**Министерство науки и высшего образования Российской Федерации**

федеральное государственное автономное образовательное учреждение высшего образования

**«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»**

**Отчет**

по лабораторной работе «Создание БД на helios**»**

по дисциплине «**Базы данных**»

Вариант 1489

Автор: Суворов Денис

Факультет: ПИиКТ

Группа: P33201

Преподаватель: Машина Екатерина Алексеевна

Санкт-Петербург 2022

# **Задание**

Для выполнения лабораторной работы №4 необходимо:

* Реализовать разработанную в рамках лабораторной работы №3 даталогическую модель в реляционной СУБД PostgreSQL.
* Заполнить созданные таблицы данными.
* Обеспечить целостность данных при помощи средств языка DDL.
* В рамках лабораторной работы должны быть разработаны скрипты для создания/удаления требуемых объектов базы данных, заполнения/удаления содержимого созданных таблиц.

# **Реализация даталогической модели в PostgreSQL**

CREATE TABLE Parent (

"ID" serial PRIMARY KEY ,

"Имя" varchar(45) NOT NULL,

"Фамилия" varchar(45) NOT NULL,

"Телефон" varchar(45) NOT NULL UNIQUE,

"Пол" varchar(15) NOT NULL

);

CREATE TABLE Student (

"ID" serial PRIMARY KEY ,

"Имя" varchar(45) NOT NULL,

"Фамилия" varchar(45) NOT NULL,

"Телефон" varchar(45) NOT NULL UNIQUE,

"Пол" varchar(15) NOT NULL,

"Parent\_id" INT,

FOREIGN KEY ("Parent\_id") REFERENCES Parent("ID")

);

CREATE TABLE Office (

"ID" serial PRIMARY KEY ,

"Город" varchar(45) NOT NULL

);

CREATE TABLE Teachers (

"ID" serial PRIMARY KEY ,

"Имя" varchar(45) NOT NULL,

"Фамилия" varchar(45) NOT NULL,

"Телефон" varchar(45) NOT NULL UNIQUE,

"Пол" varchar(15) NOT NULL

);

CREATE TABLE Tasks(

"ID" SERIAL PRIMARY KEY,

"Тема" VARCHAR(45) NOT NULL,

"Название" VARCHAR(45) NOT NULL,

"Баллы" INT NOT NULL

);

CREATE TABLE Lessons (

"ID" serial PRIMARY KEY,

"Названиие" varchar(45) NOT NULL,

"Тема" VARCHAR (45) NOT NULL,

"Task\_id" INT,

FOREIGN KEY ("Task\_id") REFERENCES Tasks("ID")

);

CREATE TABLE Course (

"ID" serial PRIMARY KEY ,

"Название" varchar(45) NOT NULL,

"Урок\_id" INT,

FOREIGN KEY ("Урок\_id") REFERENCES Lessons("ID")

);

CREATE TABLE "group" (

"ID" serial PRIMARY KEY ,

"Офис\_id" INT,

"Учитель\_id" INT,

"Course\_id" INT,

FOREIGN KEY ("Учитель\_id") REFERENCES Teachers("ID"),

FOREIGN KEY ("Course\_id") REFERENCES Course("ID"),

FOREIGN KEY ("Офис\_id") REFERENCES Office("ID")

);

CREATE TABLE Student\_aggregate(

"ID" serial PRIMARY KEY ,

"Студент\_id" INT,

"Курс\_id" INT,

"Группа\_id" INT,

"Урок\_id" INT,

"Задание\_id" INT,

"Статус" bool,

FOREIGN KEY ("Студент\_id") REFERENCES Student("ID"),

FOREIGN KEY ("Группа\_id") references "group"("ID"),

FOREIGN KEY ("Курс\_id") references Course("ID"),

FOREIGN KEY ("Урок\_id") references Lessons("ID"),

FOREIGN KEY ("Задание\_id") references Tasks("ID")

);

CREATE TABLE Group\_student(

"ID" serial PRIMARY KEY,

"Группа\_id" INT,

"Студент\_id" INT,

"Статус" bool NOT NULL,

FOREIGN KEY ("Студент\_id") references Student("ID"),

FOREIGN KEY ("Группа\_id") references "group"("ID")

)

Удаление таблицы

DROP TABLE IF EXISTS Student CASCADE

если при удалении строки из главной таблицы необходимо, чтобы были удалены все связанные строки из зависимой таблицы, то применяется каскадное удаление

Добавление нового столбца

ALTER TABLE Student ADD COLUMN pay INT CHECK (pay > 0)

Заполнение таблиц данными

INSERT INTO parent ( "Имя", "Фамилия", "Пол","Телефон" )

VALUES

( 'Денис','Суворов','М',89235024455 ),

( 'Александр','Виронов','М',89235552244 ),

( 'Алексей','Миронов','М',89235024444),

( 'Степан','Разин','М',89235025555 ),

( 'Наталья','Овсинова','Ж',89235023333 );

INSERT INTO student ( "Имя", "Фамилия", "Пол","Телефон","Parent\_id" )

VALUES

( 'Артем','Суворов','М',89235034455, 1 ),

( 'Дмитрий','Виронов','М',89234552244, 2 ),

( 'Геннадий','Миронов','М',89236024444, 3),

( 'Платон','Разин','М',89235024555, 4 ),

( 'София','Овсинова','Ж',89235025333, 5 );

INSERT INTO office ("Город")

VALUES

('Кемерово'),

('Москва'),

('Новосибирск');

INSERT INTO teachers ( "Имя", "Фамилия", "Пол","Телефон")

VALUES

( 'Виктория','Павловна','Ж',89237034455),

( 'Лев','Мальцев','М',89334552244),

( 'Ася','Макарова','Ж',89436024444),

( 'Платон','Анисимов','М',89535024555),

( 'Елизавета','Шангина','Ж',89635025333);

INSERT INTO Tasks ( "Тема", "Название", "Баллы" )

VALUES

( 'IT','Программирование',2),

( 'IT','Математика',3),

( 'IT','Программирование',2),

( 'IT','Математика',3),

( 'IT','Программирование',2),

( 'IT','Математика',3),

( 'IT','Программирование',2),

( 'IT','Программирование',2),

( 'IT','Математика',3),

( 'IT','Программирование',2);

INSERT INTO Lessons ("Названиие", "Тема", "Task\_id")

VALUES

( 'Начало прогаммирования','IT',1),

( 'Начало прогаммирования','IT',3),

( 'Начало математика','Math',2),

( 'Контрольная работа','Test',7),

( 'Все об IT','IT',1),

( 'Дополнительный урок','Extra',10);

INSERT INTO Course ("Название", "Урок\_id")

VALUES

('Подготовительная группа', 1),

('Математика', 3),

('Для вводного урока', 5),

('Для вводного урока', 1);

INSERT INTO "group" ("Офис\_id","Учитель\_id","Course\_id")

VALUES

(1,1,1),

(2,2,2),

(3,3,3);

INSERT INTO Group\_student("Группа\_id","Студент\_id","Статус")

VALUES

(1,1,TRUE),

(2,1,FALSE),

(3,2,TRUE),

(3,4,TRUE);

INSERT INTO Student\_aggregate("Студент\_id","Курс\_id","Группа\_id","Урок\_id","Задание\_id","Статус")

VALUES

(1,1,1,1,1,TRUE),

(2,3,3,5,1,TRUE);

## 

## 