ГУАП

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| старший преподаватель |  |  |  | Н.В. Путилова |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №3 |
| Cоздание и модификация базы данных и таблиц базы данных |
| по курсу: Проектирование баз данных |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4134к |  |  |  | Д. В. Самарин |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2023

1. **Цель работы**

Научится создавать базы данных и составлять SQL запросы.

1. **Задание на лабораторную работу**

В соответствии с моделью, разработанной в предыдущей работе, создать базу данных. Продемонстрировать умение добавить и удалить столбец командой ALTER TABLE.

Школьные экскурсии: тип (развлекательная/образовательная), дисциплины к которым имеет отношение образовательная экскурсия, стоимость с человека, список участников, ответственный за проведение учитель

а. экскурсии на верфи (слово «верфь» в любом месте названия)

б. экскурсии, относящиеся к биологии и географии

в. учителя, не отвечающие за экскурсии

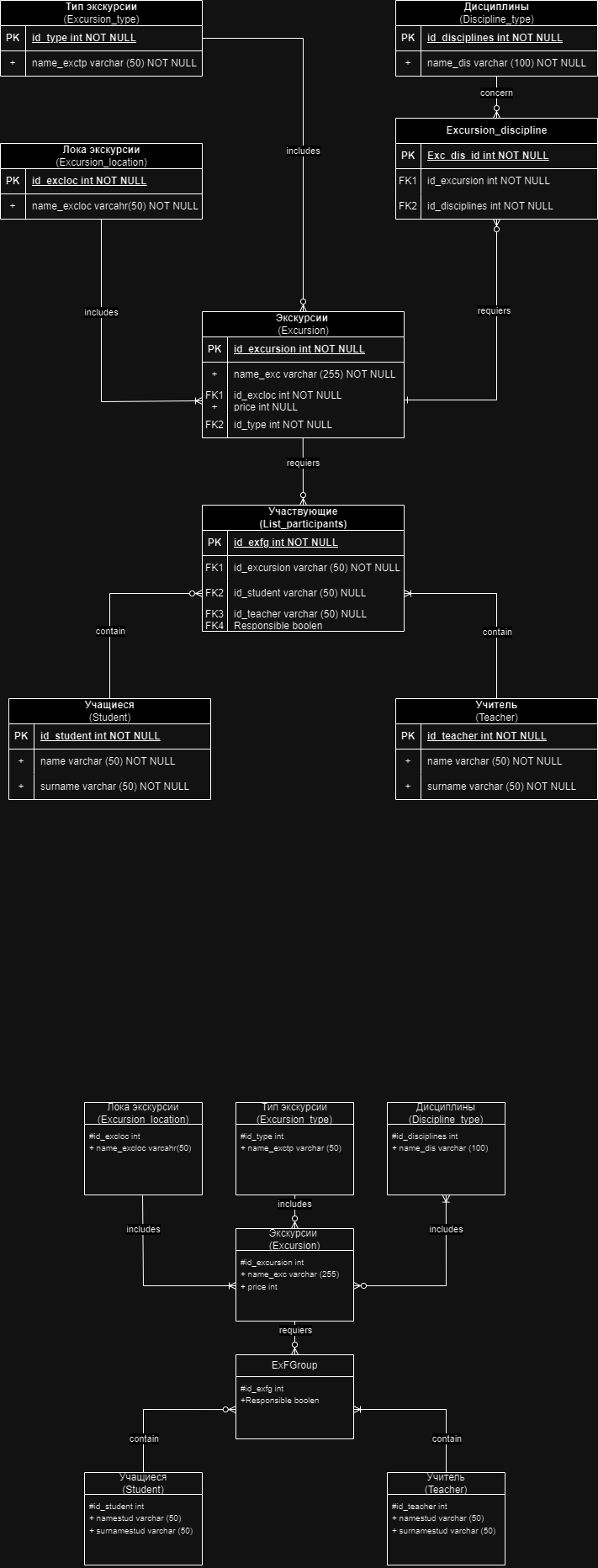
г. учащиеся, которые не ездили в музей истории религии, но ездили в Эрмитаж

д. учитель, отвечавший за число экскурсий больше среднего

е. учителя, отвечающие за самые дорогие экскурсии

ж. Экскурсии, на которые ездили все учащиеся

1. **Физическая модель**



1. **SQL создания базы данных**

CREATE TABLE discipline\_type

(

discipline\_id integer NOT NULL,

discipline\_name character varying(100) COLLATE pg\_catalog."default" NOT NULL,

CONSTRAINT discipline\_type\_pkey PRIMARY KEY (discipline\_id)

)

CREATE TABLE excursion\_location

(

excursion\_location\_id integer NOT NULL,

excursion\_location\_name character varying(50) COLLATE pg\_catalog."default" NOT NULL,

CONSTRAINT excursion\_location\_pkey PRIMARY KEY (excursion\_location\_id)

)

CREATE TABLE excursions\_type

(

excursion\_type\_id integer NOT NULL,

excursion\_type\_name character varying(50) COLLATE pg\_catalog."default" NOT NULL,

CONSTRAINT excursion\_type\_pkey PRIMARY KEY (excursion\_type\_id)

)

CREATE TABLE excursion

(

excursion\_id integer NOT NULL,

excursion\_name character varying(255) COLLATE pg\_catalog."default" NOT NULL,

excursion\_location\_id integer NOT NULL,

price integer,

excursion\_type\_id integer NOT NULL,

CONSTRAINT excursion\_pkey PRIMARY KEY (excursion\_id)

