МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

## ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

**ЛАБОРАТОРНАЯ РАБОТА №1** по дисциплине

«ИНФОРМАЦИОННЫЕ СИСТЕМЫ И БАЗЫ ДАННЫХ»

Вариант № 336805

***Выполнил:***

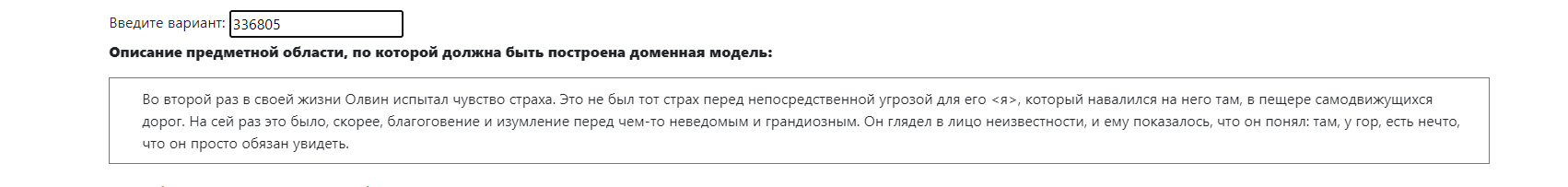
Студент группы P33312

Лысенко А.К.

