**Санкт-Петербургский политехнический университет Петра Великого**

**Институт кибербезопасности и защиты информации**

**ЛАБОРАТОРНАЯ РАБОТА №1**

**Разграничение доступа к СУБД**

по дисциплине «Основы построения защищенных баз данных»

Выполнил

студент гр. 4841001/00101 *<подпись>* А.С. Петушков

Проверил

преподаватель *<подпись>* М.А. Полтавцева

Санкт-Петербург  
2022

**Цель работы**

Получение навыков разграничения доступа при работе с СУБД.

**Задачи работы**

1. Разработать и реализовать логическую схему данных. Этот этап может выполняться совместно студентами, схема данных в вариантах которых совпадает.
2. Разработать и реализовать ограничения целостности. Учесть естественные ограничения целостности (например, число знаков и их тип в номере телефона) и дополнительно указанные в задании.
3. Заполнить базу данных информацией для реализации примеров разграничения доступа, представлений и т.д. Число записей должно быть достаточным для иллюстрации дальнейших действий.
4. Разработать матрицу доступа к данным, детализированную до атрибута. Права доступа к данным определить из набора CRUD (Create, Read, Update, Delete).
   1. Отразить права по отношению к атрибутам каждого пользователя.
   2. Отразить ограничения на работу с атрибутами (ограничения горизонтального доступа к данным, если пользователь имеет доступ не ко всем кортежам).
   3. Отразить перечень агрегированных показателей (согласно варианту) и права доступа к ним.
5. Реализовать разграничение доступа при помощи встроенных средств СУБД.
   1. Реализовать разграничение доступа к базовым объектам базы данных при помощи встроенных средств.
   2. При необходимости для разграничения доступа разработать пользовательские функции.
6. На копии базы данных реализовать разграничение доступа при помощи представлений.
   1. Реализовать представления.
   2. Реализовать разграничение доступа к представлениям при помощи встроенных средств СУБД. При необходимости – разработать пользовательские функции.
   3. Обеспечить обновляемость представлений, где это необходимо.

**Вариант №10 – Учебный центр**

**Данные:**

Преподаватель (ПДН включая ФИО и номер(а) документа(ов), специальность, квалификация, стаж, предыдущие места работы, рейтинг, зарплата)

Обучающийся (ПДН включая ФИО и номер(а) документа(ов), рейтинг, совершеннолетие, законные представители)

Курсы (название, аннотация, предшествующие курсы, длительность по времени, стоимость, специальное оборудование / ПО, преподаватели, обучающиеся)

Журнал оценок (преподаватель, обучающийся, курс, промежуточные оценки на курсе включая отдельные тесты, сданные задания, итоговая оценка)

Расписание занятий (преподаватель, курс, кабинет, обучающийся, время начала, время окончания)

Кабинеты (тип кабинета, номер, специальное оборудование / ПО)

**Ограничения целостности:**

Рейтинг обучающегося не может уменьшаться ниже пороговых значений (0, 5, 10 и т.д.) если их достиг.

Любой курс не может быть дешевле, чем любой курс, предшествующий ему.

**Пользователи и политика безопасности:**

Обучающийся видит собственные данные, кроме рейтинга.

Обучающийся может менять собственные данные, кроме статуса совершеннолетия и списка законных представителей.

Обучающийся может создать себе новую запись, тогда рейтинг будет нулевым, а статус и законные представители неизвестны.

Администратор видит среднюю, минимальную и максимальную стоимость курсов по преподавателям, без прочей детализации.

**Ход работы**

На рисунке 1 представлена логическая схема данных.

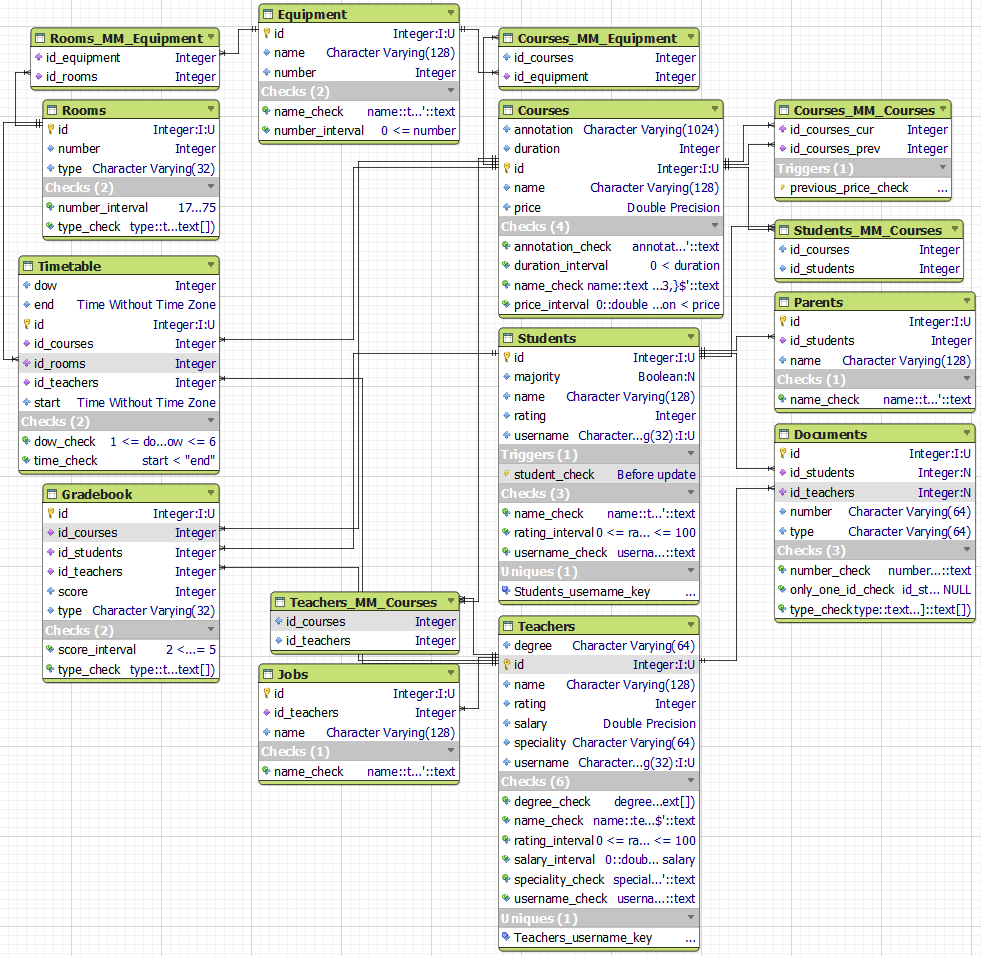


Рисунок 1 – Логическая схема данных

Реализованные объекты содержат различные ограничения целостности. Они отражены на рисунке 1. Также реализованы следующие триггеры:

* Таблица Courses\_MM\_Courses: любой курс не может быть дешевле, чем любой курс, предшествующий ему.
* Таблица Students: рейтинг обучающегося не может уменьшаться ниже пороговых значений (0, 5, 10 и т.д.) если их достиг.

В приложении 2 приведен код, выполняющий создание данной схемы.

В приложении 4 приведен код, выполняющий заполнение схемы тестовыми данными.

В таблице 1 представлена матрица доступа. Где необходимо, указаны поля таблиц.

Таблица 1 – Матрица доступа

|  |  |  |
| --- | --- | --- |
| **Таблицы и поля** | **Администратор** | **Обучающийся** |
| **Students** |  |  |
| id |  | CR |
| username |  | CR |
| name |  | CRU |
| rating |  |  |
| majority |  | R |
| **Teachers** |  |  |
| id | R |  |
| username |  |  |
| name | R |  |
| speciality |  |  |
| degree |  |  |
| rating |  |  |
| salary |  |  |
| **Courses** |  |  |
| id | R |  |
| name |  |  |
| annotation |  |  |
| duration |  |  |
| price | R |  |
| **Documents** |  |  |
| id |  | CR |
| type |  | CRU |
| number |  | CRU |
| id\_students |  | CR |
| id\_teachers |  |  |
| **Parents** |  |  |
| id |  |  |
| name |  | R |
| id\_students |  | R |
| **Teachers\_MM\_Courses** |  |  |
| id\_teachers | R |  |
| id\_courses | R |  |
| **Jobs** |  |  |
| **Equipment** |  |  |
| **Courses\_MM\_Equipment** |  |  |
| **Rooms\_MM\_Equipment** |  |  |
| **Courses\_MM\_Courses** |  |  |
| **Gradebook** |  |  |
| **Rooms** |  |  |
| **Timetable** |  |  |
| **Students\_MM\_Courses** |  |  |

В приложении 1 приведены результаты тестирования. Здесь и далее ссылки на номера тестовых сценариев см. в этом приложении; результаты выполнения каждого сценария приводятся в секции Expected.

