

*- id целое число первичный ключ автоинкремент*

​

*- student\_id целое число*

​

*- course\_id целое число*

*\*/*

​

create database University;

use University;

​

create table Students (

id integer primary key auto\_increment,

name varchar(128) not null,

age integer

);

​

create table Teachers (

id integer primary key auto\_increment,

name varchar(128) not null,

age integer

);

​

create table Competencies (

id integer primary key auto\_increment,

title varchar(128) not null

);

​

create table Teachers2Competencies (

id integer primary key auto\_increment,

teacher\_id integer,

competencies\_id integer

);

​

create table Courses (

id integer primary key auto\_increment,

teacher\_id integer,

title varchar(128) not null,

headman\_id integer

);

​

create table Students2Courses (

id integer primary key auto\_increment,

student\_id integer,

course\_id integer

);

​

​

*/\**

*Добавить 6 записей в таблицу Students*

​

*Анатолий 29*

*Олег 25*

*Семен 27*

*Олеся 28*

*Ольга 31*

*Иван 22*

​

*Добавить 6 записей в таблицу Teachers*

​

*Петр 39*

*Максим 35*

*Антон 37*

*Всеволод 38*

*Егор 41*

*Светлана 32*

​

*Добавить 4 записей в таблицу Competencies*

​

*Математика*

*Информатика*

*Программирование*

*Графика*

​

*Добавьте 6 записей в таблицу Teachers2Competencies*

​

*1 1*

*2 1*

*2 3*

*3 2*

*4 1*

*5 3*

​

*Добавьте 5 записей в таблицу Courses*

​

*1 Алгебра логики 2*

*2 Математическая статистика 3*

*4 Высшая математика 5*

*5 Javascript 1*

*5 Базовый Python 1*

​

*Добавьте 5 записей в таблицу students2courses*

​

*1 1*

*2 1*

*3 2*

*3 3*

*4 5*

*\*/*

​

insert into Students (name, age) values ('Анатолий', 29), ('Олег', 25), ('Семен', 27), ('Олеся', 28), ('Ольга', 31), ('Иван', 22);

insert into Teachers (name, age) values ('Петр', 39), ('Максим', 35), ('Антон', 37), ('Всеволод', 38), ('Егор', 41), ('Светлана', 32);

insert into Competencies (title) values ('Математика'), ('Информатика'), ('Программирование'), ('Графика');

insert into Teachers2Competencies (teacher\_id, competencies\_id) values (1, 1), (2, 1), (2, 3), (3, 2), (4, 1), (5, 3);

insert into Courses (teacher\_id, title, headman\_id) values (1, 'Алгебра логики', 2), (2, 'Математическая статистика', 3),

(4, 'Высшая математика', 5), (5, 'Javascript', 1), (5, 'Базовый Python', 1);

insert into Students2Courses (student\_id, course\_id) values (1, 1), (2, 1), (3, 2), (3, 3), (4, 5);

​

​

*-- ЗАДАЧИ*

*-- 1) вывести имена студентов и курсы, которые они проходят*

​

select

t1*.name*,

t3*.title*

from students t1

inner join Students2Courses t2

on t1*.id* = t2*.student\_id*

inner join Courses t3

on t2*.course\_id* = t3*.id*;

​

*-- 1b) вывести имена ВСЕХ студентов и курсы, которые они проходят*

​

select

t1*.name*,

t3*.title*

from students t1

left join Students2Courses t2

on t1*.id* = t2*.student\_id*

left join Courses t3

on t2*.course\_id* = t3*.id*;

​

*-- 2) выведите имена всех преподавателей с их компетенциями*

​

select

t1*.name*,

t3*.title*

from teachers t1

left join Teachers2Competencies t2

on t1*.id* = t2*.teacher\_id*

left join Competencies t3

on t2*.competencies\_id* = t3*.id*;

​

*-- 3) найдите преподавателя, у которого нет компетенций*

​

select

t1*.name*,

t3*.title*

from teachers t1

left join Teachers2Competencies t2

on t1*.id* = t2*.teacher\_id*

left join Competencies t3

on t2*.competencies\_id* = t3*.id*

where t3*.title* is null;

​

*-- 4) Найдите имена студентов, которые не проходят ни один курс*

​

select

t1*.name*,

t3*.title*

from students t1

left join Students2Courses t2

on t1*.id* = t2*.student\_id*

left join Courses t3

on t2*.course\_id* = t3*.id*

where t3*.title* is null;

​

​

select

t1*.name*

from students t1

left join Students2Courses t2

on t1*.id* = t2*.student\_id*

where t2*.student\_id* is null;

​

*-- 5) Найдите курс, который не посещает ни один студент*

​

select

\*

from courses t1

left join Students2Courses t2

on t1*.id* = t2*.course\_id*

where t2*.course\_id* is null;

​

*-- 6) выведите название курса и имя старосты*

​

select

t1*.title*,

t2*.name*

from courses t1

inner join students t2

on t1*.headman\_id* = t2*.id*;

​

​

*-- ДЗ*

*-- 7) найдите компетенции, которых нет ни у одного преподавателя*

*-- 8) выведите имя студента и имена старост, которые есть на курсах, которые он проходит*

​

**Self JOIN**

**SELECT**

**\***

**FROM Customers t1**

**inner join Customers t2**

**on t1.customer\_id = t2.customer\_id;**

**/\***

**АГРЕГАЦИЯ**

**1. count() - кол-во строк, которые не null**

**2. max() - максимальное значение**

**3. min() - минимальное значение**

**4. avg() - среднее значение**

**5. sum() - сумма значений**

**\*/**

**SELECT**

**count(\*) as orders\_count**

**FROM orders;**

**SELECT**

**count(\*) as orders\_count**

**FROM orders;**

**SELECT**

**avg(amount) as avg\_amount**

**from orders;**

**SELECT**

**sum(amount) as sum\_amount,**

**count(\*) as ordes\_count,**

**avg(amount) as avg\_amount**

**from orders;**

**SELECT**

**item,**

**sum(amount) as sum\_amount,**

**count(\*) as ordes\_count,**

**avg(amount) as avg\_amount**

**from orders**

**group by item;**

**SELECT**

**item,**

**count(amount) as count\_amount**

**from orders**

**group by item;**

**-- Найти сумму возрастов в разбивке по странам**

**-- USA - 53**

**-- UK - 47**

**-- UAE - 28**

**SELECT**

**country,**

**sum(age) as sum\_age**

**from Customers**

**group by country;**

* **Найти средний возраст всех клиентов из USA**

**SELECT**

**country,**

**avg(age) as avg\_age**

**from Customers**

**where country = 'USA';**

метод Object equals() для JAVA