**Министерство образования и науки Российской Федерации**

**Московский политехнический университет**

**Кафедра информационной безопасности**

**КУРСОВАЯ РАБОТА**

**по дисциплине «Системы управления базами данных»**

на тему

«База данных о учебной литературе»

Выполнил:

студент гр. 201-351

Чернушкин Георгий Александрович

Проверил:

Красников Александр Сергеевич

Москва 2020

УТВЕРЖДАЮ

Зав. кафедрой ИБ

Подпись: / В.Н.Федоров

ЗАДАНИЕ

на курсовую работу по дисциплине «Системы управления базами данных»

студенту Чернушкину Георгию Александровичу 201-352

тема «База данных о учебной литературе»

При выполнении курсовой работы требуется учесть, что необходимо хранить информацию о всех книгах.

Владельцам этих данных может потребоваться информация о точном названии имен писателей, дате издания, краткого описания;

Задание принял:

Чернушкин Георгий Александрович, 26.09.20

Руководитель курсовой работы:

Подпись: / Красников А. С.

**ОГЛАВЛЕНИЕ**

**Описание предметной области……………………………………….………..4**

**Концептуальное проектирование…………...…………………………………4**

**Обоснование выбора СУБД.……………………………………………………5**

**Преобразование концептуальной модели в реляционную модель……….6**

**Нормализация базы данных ..…………………………………………………7**

**Определение характеристик атрибутов.…..………..…………………..……7**

**Процедурная поддержка ограничений целостности……………………….8**

**Реализация операций над данными …………………………….....….……..10**

**Описание предметной области**

Данная база создается для получения и обработки данных о книгах. Работа с базой данных осуществляется посредством выполнения запросов к данной базе. В нашей базе имеется пять основных сущностей, одна из которых для описания книг и их даты выхода: books. Остальные сущности хранят в себе автора книги, краткое описание, к каким категориям IT они относятся и оценку книги с интернет-магазина «Читай Город». Если подробнее описывать, то данная структура представляет дерево, где на первом уровне находится книга и год выхода, на втором - автор, описание, область IT, оценка читателей. Тем самым, создав такую структуры, клиент может выполнять запросы к базе данных для получения данных о книгах из определенной предметной области, определенного автора, по оценке пользователей или по описанию.

**Концептуальное проектирование**

Данная база состоит из пяти сущностей, каждая из которых имеет свои атрибуты. Подробное описание каждой из сущностей представлено ниже.