Тестовый сценарий 1.1 – Уменьшение рейтинга обучающегося ниже порогового значения – запрещается.

Тестовый сценарий 1.2 – Попытка добавления курса с большей стоимостью, чем у последующего – запрещается.

В приложении 3 приведен код, реализующий политику разграничения доступа (в том числе с помощью представлений) и создание пользователей.

Тестовые сценарии 2.1.X – Политика безопасности для обучающихся – чтение данных:

* попытка обучающегося узнать свой рейтинг – запрещается;
* попытка обучающегося получить свои основные данные – разрешается;
* попытка обучающегося получить свои паспортные данные – разрешается;
* попытка обучающегося получить данные о своих законных представителях – разрешается.

Тестовые сценарии 2.2.X – Политика безопасности для обучающихся – обновление данных:

* попытка обучающегося изменить свое ФИО – разрешается;
* попытка обучающегося изменить свой статус совершеннолетия – запрещается.

Тестовый сценарий 2.3 – Политика безопасности для обучающихся – создание новой записи обучающимся – разрешается.

Тестовые сценарии 2.4.X – Политика безопасности для администратора:

* попытка администратора вызвать хранимую процедуру для получения средней стоимости курсов по преподавателю – разрешается;
* попытка обучающегося вызвать хранимую процедуру для получения средней стоимости курсов по преподавателю – запрещается.

Для реализации политики безопасности с помощью представлений произведены тесты, подобные предыдущим с таким же успешным результатом. Они приводятся в приложении 1 под номерами 3.X.

**Вывод**

В результате выполнения данной лабораторной работы были изучены механизмы разграничения доступа в СУБД PostgreSQL и получены навыки работы с ними. Согласно варианту была реализована тестовая база данных «Учебный центр» в ней внедрена политика разграничения доступа при помощи встроенных средств СУБД и при помощи представлений. В результате все поставленные задачи работы были успешно решены.

Приложение 1 – Результаты тестирования

/\*

\* Tests

\*/

/\* Test case 1.1 -- Decrease of Student's rating \*/

UPDATE "Students" SET "rating" = 19 WHERE "id" = 2;

-- Expected:

-- ERROR: student\_check(): Student's rating can't go lower then the threshold level

-- CONTEXT: PL/pgSQL function student\_check() line 7 at RAISE

-- SQL state: P0001

/\* Test case 1.2 -- Previous Courses' price \*/

INSERT INTO "Courses\_MM\_Courses" ("id\_courses\_cur", "id\_courses\_prev") VALUES (1, 0);

-- Expected:

-- ERROR: previous\_price\_check(): No course can be cheaper that any of it's previous course

-- CONTEXT: PL/pgSQL function previous\_price\_check() line 10 at RAISE

-- SQL state: P0001

/\* Test cases 2.1.X -- Student's policy - SELECT \*/

/\* Test case 2.1.1 \*/

RESET ROLE;

SET ROLE \_dementyeva;

SELECT "rating" FROM "Students";

-- Expected:

-- ERROR: permission denied for table Students

-- SQL state: 42501

RESET ROLE;

/\* Test case 2.1.2 \*/

RESET ROLE;

SET ROLE \_dementyeva;

SELECT "username", "name", "majority" FROM "Students";

-- Expected:

-- "\_dementyeva" "Дементьева Наталия Игоревна" true

RESET ROLE;

/\* Test case 2.1.3 \*/

RESET ROLE;

SET ROLE \_dementyeva;

SELECT "type", "number" FROM "Documents" WHERE "id\_students" = (SELECT "id" FROM "Students");

-- Expected:

-- "Паспорт" "4018 850788"

RESET ROLE;

/\* Test case 2.1.4 \*/

RESET ROLE;

SET ROLE \_dementyeva;

SELECT "name" FROM "Parents" WHERE "id\_students" = (SELECT "id" FROM "Students");

-- Expected:

-- "Дементьев Игорь Матвеевич"

-- "Дементьева Ольга Владиславовна"

RESET ROLE;

/\* Test cases 2.2.X -- Student's policy - UPDATE \*/

/\* Test case 2.2.1 \*/

RESET ROLE;

SET ROLE \_dementyeva;

UPDATE "Students" set "name" = 'Тест' WHERE "username" = '\_dementyeva';

-- Expected:

-- UPDATE 1

SELECT "name" FROM "Students";

-- Expected:

-- "Тест"

RESET ROLE;

/\* Test case 2.2.2 \*/

RESET ROLE;

SET ROLE \_dementyeva;

UPDATE "Students" set "majority" = FALSE WHERE "username" = '\_dementyeva';

-- Expected:

-- ERROR: permission denied for table Students

-- SQL state: 42501

RESET ROLE;

/\* Test case 2.3 -- Student's policy - INSERT \*/

RESET ROLE;

SET ROLE \_newstudent;

INSERT INTO "Students" ("id", "username", "name")

VALUES (10, '\_newstudent', 'Тест Тест');

-- Expected:

-- INSERT 0 1

RESET ROLE;

SELECT \* FROM "Students" WHERE "username" = '\_newstudent';

-- Expected:

-- 10 "\_newstudent" "Тест Тест" 0 [null]

/\* Test cases 2.4.X -- Administrator's policy \*/

/\* Test case 2.4.1 \*/

RESET ROLE;

SET ROLE role\_administrator;

SELECT \* FROM get\_price\_avg\_by\_teacher();

-- Expected:

-- 79000 "Сысоева Лариса Вениаминовна"

-- 120000 "Иванков Илья Дмитриевич"

-- 120000 "Тарасов Роман Семенович"

-- 40000 "Мишин Даниил Романович"

RESET ROLE;

/\* Test case 2.4.2 \*/

RESET ROLE;

SET ROLE role\_student;

SELECT \* FROM get\_price\_avg\_by\_teacher();

-- Expected:

-- ERROR: permission denied for table Teachers\_MM\_Courses

-- CONTEXT: SQL function "get\_price\_avg\_by\_teacher" statement 1

-- SQL state: 42501

RESET ROLE;

/\* Test case 3.1 -- Student's policy (views) - SELECT \*/

RESET ROLE;

SET ROLE \_dementyeva;

SELECT \* FROM view\_students, view\_documents, view\_parents;

-- Expected:

-- 0 "\_dementyeva" "Дементьева Наталия Игоревна" true 29 "Паспорт" "4018 850788" 0 "Дементьев Игорь Матвеевич" 0

-- 0 "\_dementyeva" "Дементьева Наталия Игоревна" true 29 "Паспорт" "4018 850788" 0 "Дементьева Ольга Владиславовна" 0

RESET ROLE;

/\* Test cases 3.2.X -- Student's policy (views) - UPDATE \*/

/\* Test case 3.2.1 \*/

RESET ROLE;

SET ROLE \_dementyeva;

UPDATE "view\_students" set "name" = 'Тест' WHERE "username" = '\_dementyeva';

-- Expected:

-- UPDATE 1

SELECT "name" FROM "view\_students";

-- Expected:

-- "Тест"

RESET ROLE;

/\* Test case 3.2.2 \*/

RESET ROLE;

SET ROLE \_dementyeva;

UPDATE "view\_students" set "majority" = FALSE WHERE "username" = '\_dementyeva';

-- Expected:

-- ERROR: permission denied for view view\_students

-- SQL state: 42501

RESET ROLE;

/\* Test case 3.3 -- Student's policy (views) - INSERT \*/

RESET ROLE;

SET ROLE \_newstudent;

INSERT INTO "view\_students" ("id", "username", "name")

VALUES (10, '\_newstudent', 'Тест Тест');

-- Expected:

-- INSERT 0 1

RESET ROLE;

SELECT \* FROM "Students" WHERE "username" = '\_newstudent';

-- Expected:

-- 10 "\_newstudent" "Тест Тест" 0 [null]

/\* Test cases 3.4.X -- Administrator's policy \*/

/\* Test case 3.4.1 \*/

RESET ROLE;