CONSTRAINT fk\_excursion\_locid FOREIGN KEY (excursion\_location\_id)

REFERENCES public.excursion\_loc (excursion\_location\_id) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

CONSTRAINT fk\_excursion\_type FOREIGN KEY (excursion\_type\_id)

REFERENCES public.excursions\_type (excursion\_type\_id) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE RESTRICT

)

CREATE TABLE excursion\_discipline

(

excursion\_discipline\_id integer NOT NULL,

excursion\_id integer NOT NULL,

discipline\_id integer NOT NULL,

CONSTRAINT excursion\_discipline\_pkey PRIMARY KEY (excursion\_discipline\_id),

CONSTRAINT fk\_discipline FOREIGN KEY (discipline\_id)

REFERENCES public.discipline\_type (discipline\_id) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE RESTRICT,

CONSTRAINT fk\_excusrion\_discipline FOREIGN KEY (excursion\_id)

REFERENCES public.excursion (excursion\_id) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE RESTRICT

)

CREATE TABLE student

(

student\_id integer NOT NULL,

student\_name character varying(50) COLLATE pg\_catalog."default" NOT NULL,

student\_Secondname character varying(50) COLLATE pg\_catalog."default" NOT NULL,

CONSTRAINT student\_pkey PRIMARY KEY (student\_id)

)

CREATE TABLE teacher

(

teacher\_id integer NOT NULL,

teacher\_name character varying(50) COLLATE pg\_catalog."default" NOT NULL,

teacher\_Secondname character varying(50) COLLATE pg\_catalog."default" NOT NULL,

CONSTRAINT teacher\_pkey PRIMARY KEY (teacher\_id)

)

CREATE TABLE List\_of\_participants

(

List\_of\_particip\_id integer NOT NULL,

excursion\_id integer NOT NULL,

student\_id integer,

teacher\_id integer,

isAvailable BOOLEAN,

CONSTRAINT List\_of\_participants\_pkey PRIMARY KEY (List\_of\_particip\_id),

CONSTRAINT fk\_excursion FOREIGN KEY (excursion\_id)

REFERENCES public.excursion (excursion\_id) MATCH SIMPLE

ON UPDATE RESTRICT

ON DELETE CASCADE,

CONSTRAINT fk\_student FOREIGN KEY (student\_id)

REFERENCES public.student (student\_id) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE,

CONSTRAINT fk\_teacher FOREIGN KEY (teacher\_id)

REFERENCES public.teacher (teacher\_id) MATCH SIMPLE

ON UPDATE CASCADE

ON DELETE CASCADE

)

**Примеры с использованием ALTER TABLE**

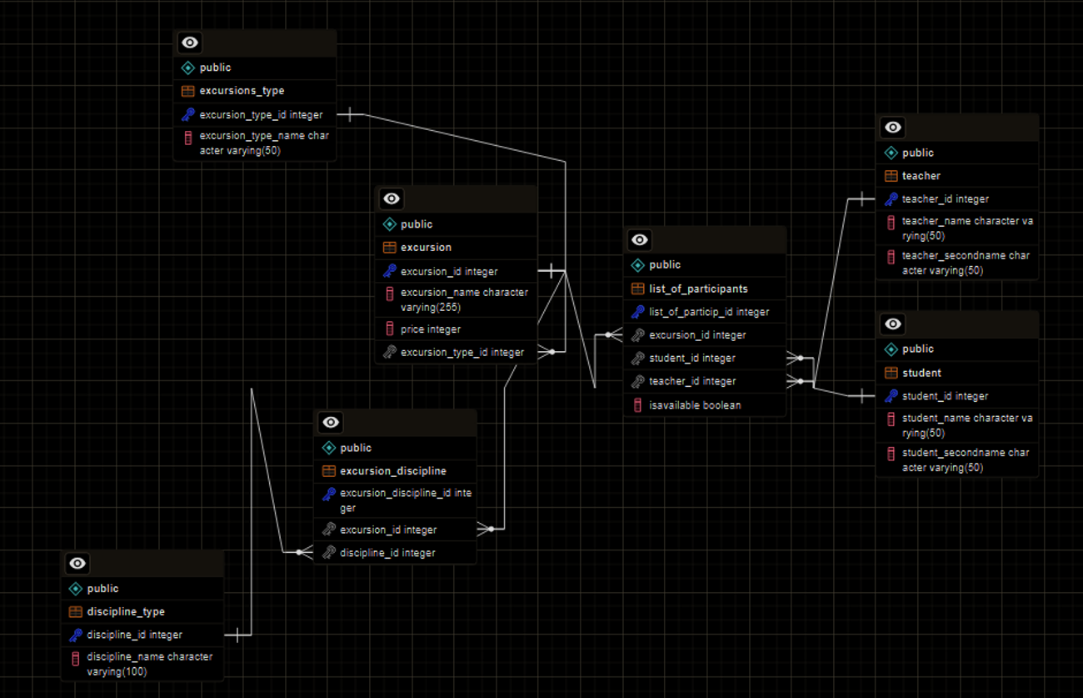
ALTER TABLE student

ADD COLUMN Class\_number VARCHAR(100);

ALTER TABLE student

DROP COLUMN Class\_number;

**EPR представление базы данных на основе выполненного кода (PostgreSQL)**

****

**Вывод**

В ходе лабораторной работы были получены навыки в проектировании баз данных при помощи PostgreSQL.