Author – Книга и ее автор

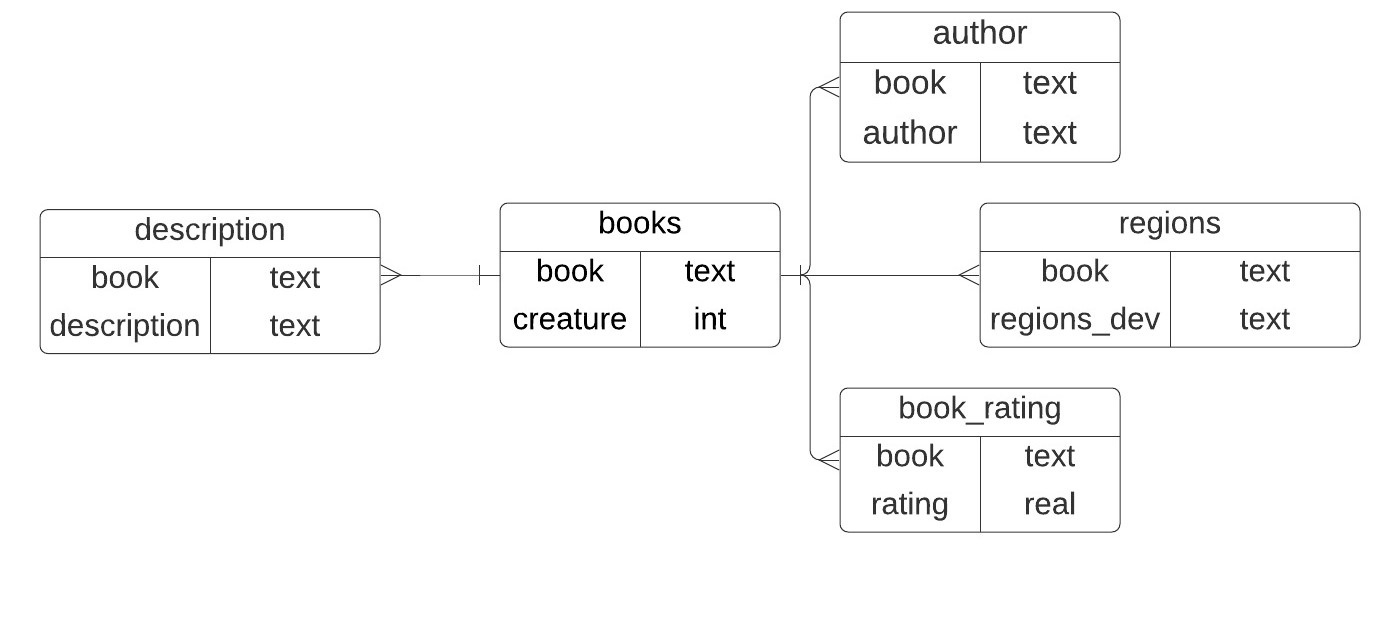
Books – Книга и ее год выхода

Description – Книга и краткое описание

Book\_rating – Книга и ее оценка читателей

Regions – Книга и ее предметные области

На ниже представленное мы можем увидеть все вышеупомянутые сущности в схеме.