SET ROLE role\_administrator;

SELECT \* FROM get\_price\_avg\_by\_teacher();

-- Expected:

-- 79000 "Сысоева Лариса Вениаминовна"

-- 120000 "Иванков Илья Дмитриевич"

-- 120000 "Тарасов Роман Семенович"

-- 40000 "Мишин Даниил Романович"

RESET ROLE;

/\* Test case 3.4.2 \*/

RESET ROLE;

SET ROLE role\_student;

SELECT \* FROM get\_price\_avg\_by\_teacher();

-- Expected:

-- ERROR: permission denied for view view\_teachers\_mm\_courses

-- CONTEXT: SQL function "get\_price\_avg\_by\_teacher" statement 1

-- SQL state: 42501

RESET ROLE;

Приложение 2 – Код создания БД

/\*

\* Initial drop all tables and triggers

\*/

DROP TABLE IF EXISTS "Students" CASCADE;

DROP TABLE IF EXISTS "Teachers" CASCADE;

DROP TABLE IF EXISTS "Courses" CASCADE;

DROP TABLE IF EXISTS "Gradebook" CASCADE;

DROP TABLE IF EXISTS "Rooms" CASCADE;

DROP TABLE IF EXISTS "Timetable" CASCADE;

DROP TABLE IF EXISTS "Documents" CASCADE;

DROP TABLE IF EXISTS "Jobs" CASCADE;

DROP TABLE IF EXISTS "Parents" CASCADE;

DROP TABLE IF EXISTS "Equipment" CASCADE;

DROP TABLE IF EXISTS "Courses\_MM\_Equipment" CASCADE;

DROP TABLE IF EXISTS "Rooms\_MM\_Equipment" CASCADE;

DROP TABLE IF EXISTS "Courses\_MM\_Courses" CASCADE;

DROP TABLE IF EXISTS "Teachers\_MM\_Courses" CASCADE;

DROP TABLE IF EXISTS "Students\_MM\_Courses" CASCADE;

DROP TRIGGER IF EXISTS student\_check ON "Students";

DROP FUNCTION IF EXISTS student\_check();

DROP TRIGGER IF EXISTS previous\_price\_check ON "Courses\_MM\_Courses";

DROP FUNCTION IF EXISTS previous\_price\_check();

/\*

\* Tables creation

\*/

/\* Main tables \*/

CREATE TABLE "Students" (

/\* Attributes \*/

"id" integer,

"username" varchar(32) NOT NULL UNIQUE,

"name" varchar(128) NOT NULL,

"rating" integer NOT NULL DEFAULT 0,

"majority" BOOLEAN,

/\* Attributes constraints \*/

CONSTRAINT "username\_check" CHECK ( -- English alphabet and numbers. Length >= 3. Starts with '\_'

"username" ~ '^\_([a-z]|[A-Z]|[0-9]){3,}$'),

CONSTRAINT "name\_check" CHECK ( -- Russian alphabet, space and '-'. Length >= 3

"name" ~ '^([а-я]|[А-Я]|[ -]){3,}$'),

CONSTRAINT "rating\_interval" CHECK ( -- Rating in range [0, 100]

0 <= "rating" AND rating <= 100),

/\* Foreign keys \*/

/\* Primary key \*/

CONSTRAINT "Students\_pk" PRIMARY KEY ("id")

);

CREATE TABLE "Teachers" (

/\* Attributes \*/

"id" integer,

"username" varchar(32) NOT NULL UNIQUE,

"name" varchar(128) NOT NULL,

"speciality" varchar(64) NOT NULL,

"degree" varchar(64) NOT NULL,

"rating" integer NOT NULL,

"salary" FLOAT NOT NULL,

/\* Attributes constraints \*/

CONSTRAINT "username\_check" CHECK ( -- English alphabet and numbers. Length >= 3

"username" ~ '^([a-z]|[A-Z]|[0-9]){3,}$'),

CONSTRAINT "name\_check" CHECK ( -- Russian alphabet, space and '-'. Length >= 3

"name" ~ '^([а-я]|[А-Я]|[ -]){3,}$'),

CONSTRAINT "speciality\_check" CHECK ( -- Russian alphabet, space, ',' and '-'. Length >= 3

"speciality" ~ '^([а-я]|[А-Я]|[ ,-]){3,}$'),

CONSTRAINT "degree\_check" CHECK ( -- Допустимые значения

"degree" IN (

'Бакалавр'

, 'Специалист'

, 'Магистр'

, 'Кандидат наук'

, 'Доктор наук'

)),

CONSTRAINT "rating\_interval" CHECK ( -- Rating in range [0, 100]

0 <= "rating" AND rating <= 100),

CONSTRAINT "salary\_interval" CHECK ( -- Salary in range (0, inf)

0 < "salary"),

/\* Foreign keys \*/

/\* Primary key \*/

CONSTRAINT "Teachers\_pk" PRIMARY KEY ("id")

);

CREATE TABLE "Courses" (

/\* Attributes \*/

"id" integer,

"name" varchar(128) NOT NULL,

"annotation" varchar(1024) NOT NULL,

"duration" integer NOT NULL,

"price" FLOAT NOT NULL,

/\* Attributes constraints \*/

CONSTRAINT "name\_check" CHECK ( -- Russian alphabet, space and '-'. Length >= 3

"name" ~ '^([а-я]|[А-Я]|[ -]){3,}$'),

CONSTRAINT "annotation\_check" CHECK ( -- Russian and English alphabet, special symbols. Length >= 15

"annotation" ~ '^([а-я]|[А-Я]|[a-z]|[A-Z]|[0-9]|[ .,;:@№%()/-]){15,}$'),

CONSTRAINT "duration\_interval" CHECK ( -- Duration in range (0, inf)

0 < "duration"),

CONSTRAINT "price\_interval" CHECK ( -- Price in range (0, inf)

0 < "price"),

/\* Foreign keys \*/

/\* Primary key \*/

CONSTRAINT "Courses\_pk" PRIMARY KEY ("id")

);

CREATE TABLE "Gradebook" (

/\* Attributes \*/

"id" integer NOT NULL,

"type" varchar(32) NOT NULL,

"score" integer NOT NULL,

/\* Attributes constraints \*/

CONSTRAINT "type\_check" CHECK ( -- Допустимые значения

"type" IN (

'Занятие'

, 'Лаба'

, 'Тест'

, 'Итог'

)),

CONSTRAINT "score\_interval" CHECK ( -- Score in range [2, 5]

2 <= "score" AND score <= 5),

/\* Foreign keys \*/

"id\_courses" integer NOT NULL,

"id\_students" integer NOT NULL,

"id\_teachers" integer NOT NULL,

/\* Primary key \*/

CONSTRAINT "Gradebook\_pk" PRIMARY KEY ("id")

);

CREATE TABLE "Rooms" (

/\* Attributes \*/

"id" integer NOT NULL,

"type" varchar(32) NOT NULL,

"number" integer NOT NULL,

/\* Attributes constraints \*/

CONSTRAINT "type\_check" CHECK ( -- Допустимые значения

"type" IN (

'Учебная аудитория'

, 'Преподавательская'

)),

CONSTRAINT "number\_interval" CHECK ( -- Number in range [170, 175]

170 <= "number" AND "number" <= 175),

/\* Foreign keys \*/

/\* Primary key \*/

CONSTRAINT "Rooms\_pk" PRIMARY KEY ("id")

);

CREATE TABLE "Timetable" (

/\* Attributes \*/

"id" integer NOT NULL,

"start" time NOT NULL,

"end" time NOT NULL,

"dow" integer NOT NULL,

/\* Attributes constraints \*/

CONSTRAINT "time\_check" CHECK ( -- Start must be erlier than end

"start" < "end"),

CONSTRAINT "dow\_check" CHECK ( -- Day of week in [1-6] -- Monday to Saturday

1 <= "dow" AND "dow" <= 6),

/\* Foreign keys \*/

"id\_courses" integer NOT NULL,

"id\_rooms" integer NOT NULL,

"id\_teachers" integer NOT NULL,

/\* Primary key \*/

CONSTRAINT "Timetable\_pk" PRIMARY KEY ("id")

);

CREATE TABLE "Documents" (

/\* Attributes \*/

"id" integer NOT NULL,

"type" varchar(64) NOT NULL,

"number" varchar(64) NOT NULL,

/\* Attributes constraints \*/

CONSTRAINT "type\_check" CHECK ( -- Допустимые значения

"type" IN (

'Паспорт'