***Преподаватель:***  Николаев В.В.

Санкт-Петербург, 2023

**Текст задания**



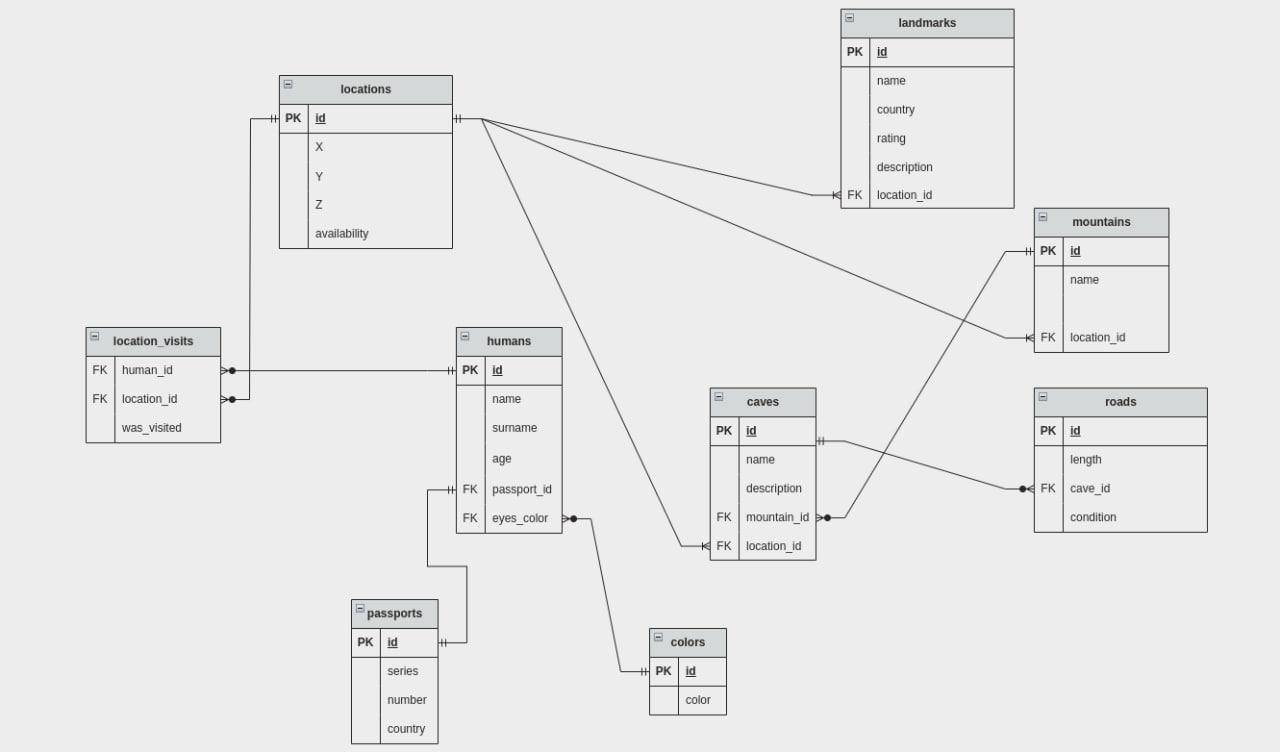
**Описание предметной области**

Все события происходят в мире, похожем на наш. В этом мире есть такие же **люди,** как и мы. У каждого из людей есть **паспорт**, в котором указана **серия**, **номер**, а также **страна**. Каждый человек имеет свое **имя**, **фамилию**, **возраст**, **цвет глаз** и **паспорт**. Помимо этого, в данном мире есть **локации**, которые можно посещать. У этих локаций есть **координаты X, Y, Z** и **доступность**. Этой локацией может быть различное место. Например, в этом мире есть **достопримечательность,** у которой есть **название**, **страна**, **рейтинг**, **описание**. Также есть горы, у которых есть **название**, **описание**. Также в этих горах могут быть **пещеры**. У этих пещер есть название, описание. Также в этих пещерах могут быть **дороги**, которые имеют **длину** и **состояние**. **Рассказать подробнее**