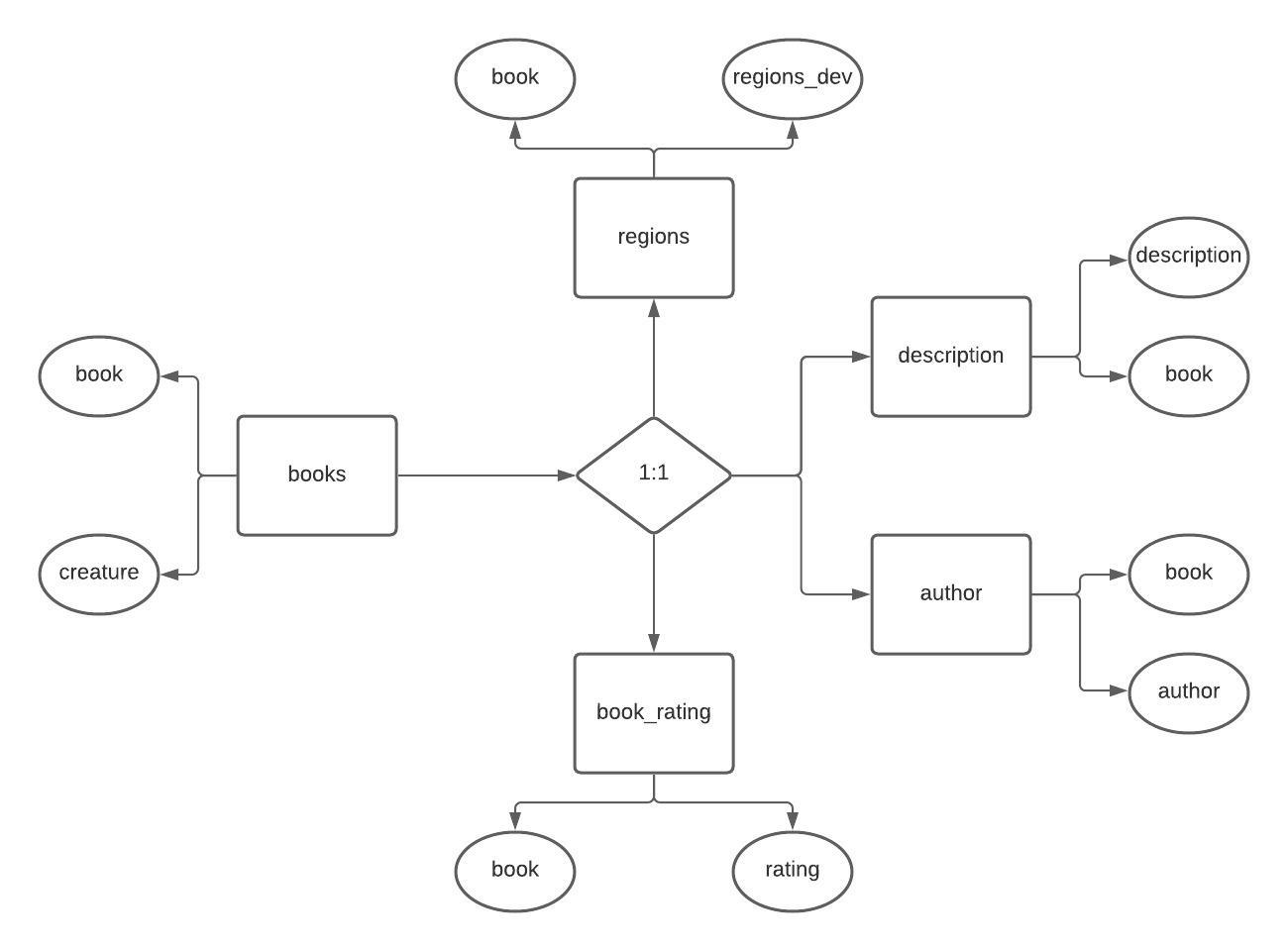


**Обоснование выбора СУБД**

При выборе CУБД учитывались такие факторы, как эффективность и производительность выбранной СУБД, а также фактор того, что данная база разрабатывается обучающимся в рамках программы Основы базы данных. Весь обучающий процесс проходит с использованием PostgreSQL. Исходя из всех вышеперечисленных факторов, было принято решение в качестве главной СУБД использовать PostgreSQL

.

**Преобразование концептуальной модели в реляционную модель**



**Нормализация базы данных**

Архитектура базой данных удовлетворяет всем условиям первой нормальной формы.

**Определение характеристик атрибутов**

Каждая из таблиц в представленной базе данных имеет одинаковые аргументы. Подробное описание всех аргументов представлено ниже:

Book – название книги

Author – автор книги

Regions\_dev – предметная область IT

Description – описание книги

Creature – год выхода книги

Rating – оценки читателей

**АЛГОРИТМ СОЗДАНИЯ ДАННОЙ БАЗЫ ДАННЫХ**

1. Скачать СУБД PostgreSQL.

sudo apt-get install postgresql

1. Зайти в пользователя postgres.

sudo -i -u postgres

1. Создать базу данных командой CREATEDB %name%.
2. Зайти в эту базу.

psql %name%

1. Заполнить базу данных.
2. Выйти из базы.
3. Импортировать таблицы с данными и настройками ключей командой

pg\_dump %name% > Coursework.sql

1. Выполнить переформирование структуры базы данных.

**Процедурная поддержка ограничений целостности**

Ниже представленный запрос создает триггер, который не дает внести книгу, написанную раньше 1900 года.

CREATE FUNCTION public.trigger\_creature() RETURNS trigger

LANGUAGE plpgsql

AS $$

BEGIN

IF TG\_OP = ‘INSERT’ THEN

IF NEW.creature > 1900 THEN

return NEW;

ELSE

return OLD;

END IF;

END IF;

END;

$$;

CREATE TRIGGER stop

BEFORE INSERT OR UPDATE

ON public.books

FOR EACH ROW

EXECUTE FUNCTION public.trigger\_creature();

**Реализация операций над данными**

При работе с данной базой данных необходимо выполнять запросы к ней. Некоторые из запросов, которые могут быть необходимы, приведены ниже:

SELECT \* FROM books;

SELECT \* FROM book\_rating;

SELECT \* FROM author;

SELECT \* FROM description;

SELECT \* FROM regions;

SELECT \* FROM regions WHERE regions\_dev = ‘software’;

SELECT

**СПИСОК ИСТОЧНИКОВ И ЛИТЕРАТУРЫ**

# Документация к Postgres Pro Standard 13.1.1 <https://postgrespro.ru/docs/postgrespro/13/index>

Читай город

<https://www.chitai-gorod.ru>