, 'Свидетельство о рождении'

, 'Диплом'

)),

CONSTRAINT "number\_check" CHECK ( -- Russian alphabet, latin alphabet, numbers, space and '-'. Length >= 3

"number" ~ '^([а-я]|[А-Я]|[a-z]|[A-Z]|[0-9]|[ -]){3,}$'),

/\* Foreign keys \*/

"id\_students" integer DEFAULT NULL,

"id\_teachers" integer DEFAULT NULL,

/\* Foreign keys constraints \*/

CONSTRAINT "only\_one\_id\_check" CHECK ( -- Check if only one is set

"id\_students" IS NULL OR "id\_teachers" IS NULL),

/\* Primary key \*/

CONSTRAINT "Documents\_pk" PRIMARY KEY ("id")

);

CREATE TABLE "Jobs" (

/\* Attributes \*/

"id" integer NOT NULL,

"name" varchar(128) NOT NULL,

/\* Attributes constraints \*/

CONSTRAINT "name\_check" CHECK ( -- Russian alphabet, space and '-'. Length >= 3

"name" ~ '^([а-я]|[А-Я]|[ -]){3,}$'),

/\* Foreign keys \*/

"id\_teachers" integer NOT NULL,

/\* Primary key \*/

CONSTRAINT "Jobs\_pk" PRIMARY KEY ("id")

);

CREATE TABLE "Parents" (

/\* Attributes \*/

"id" integer NOT NULL,

"name" varchar(128) NOT NULL,

/\* Attributes constraints \*/

CONSTRAINT "name\_check" CHECK ( -- Russian alphabet, space and '-'. Length >= 3

"name" ~ '^([а-я]|[А-Я]|[ -]){3,}$'),

/\* Foreign keys \*/

"id\_students" integer NOT NULL,

/\* Primary key \*/

CONSTRAINT "Parents\_pk" PRIMARY KEY ("id")

);

CREATE TABLE "Equipment" (

/\* Attributes \*/

"id" integer NOT NULL,

"name" varchar(128) NOT NULL,

"number" integer NOT NULL, -- Количество, шт

/\* Attributes constraints \*/

CONSTRAINT "name\_check" CHECK ( -- Russian alphabet, space and '-'. Length >= 2

"name" ~ '^([а-я]|[А-Я]|[ -]){2,}$'),

CONSTRAINT "number\_interval" CHECK ( -- Number of items in range [0, inf)

0 <= "number"),

/\* Foreign keys \*/

/\* Primary key \*/

CONSTRAINT "Equipment\_pk" PRIMARY KEY ("id")

);

/\* Intermediate tables \*/

CREATE TABLE "Courses\_MM\_Equipment" (

/\* Foreign keys \*/

"id\_courses" integer NOT NULL,

"id\_equipment" integer NOT NULL,

CONSTRAINT "C\_mm\_E\_fk0" FOREIGN KEY ("id\_courses") REFERENCES "Courses"("id"),

CONSTRAINT "C\_mm\_E\_fk1" FOREIGN KEY ("id\_equipment") REFERENCES "Equipment"("id")

);

CREATE TABLE "Rooms\_MM\_Equipment" (

/\* Foreign keys \*/

"id\_rooms" integer NOT NULL,

"id\_equipment" integer NOT NULL,

CONSTRAINT "C\_mm\_E\_fk0" FOREIGN KEY ("id\_rooms") REFERENCES "Rooms"("id"),

CONSTRAINT "C\_mm\_E\_fk1" FOREIGN KEY ("id\_equipment") REFERENCES "Equipment"("id")

);

CREATE TABLE "Courses\_MM\_Courses" ( -- Предшествующие курсы

/\* Foreign keys \*/

"id\_courses\_cur" integer NOT NULL,

"id\_courses\_prev" integer NOT NULL,

CONSTRAINT "C\_mm\_С\_fk0" FOREIGN KEY ("id\_courses\_cur") REFERENCES "Courses"("id"),

CONSTRAINT "C\_mm\_С\_fk1" FOREIGN KEY ("id\_courses\_prev") REFERENCES "Courses"("id")

);

CREATE TABLE "Teachers\_MM\_Courses" (

/\* Foreign keys \*/

"id\_teachers" integer NOT NULL,

"id\_courses" integer NOT NULL,

CONSTRAINT "C\_mm\_С\_fk0" FOREIGN KEY ("id\_teachers") REFERENCES "Teachers"("id"),

CONSTRAINT "C\_mm\_С\_fk1" FOREIGN KEY ("id\_courses") REFERENCES "Courses"("id")

);

CREATE TABLE "Students\_MM\_Courses" (

/\* Foreign keys \*/

"id\_students" integer NOT NULL,

"id\_courses" integer NOT NULL,

CONSTRAINT "C\_mm\_С\_fk0" FOREIGN KEY ("id\_students") REFERENCES "Students"("id"),

CONSTRAINT "C\_mm\_С\_fk1" FOREIGN KEY ("id\_courses") REFERENCES "Courses"("id")

);

/\* Foreign keys \*/

/\* Main tables \*/

ALTER TABLE "Gradebook" ADD CONSTRAINT "Gradebook\_fk0" FOREIGN KEY ("id\_courses") REFERENCES "Courses"("id");

ALTER TABLE "Gradebook" ADD CONSTRAINT "Gradebook\_fk1" FOREIGN KEY ("id\_students") REFERENCES "Students"("id");

ALTER TABLE "Gradebook" ADD CONSTRAINT "Gradebook\_fk2" FOREIGN KEY ("id\_teachers") REFERENCES "Teachers"("id");

ALTER TABLE "Timetable" ADD CONSTRAINT "Timetable\_fk0" FOREIGN KEY ("id\_courses") REFERENCES "Courses"("id");

ALTER TABLE "Timetable" ADD CONSTRAINT "Timetable\_fk1" FOREIGN KEY ("id\_rooms") REFERENCES "Rooms"("id");

ALTER TABLE "Timetable" ADD CONSTRAINT "Timetable\_fk2" FOREIGN KEY ("id\_teachers") REFERENCES "Teachers"("id");

ALTER TABLE "Documents" ADD CONSTRAINT "Documents\_fk0" FOREIGN KEY ("id\_students") REFERENCES "Students"("id");

ALTER TABLE "Documents" ADD CONSTRAINT "Documents\_fk1" FOREIGN KEY ("id\_teachers") REFERENCES "Teachers"("id");

ALTER TABLE "Jobs" ADD CONSTRAINT "Jobs\_fk0" FOREIGN KEY ("id\_teachers") REFERENCES "Teachers"("id");

ALTER TABLE "Parents" ADD CONSTRAINT "Parents\_fk0" FOREIGN KEY ("id\_students") REFERENCES "Students"("id");

/\*

\* Triggers

\*/

/\* Check student's rating \*/

CREATE FUNCTION student\_check() RETURNS TRIGGER AS $student\_check$

DECLARE

"rating\_prev" integer;

BEGIN

IF (SELECT current\_role = 'postgres') THEN

"rating\_prev" := (SELECT "rating" FROM "Students" WHERE "id" = NEW."id");

IF (NEW."rating" < div("rating\_prev", 5) \* 5) THEN

RAISE EXCEPTION 'student\_check(): Student''s rating can''t go lower then the threshold level';

END IF;

END IF;

RETURN NEW;

END

$student\_check$ LANGUAGE plpgsql;

/\* Check the price of previous courses \*/

CREATE FUNCTION previous\_price\_check() RETURNS TRIGGER AS $previous\_price\_check$

DECLARE

"price\_cur" integer;

"price\_prev" integer;

BEGIN

IF (SELECT current\_role = 'postgres') THEN

"price\_cur" := (SELECT "price" FROM "Courses" WHERE "id" = NEW."id\_courses\_cur");

"price\_prev" := (SELECT "price" FROM "Courses" WHERE "id" = NEW."id\_courses\_prev");

IF "price\_cur" < "price\_prev" THEN

RAISE EXCEPTION 'previous\_price\_check(): No course can be cheaper that any of it''s previous course';

END IF;

END IF;

RETURN NEW;

END

$previous\_price\_check$ LANGUAGE plpgsql;

/\* Creation \*/

CREATE TRIGGER student\_check BEFORE UPDATE ON "Students" FOR EACH ROW EXECUTE PROCEDURE student\_check();

CREATE TRIGGER previous\_price\_check BEFORE INSERT ON "Courses\_MM\_Courses" FOR EACH ROW EXECUTE PROCEDURE previous\_price\_check();

Приложение 3 – Реализация политики разграничения доступа

/\*

\* Users and Security Policy

\*/

/\*

\* Initial drop

\*/

DROP ROLE IF EXISTS role\_administrator;

DROP ROLE IF EXISTS role\_student;

DROP USER IF EXISTS ivankov;

DROP USER IF EXISTS \_dementyeva;

DROP USER IF EXISTS \_newstudent;

DROP POLICY IF EXISTS policy\_administrator ON "Teachers";

DROP POLICY IF EXISTS policy\_administrator ON "Courses";

DROP POLICY IF EXISTS policy\_administrator ON "Teachers\_MM\_Courses";

DROP POLICY IF EXISTS policy\_student ON "Students";

DROP POLICY IF EXISTS policy\_student ON "Documents";

DROP POLICY IF EXISTS policy\_student ON "Parents";

DROP FUNCTION IF EXISTS get\_price\_avg\_by\_teacher();

DROP FUNCTION IF EXISTS get\_price\_min\_by\_teacher();

DROP FUNCTION IF EXISTS get\_price\_max\_by\_teacher();

/\*

\* Creation of roles and users

\*/

CREATE ROLE role\_administrator;

CREATE ROLE role\_student;

CREATE USER ivankov;

CREATE USER \_dementyeva;

CREATE USER \_newstudent;

GRANT role\_administrator TO ivankov;

GRANT role\_student TO \_dementyeva;

GRANT role\_student TO \_newstudent;

/\*

\* Policy

\*/

ALTER TABLE "Teachers" ENABLE ROW LEVEL SECURITY;

ALTER TABLE "Courses" ENABLE ROW LEVEL SECURITY;

ALTER TABLE "Teachers\_MM\_Courses" ENABLE ROW LEVEL SECURITY;

ALTER TABLE "Students" ENABLE ROW LEVEL SECURITY;

ALTER TABLE "Documents" ENABLE ROW LEVEL SECURITY;

ALTER TABLE "Parents" ENABLE ROW LEVEL SECURITY;

/\*

\* role\_administrator

\*/

CREATE POLICY policy\_administrator ON "Teachers"

TO role\_administrator

USING (TRUE);

CREATE POLICY policy\_administrator ON "Courses"

TO role\_administrator

USING (TRUE);

CREATE POLICY policy\_administrator ON "Teachers\_MM\_Courses"

TO role\_administrator

USING (TRUE);

GRANT SELECT (

"id"

, "name"

) ON "Teachers"

TO role\_administrator;

GRANT SELECT (

"id"

, "price"

) ON "Courses"

TO role\_administrator;

GRANT SELECT (

"id\_teachers"

, "id\_courses"

) ON "Teachers\_MM\_Courses"

TO role\_administrator;

/\* Average price \*/

CREATE FUNCTION get\_price\_avg\_by\_teacher()

RETURNS TABLE(

"price\_avg" FLOAT

, "teacher" varchar(128)

) AS $$

SELECT

AVG("Courses"."price") AS "price\_avg"

, "Teachers"."name" AS "teacher"

FROM

"Teachers\_MM\_Courses"

, "Courses"

, "Teachers"

WHERE (

"Teachers\_MM\_Courses"."id\_courses" = "Courses"."id"

AND "Teachers\_MM\_Courses"."id\_teachers" = "Teachers"."id"

) GROUP BY "teacher"

$$

LANGUAGE SQL;

/\* Minimal price \*/

CREATE FUNCTION get\_price\_min\_by\_teacher()

RETURNS TABLE(

"price\_min" FLOAT

, "teacher" varchar(128)

) AS $$

SELECT

MIN("Courses"."price") AS "price\_min"

, "Teachers"."name" AS "teacher"

FROM

"Teachers\_MM\_Courses"

, "Courses"

, "Teachers"

WHERE (

"Teachers\_MM\_Courses"."id\_courses" = "Courses"."id"

AND "Teachers\_MM\_Courses"."id\_teachers" = "Teachers"."id"

) GROUP BY "teacher"

$$

LANGUAGE SQL;

/\* Maximal price \*/

CREATE FUNCTION get\_price\_max\_by\_teacher()

RETURNS TABLE(

"price\_max" FLOAT

, "teacher" varchar(128)

) AS $$

SELECT

MAX("Courses"."price") AS "price\_max"

, "Teachers"."name" AS "teacher"

FROM

"Teachers\_MM\_Courses"

, "Courses"

, "Teachers"

WHERE (

"Teachers\_MM\_Courses"."id\_courses" = "Courses"."id"

AND "Teachers\_MM\_Courses"."id\_teachers" = "Teachers"."id"

) GROUP BY "teacher"

$$

LANGUAGE SQL;

/\*

\* role\_student

\*/

CREATE POLICY policy\_student ON "Students"

TO role\_student

USING (current\_user = "Students"."username")

WITH CHECK (current\_user = "Students"."username");

CREATE POLICY policy\_student ON "Documents"

TO role\_student

USING (current\_user = (SELECT "username" FROM "Students" WHERE "id" = "Documents"."id\_students"))

WITH CHECK (current\_user = (SELECT "username" FROM "Students" WHERE "id" = "Documents"."id\_students"));

CREATE POLICY policy\_student ON "Parents"

TO role\_student

USING (current\_user = (SELECT "username" FROM "Students" WHERE "id" = "Parents"."id\_students"));

GRANT

INSERT (

"id"

, "username"

, "name"

),

SELECT (

"id"

, "username"

, "name"

, "majority"

),

UPDATE (

"name"

)

ON "Students"

TO role\_student;

GRANT

INSERT (

"id"

, "type"

, "number"

, "id\_students"

),

SELECT (

"id"

, "type"

, "number"

, "id\_students"

),

UPDATE (

"type"

, "number"

)

ON "Documents"

TO role\_student;

GRANT

SELECT (

"name"

, "id\_students"

)

ON "Parents"

TO role\_student;

/\*

\* Users and Security Policy -- views

\*/

/\*

\* Initial drop

\*/

DROP ROLE IF EXISTS role\_administrator;

DROP ROLE IF EXISTS role\_student;

DROP USER IF EXISTS ivankov;

DROP USER IF EXISTS \_dementyeva;

DROP USER IF EXISTS \_newstudent;

DROP VIEW IF EXISTS view\_student\_info;

--DROP VIEW IF EXISTS view\_student\_info;

/\*

\* Creation of roles and users

\*/

CREATE ROLE role\_administrator;

CREATE ROLE role\_student;

CREATE USER ivankov;

CREATE USER \_dementyeva;

CREATE USER \_newstudent;

GRANT role\_administrator TO ivankov;

GRANT role\_student TO \_dementyeva;

GRANT role\_student TO \_newstudent;

/\*

\* Policy

\*/

/\*

\* role\_administrator

\*/

CREATE VIEW view\_teachers AS

SELECT

"id"

, "name"

FROM "Teachers";

CREATE VIEW view\_courses AS

SELECT

"id"

, "price"

FROM "Courses";

CREATE VIEW view\_teachers\_mm\_courses AS

SELECT

"id\_teachers"

, "id\_courses"

FROM "Teachers\_MM\_Courses";

GRANT

SELECT

ON view\_teachers

TO role\_administrator;

GRANT

SELECT

ON view\_courses

TO role\_administrator;

GRANT

SELECT

ON view\_teachers\_mm\_courses

TO role\_administrator;

/\* Average price \*/

CREATE FUNCTION get\_price\_avg\_by\_teacher()

RETURNS TABLE(

"price\_avg" FLOAT

, "teacher" varchar(128)

) AS $$

SELECT

AVG("view\_courses"."price") AS "price\_avg"

, "view\_teachers"."name" AS "teacher"

FROM

"view\_teachers\_mm\_courses"

, "view\_courses"

, "view\_teachers"

WHERE (

"view\_teachers\_mm\_courses"."id\_courses" = "view\_courses"."id"

AND "view\_teachers\_mm\_courses"."id\_teachers" = "view\_teachers"."id"

) GROUP BY "teacher"

$$

LANGUAGE SQL;

/\* Minimal price \*/

CREATE FUNCTION get\_price\_min\_by\_teacher()

RETURNS TABLE(

"price\_min" FLOAT

, "teacher" varchar(128)

) AS $$

SELECT

MIN("view\_courses"."price") AS "price\_avg"

, "view\_teachers"."name" AS "teacher"

FROM

"view\_teachers\_mm\_courses"

, "view\_courses"

, "view\_teachers"

WHERE (

"view\_teachers\_mm\_courses"."id\_courses" = "view\_courses"."id"

AND "view\_teachers\_mm\_courses"."id\_teachers" = "view\_teachers"."id"

) GROUP BY "teacher"

$$

LANGUAGE SQL;

/\* Maximal price \*/

CREATE FUNCTION get\_price\_max\_by\_teacher()

RETURNS TABLE(

"price\_max" FLOAT

, "teacher" varchar(128)

) AS $$

SELECT

MAX("view\_courses"."price") AS "price\_avg"

, "view\_teachers"."name" AS "teacher"

FROM

"view\_teachers\_mm\_courses"

, "view\_courses"

, "view\_teachers"

WHERE (

"view\_teachers\_mm\_courses"."id\_courses" = "view\_courses"."id"

AND "view\_teachers\_mm\_courses"."id\_teachers" = "view\_teachers"."id"

) GROUP BY "teacher"

$$

LANGUAGE SQL;

/\*

\* role\_student

\*/

CREATE VIEW view\_students AS

SELECT

"id"

, "username"

, "name"

, "majority"

FROM "Students"

WHERE current\_user = "username";

CREATE VIEW view\_documents AS

SELECT

"id"

, "type"

, "number"

, "id\_students"

FROM "Documents"

WHERE current\_user = (SELECT "username" FROM "Students" WHERE "id" = "Documents"."id\_students");

CREATE VIEW view\_parents AS

SELECT

"name"

, "id\_students"

FROM "Parents"

WHERE current\_user = (SELECT "username" FROM "Students" WHERE "id" = "Parents"."id\_students");

GRANT

SELECT,

INSERT (

"id"

, "username"

, "name"

),

UPDATE (

"name"

)

ON view\_students

TO role\_student;

GRANT

SELECT,

INSERT (

"id"

, "type"

, "number"

, "id\_students"

),

UPDATE (

"type"

, "number"

)

ON view\_documents

TO role\_student;

GRANT

SELECT

ON view\_parents

TO role\_student;

Приложение 4 – Заполнение БД данными

/\*

\* Data filling

\*/

/\* Teachers \*/

INSERT INTO "Teachers" ("id", "username", "name", "speciality", "degree", "rating", "salary")

VALUES (1, 'ivankov', 'Иванков Илья Дмитриевич', 'Информационная безопасность', 'Кандидат наук', 79, 112000 );

INSERT INTO "Teachers" ("id", "username", "name", "speciality", "degree", "rating", "salary")

VALUES (2, 'mishin', 'Мишин Даниил Романович', 'Информатика и вычислительная техника', 'Магистр', 46, 60000 );

INSERT INTO "Teachers" ("id", "username", "name", "speciality", "degree", "rating", "salary")

VALUES (3, 'sysoeva', 'Сысоева Лариса Вениаминовна', 'Управление в технических системах', 'Кандидат наук', 72, 109000 );

INSERT INTO "Teachers" ("id", "username", "name", "speciality", "degree", "rating", "salary")

VALUES (5, 'tarasov', 'Тарасов Роман Семенович', 'Математическая логика, алгебра и теория чисел', 'Доктор наук', 99, 200000 );

/\* Courses \*/

INSERT INTO "Courses" ("id", "name", "annotation", "duration", "price" )

VALUES (0, 'Информационная безопасность', 'Программа переподготовки специалистов', 512, 120000 );

INSERT INTO "Courses" ("id", "name", "annotation", "duration", "price" )

VALUES (1, 'Компьютерные сети', 'Введение в компьютерные сети', 103, 40000 );

INSERT INTO "Courses" ("id", "name", "annotation", "duration", "price" )

VALUES (2, 'Системное программирование', 'Разработка системного ПО для ОС Windows', 300, 79000 );

/\* Courses\_MM\_Courses - предшествующие курсы \*/

INSERT INTO "Courses\_MM\_Courses" ("id\_courses\_cur", "id\_courses\_prev") VALUES (0, 1);

INSERT INTO "Courses\_MM\_Courses" ("id\_courses\_cur", "id\_courses\_prev") VALUES (0, 2);

/\* Teachers\_MM\_Courses \*/

INSERT INTO "Teachers\_MM\_Courses" ("id\_teachers", "id\_courses") VALUES (1, 0);

INSERT INTO "Teachers\_MM\_Courses" ("id\_teachers", "id\_courses") VALUES (2, 1);

INSERT INTO "Teachers\_MM\_Courses" ("id\_teachers", "id\_courses") VALUES (3, 2);

INSERT INTO "Teachers\_MM\_Courses" ("id\_teachers", "id\_courses") VALUES (5, 0);

/\* Rooms \*/

INSERT INTO "Rooms" ("id", "type", "number")

VALUES (0, 'Учебная аудитория', 170);

INSERT INTO "Rooms" ("id", "type", "number")

VALUES (1, 'Преподавательская', 171);

INSERT INTO "Rooms" ("id", "type", "number")

VALUES (3, 'Преподавательская', 173);

INSERT INTO "Rooms" ("id", "type", "number")

VALUES (4, 'Учебная аудитория', 174);

INSERT INTO "Rooms" ("id", "type", "number")

VALUES (5, 'Учебная аудитория', 175);

/\* Students \*/

INSERT INTO "Students" ("id", "username", "name", "rating", "majority")

VALUES (0, '\_dementyeva', 'Дементьева Наталия Игоревна', 63, TRUE);

INSERT INTO "Students" ("id", "username", "name", "rating", "majority")

VALUES (1, '\_mihaylov', 'Михайлов Александр Павлович', 44, FALSE);

INSERT INTO "Students" ("id", "username", "name", "rating", "majority")

VALUES (2, '\_shashkov', 'Шашков Семен Андреевич', 23, TRUE);

INSERT INTO "Students" ("id", "username", "name", "rating", "majority")

VALUES (3, '\_avdeeva', 'Авдеева Олеся Святославовна', 79, FALSE);

INSERT INTO "Students" ("id", "username", "name", "rating", "majority")

VALUES (4, '\_semenov', 'Семенов Марк Родионович', 45, FALSE);

INSERT INTO "Students" ("id", "username", "name", "rating", "majority")

VALUES (5, '\_maksimova', 'Максимова Ангелина Петровна', 78, TRUE);

INSERT INTO "Students" ("id", "username", "name", "rating", "majority")

VALUES (6, '\_loginov', 'Логинов Эрик Робертович', 30, FALSE);

INSERT INTO "Students" ("id", "username", "name", "rating", "majority")

VALUES (7, '\_krylov', 'Крылов Алексей Васильевич', 81, TRUE);

/\* Students\_MM\_Courses \*/

INSERT INTO "Students\_MM\_Courses" ("id\_students", "id\_courses") VALUES (0, 1);

INSERT INTO "Students\_MM\_Courses" ("id\_students", "id\_courses") VALUES (0, 2);

INSERT INTO "Students\_MM\_Courses" ("id\_students", "id\_courses") VALUES (1, 2);

INSERT INTO "Students\_MM\_Courses" ("id\_students", "id\_courses") VALUES (2, 0);

INSERT INTO "Students\_MM\_Courses" ("id\_students", "id\_courses") VALUES (3, 0);

INSERT INTO "Students\_MM\_Courses" ("id\_students", "id\_courses") VALUES (4, 0);

INSERT INTO "Students\_MM\_Courses" ("id\_students", "id\_courses") VALUES (5, 0);

INSERT INTO "Students\_MM\_Courses" ("id\_students", "id\_courses") VALUES (6, 0);

INSERT INTO "Students\_MM\_Courses" ("id\_students", "id\_courses") VALUES (7, 0);

/\* Timetable \*/

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (0, '10:00:00', '12:30:00', 1, 0, 0, 1);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (1, '13:00:00', '13:30:00', 1, 0, 4, 1);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (2, '13:00:00', '13:30:00', 1, 0, 5, 5);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (3, '10:00:00', '12:30:00', 3, 0, 0, 1);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (4, '13:00:00', '13:30:00', 3, 0, 4, 5);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (5, '10:00:00', '12:30:00', 4, 0, 0, 5);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (6, '10:00:00', '12:30:00', 4, 0, 4, 1);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (7, '13:00:00', '13:30:00', 4, 0, 5, 5);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (8, '10:00:00', '12:30:00', 1, 1, 4, 2);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (9, '13:00:00', '13:30:00', 1, 1, 5, 2);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (10, '13:00:00', '13:30:00', 1, 1, 0, 2);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (11, '10:00:00', '12:30:00', 3, 1, 5, 2);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (12, '13:00:00', '13:30:00', 3, 1, 5, 2);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (13, '10:00:00', '12:30:00', 4, 1, 4, 2);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (14, '10:00:00', '12:30:00', 4, 1, 5, 2);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (15, '13:00:00', '13:30:00', 4, 1, 0, 2);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (16, '10:00:00', '12:30:00', 1, 2, 4, 3);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (17, '13:00:00', '13:30:00', 1, 2, 0, 3);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (18, '13:00:00', '13:30:00', 1, 2, 5, 3);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (19, '10:00:00', '12:30:00', 3, 2, 4, 3);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (20, '13:00:00', '13:30:00', 3, 2, 0, 3);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (21, '10:00:00', '12:30:00', 4, 2, 5, 3);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (22, '10:00:00', '12:30:00', 4, 2, 0, 3);

INSERT INTO "Timetable" ("id", "start", "end", "dow", "id\_courses", "id\_rooms", "id\_teachers") VALUES (23, '13:00:00', '13:30:00', 4, 2, 4, 3);

/\* Documents \*/

INSERT INTO "Documents" ("id", "type", "number", "id\_teachers") VALUES (0, 'Паспорт', '4004 078567', 1);

INSERT INTO "Documents" ("id", "type", "number", "id\_teachers") VALUES (2, 'Паспорт', '4015 427247', 2);

INSERT INTO "Documents" ("id", "type", "number", "id\_teachers") VALUES (4, 'Паспорт', '4000 934575', 3);

INSERT INTO "Documents" ("id", "type", "number", "id\_teachers") VALUES (6, 'Паспорт', '4003 524157', 5);

INSERT INTO "Documents" ("id", "type", "number", "id\_teachers") VALUES (1, 'Диплом', '473892 7462947', 1);

INSERT INTO "Documents" ("id", "type", "number", "id\_teachers") VALUES (3, 'Диплом', '471548 2874782', 2);

INSERT INTO "Documents" ("id", "type", "number", "id\_teachers") VALUES (5, 'Диплом', '215678 3205704', 3);

INSERT INTO "Documents" ("id", "type", "number", "id\_teachers") VALUES (7, 'Диплом', '393024 7882445', 5);

INSERT INTO "Documents" ("id", "type", "number", "id\_students") VALUES (23, 'Свидетельство о рождении', 'IV-ЖА 837462', 1);

INSERT INTO "Documents" ("id", "type", "number", "id\_students") VALUES (24, 'Свидетельство о рождении', 'XI-ВЫ 643254', 3);

INSERT INTO "Documents" ("id", "type", "number", "id\_students") VALUES (25, 'Свидетельство о рождении', 'XV-МК 867014', 4);

INSERT INTO "Documents" ("id", "type", "number", "id\_students") VALUES (26, 'Свидетельство о рождении', 'II-ЛУ 953135', 6);

INSERT INTO "Documents" ("id", "type", "number", "id\_students") VALUES (29, 'Паспорт', '4018 850788', 0);

INSERT INTO "Documents" ("id", "type", "number", "id\_students") VALUES (32, 'Паспорт', '4017 634234', 2);

INSERT INTO "Documents" ("id", "type", "number", "id\_students") VALUES (33, 'Паспорт', '4019 853456', 5);

INSERT INTO "Documents" ("id", "type", "number", "id\_students") VALUES (34, 'Паспорт', '4020 352465', 7);

/\* Jobs \*/

INSERT INTO "Jobs" ("id", "name", "id\_teachers") VALUES (0, 'ФГАОУ ВО СПбПУ', 3);

INSERT INTO "Jobs" ("id", "name", "id\_teachers") VALUES (1, 'ФГАОУ ВО СПбПУ', 5);

INSERT INTO "Jobs" ("id", "name", "id\_teachers") VALUES (2, 'ООО НеоБИТ', 1);

INSERT INTO "Jobs" ("id", "name", "id\_teachers") VALUES (3, 'АО МЦСТ', 2);

/\* Parents \*/

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (0, 'Дементьев Игорь Матвеевич', 0);

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (1, 'Дементьева Ольга Владиславовна', 0);

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (2, 'Михайлов Павел Антонович', 1);

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (3, 'Михайлова Зоя Васильевна', 1);

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (4, 'Шашков Андрей Артурович', 2);

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (5, 'Шашкова Марта Ивановна', 2);

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (6, 'Авдеев Святослав Богданович', 3);

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (7, 'Авдеева Елена Юрьевна', 3);

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (8, 'Семенов Родион Витальевич', 4);

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (9, 'Семенова Валерия Мартыновна', 4);

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (10, 'Максимов Петр Павлович', 5);

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (11, 'Максимова Ульяна Викторовна', 5);

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (12, 'Логинов Роберт Геннадьевич', 6);

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (13, 'Логинова Кристина Матвеевна', 6);

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (14, 'Крылов Василий Григорьевич', 7);

INSERT INTO "Parents" ("id", "name", "id\_students") VALUES (15, 'Крылова София Антоновна', 7);

/\* Equipment \*/

INSERT INTO "Equipment" ("id", "name", "number") VALUES (0, 'ПК', 1);

INSERT INTO "Equipment" ("id", "name", "number") VALUES (1, 'ПК', 2);

INSERT INTO "Equipment" ("id", "name", "number") VALUES (2, 'ПК', 6);

INSERT INTO "Equipment" ("id", "name", "number") VALUES (3, 'ПК', 4);

INSERT INTO "Equipment" ("id", "name", "number") VALUES (4, 'ПК', 8);

INSERT INTO "Equipment" ("id", "name", "number") VALUES (5, 'Проектор', 1);

INSERT INTO "Equipment" ("id", "name", "number") VALUES (6, 'Учебный коммутатор', 2);

INSERT INTO "Equipment" ("id", "name", "number") VALUES (7, 'Учебный маршрутизатор', 2);

/\* Courses\_MM\_Equipment \*/

INSERT INTO "Courses\_MM\_Equipment" ("id\_courses", "id\_equipment") VALUES (0, 2);

INSERT INTO "Courses\_MM\_Equipment" ("id\_courses", "id\_equipment") VALUES (1, 2);

INSERT INTO "Courses\_MM\_Equipment" ("id\_courses", "id\_equipment") VALUES (1, 6);

INSERT INTO "Courses\_MM\_Equipment" ("id\_courses", "id\_equipment") VALUES (1, 7);

INSERT INTO "Courses\_MM\_Equipment" ("id\_courses", "id\_equipment") VALUES (2, 2);

/\* Rooms\_MM\_Equipment \*/

INSERT INTO "Rooms\_MM\_Equipment" ("id\_rooms", "id\_equipment") VALUES (0, 5);

INSERT INTO "Rooms\_MM\_Equipment" ("id\_rooms", "id\_equipment") VALUES (0, 3);

INSERT INTO "Rooms\_MM\_Equipment" ("id\_rooms", "id\_equipment") VALUES (1, 4);

INSERT INTO "Rooms\_MM\_Equipment" ("id\_rooms", "id\_equipment") VALUES (3, 2);

INSERT INTO "Rooms\_MM\_Equipment" ("id\_rooms", "id\_equipment") VALUES (4, 4);

INSERT INTO "Rooms\_MM\_Equipment" ("id\_rooms", "id\_equipment") VALUES (5, 4);

INSERT INTO "Rooms\_MM\_Equipment" ("id\_rooms", "id\_equipment") VALUES (5, 6);

INSERT INTO "Rooms\_MM\_Equipment" ("id\_rooms", "id\_equipment") VALUES (5, 7);

/\* Gradebook \*/

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (123, 'Занятие', 4, 1, 0, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (125, 'Занятие', 3, 1, 2, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (126, 'Занятие', 4, 1, 3, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (127, 'Занятие', 5, 1, 4, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (128, 'Занятие', 4, 1, 5, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (129, 'Занятие', 5, 1, 6, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (130, 'Занятие', 5, 1, 7, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (131, 'Тест', 5, 1, 0, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (133, 'Тест', 5, 1, 2, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (134, 'Тест', 3, 1, 3, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (135, 'Тест', 5, 1, 4, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (136, 'Тест', 3, 1, 5, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (137, 'Тест', 4, 1, 6, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (138, 'Тест', 3, 1, 7, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (139, 'Лаба', 4, 1, 0, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (141, 'Лаба', 5, 1, 2, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (142, 'Лаба', 4, 1, 3, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (143, 'Лаба', 3, 1, 4, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (144, 'Лаба', 4, 1, 5, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (145, 'Лаба', 3, 1, 6, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (146, 'Лаба', 3, 1, 7, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (147, 'Занятие', 4, 1, 0, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (149, 'Занятие', 5, 1, 2, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (150, 'Занятие', 5, 1, 3, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (151, 'Занятие', 3, 1, 4, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (152, 'Занятие', 3, 1, 5, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (153, 'Занятие', 5, 1, 6, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (154, 'Занятие', 5, 1, 7, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (155, 'Занятие', 3, 1, 0, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (157, 'Занятие', 4, 1, 2, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (158, 'Занятие', 3, 1, 3, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (159, 'Занятие', 5, 1, 4, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (160, 'Занятие', 3, 1, 5, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (161, 'Тест', 4, 1, 0, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (163, 'Тест', 4, 1, 2, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (164, 'Тест', 4, 1, 3, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (165, 'Тест', 5, 1, 4, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (166, 'Тест', 5, 1, 5, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (167, 'Тест', 3, 1, 6, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (168, 'Тест', 5, 1, 7, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (169, 'Лаба', 4, 1, 0, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (171, 'Лаба', 3, 1, 2, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (172, 'Лаба', 5, 1, 3, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (173, 'Лаба', 3, 1, 4, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (174, 'Лаба', 5, 1, 5, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (175, 'Лаба', 5, 1, 6, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (176, 'Лаба', 5, 1, 7, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (177, 'Занятие', 5, 1, 0, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (179, 'Занятие', 3, 1, 2, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (180, 'Занятие', 4, 1, 3, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (181, 'Занятие', 5, 1, 4, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (182, 'Итог', 3, 1, 0, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (184, 'Итог', 5, 1, 2, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (185, 'Итог', 5, 1, 3, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (186, 'Итог', 4, 1, 4, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (187, 'Итог', 3, 1, 5, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (188, 'Итог', 4, 1, 6, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (189, 'Итог', 5, 1, 7, 2);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (190, 'Занятие', 4, 2, 4, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (191, 'Занятие', 4, 2, 3, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (192, 'Занятие', 3, 2, 1, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (193, 'Занятие', 4, 2, 6, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (194, 'Занятие', 5, 2, 2, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (195, 'Занятие', 4, 2, 7, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (196, 'Занятие', 5, 2, 0, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (197, 'Занятие', 5, 2, 5, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (198, 'Тест', 5, 2, 4, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (199, 'Тест', 3, 2, 3, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (200, 'Тест', 5, 2, 1, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (201, 'Тест', 3, 2, 6, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (202, 'Тест', 5, 2, 2, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (203, 'Тест', 3, 2, 7, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (204, 'Тест', 4, 2, 0, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (205, 'Тест', 3, 2, 5, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (206, 'Лаба', 4, 2, 4, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (207, 'Лаба', 3, 2, 3, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (208, 'Лаба', 5, 2, 1, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (209, 'Лаба', 4, 2, 6, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (210, 'Лаба', 3, 2, 2, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (211, 'Лаба', 4, 2, 7, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (212, 'Лаба', 3, 2, 0, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (213, 'Лаба', 3, 2, 5, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (214, 'Занятие', 4, 2, 4, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (215, 'Занятие', 5, 2, 3, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (216, 'Занятие', 5, 2, 1, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (217, 'Занятие', 5, 2, 6, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (218, 'Занятие', 3, 2, 2, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (219, 'Занятие', 3, 2, 7, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (220, 'Занятие', 5, 2, 0, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (221, 'Занятие', 5, 2, 5, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (222, 'Занятие', 3, 2, 4, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (223, 'Занятие', 3, 2, 3, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (224, 'Занятие', 4, 2, 1, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (225, 'Занятие', 3, 2, 6, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (226, 'Занятие', 5, 2, 2, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (227, 'Занятие', 3, 2, 7, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (228, 'Тест', 4, 2, 4, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (229, 'Тест', 5, 2, 3, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (230, 'Тест', 4, 2, 1, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (231, 'Тест', 4, 2, 6, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (232, 'Тест', 5, 2, 2, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (233, 'Тест', 5, 2, 7, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (234, 'Тест', 3, 2, 0, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (235, 'Тест', 5, 2, 5, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (236, 'Лаба', 4, 2, 4, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (237, 'Лаба', 3, 2, 3, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (238, 'Лаба', 3, 2, 1, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (239, 'Лаба', 5, 2, 6, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (240, 'Лаба', 3, 2, 2, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (241, 'Лаба', 5, 2, 7, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (242, 'Лаба', 5, 2, 0, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (243, 'Лаба', 5, 2, 5, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (244, 'Занятие', 5, 2, 4, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (245, 'Занятие', 4, 2, 3, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (246, 'Занятие', 3, 2, 1, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (247, 'Занятие', 4, 2, 6, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (248, 'Занятие', 5, 2, 2, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (249, 'Итог', 3, 2, 4, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (250, 'Итог', 4, 2, 3, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (251, 'Итог', 5, 2, 1, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (252, 'Итог', 5, 2, 6, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (253, 'Итог', 4, 2, 2, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (254, 'Итог', 3, 2, 7, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (255, 'Итог', 4, 2, 0, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (256, 'Итог', 5, 2, 5, 3);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (257, 'Занятие', 4, 0, 7, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (259, 'Занятие', 3, 0, 2, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (261, 'Занятие', 5, 0, 3, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (262, 'Занятие', 4, 0, 4, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (263, 'Занятие', 5, 0, 5, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (264, 'Занятие', 5, 0, 6, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (265, 'Тест', 5, 0, 7, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (267, 'Тест', 5, 0, 2, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (269, 'Тест', 5, 0, 3, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (270, 'Тест', 3, 0, 4, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (271, 'Тест', 4, 0, 5, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (272, 'Тест', 3, 0, 6, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (273, 'Лаба', 4, 0, 7, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (275, 'Лаба', 5, 0, 2, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (277, 'Лаба', 3, 0, 3, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (278, 'Лаба', 4, 0, 4, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (279, 'Лаба', 3, 0, 5, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (280, 'Лаба', 3, 0, 6, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (281, 'Занятие', 4, 0, 7, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (283, 'Занятие', 5, 0, 2, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (285, 'Занятие', 3, 0, 3, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (286, 'Занятие', 3, 0, 4, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (287, 'Занятие', 5, 0, 5, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (288, 'Занятие', 5, 0, 6, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (289, 'Занятие', 3, 0, 7, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (291, 'Занятие', 4, 0, 2, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (293, 'Занятие', 5, 0, 3, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (294, 'Занятие', 3, 0, 4, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (295, 'Тест', 4, 0, 7, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (297, 'Тест', 4, 0, 2, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (299, 'Тест', 5, 0, 3, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (300, 'Тест', 5, 0, 4, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (301, 'Тест', 3, 0, 5, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (302, 'Тест', 5, 0, 6, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (303, 'Лаба', 4, 0, 7, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (305, 'Лаба', 3, 0, 2, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (307, 'Лаба', 3, 0, 3, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (308, 'Лаба', 5, 0, 4, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (309, 'Лаба', 5, 0, 5, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (310, 'Лаба', 5, 0, 6, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (311, 'Занятие', 5, 0, 7, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (313, 'Занятие', 3, 0, 2, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (315, 'Занятие', 5, 0, 3, 5);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (316, 'Итог', 3, 0, 7, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (318, 'Итог', 5, 0, 2, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (320, 'Итог', 4, 0, 3, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (321, 'Итог', 3, 0, 4, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (322, 'Итог', 4, 0, 5, 1);

INSERT INTO "Gradebook" ("id", "type", "score", "id\_courses", "id\_students", "id\_teachers") VALUES (323, 'Итог', 5, 0, 6, 1);