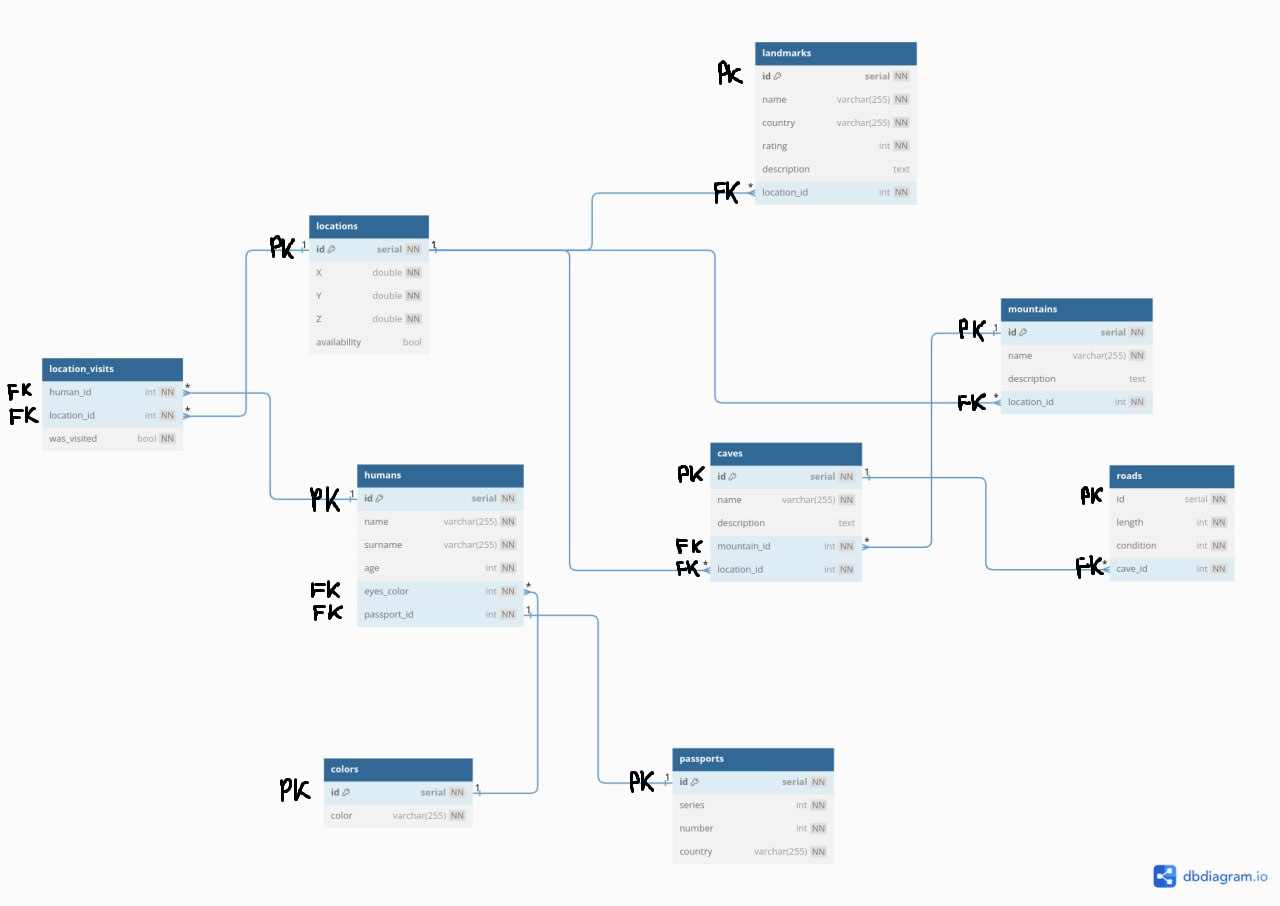
**Список сущностей и их классификация**

* **Стержневые**1) Человек – идентификатор, имя, фамилия, возраст, цвет глаз, идентификатор паспорта.  
  2) Пещера – идентификатор, название, описание, идентификатор горы, идентификатор локации.  
  3) Гора – идентификатор, название, описание, идентификатор локации.  
  4) Достопримечательность – идентификатор, название, страна, описание, идентификатор локации.  
  5) Дорога – протяженность, состояние дороги, идентификатор пещеры, в которой она находится.  
  6) Локация - идентификатор, координаты X, Y, Z, доступность.
* **Ассоциативные**1) Посещение локации – идентификатор человека, идентификатор локации, факт посещения.
* **Характеристические**1) Цвета – идентификатор цвета, название цвета.  
  2) Паспорт – идентификатор паспорта, серия, номер, страна.

**Инфологическая модель**



**Даталогическая модель**



**Реализация даталогической модели на SQL.**

**create table** if **not exists** colors (  
 id serial **primary key not null**,  
 color **varchar**(255) **unique not null check** ( **length**(color) > 0 )  
);  
  
**create table** if **not exists** passports (  
 id serial **primary key not null**,  
 series **int not null unique**,  
 **number int not null unique**,  
 country **varchar**(255) **not null**);  
  
**create table** if **not exists** humans (  
 id serial **primary key not null**,  
 **name varchar**(255) **not null check** ( **length**(**name**) > 0 ),  
 surname **varchar**(255) **not null check** ( **length**(**name**) > 0 ),  
 age **int not null check** ( age > 0 ),  
 eyes\_color **int references** colors(id) **not null** ,  
 passport\_id **int references** passports(id) **not null unique**);  
  
**create table** if **not exists** locations (  
 id serial **primary key not null**,  
 X **double precision not null**,  
 Y **double precision not null**,  
 Z **double precision not null**,  
 availability bool  
);  
  
**create table** if **not exists** location\_visits (  
 human\_id **int references** humans(id) **not null** ,  
 location\_id **int references** locations(id) **not null** ,  
 was\_visited bool **not null**);  
  
  
**create table** if **not exists** landmarks (  
 id serial **primary key not null**,  
 **name varchar**(255) **not null**,  
 country **varchar**(255) **not null**,  
 rating **int not null check** ( rating >= 1 **and** rating <= 5 ),  
 description text,  
 location\_id **int not null references** locations(id)  
);  
  
**create table** if **not exists** mountains (  
 id serial **primary key not null**,  
 **name varchar**(255) **not null check** ( **length**(**name**) > 0 ),  
 description text,  
 location\_id **int not null references** locations(id)  
);  
  
  
**create table** if **not exists** caves (  
 id serial **primary key not null**,  
 **name varchar**(255) **unique not null check** ( **length**(**name**) > 0 ),  
 description text,  
 mountain\_id **int not null references** mountains(id),  
 location\_id **int not null references** locations(id)  
);  
  
**create table** if **not exists** roads (  
 id serial **not null**,  
 **length int not null check** ( **length** > 0 ),  
 **condition int not null check** ( **condition** >= 1 **and condition** <= 5 ),  
 cave\_id **int not null references** caves(id)  
);

**insert into** colors (color) **values** ('green');  
**insert into** colors (color) **values** ('brown');  
**insert into** colors (color) **values** ('black');  
**insert into** colors (color) **values** ('blue');  
  
**insert into** passports (series, **number**, country) **VALUES** (7143, 130831, 'russia');  
**insert into** passports (series, **number**, country) **VALUES** (3141, 73881, 'usa');  
**insert into** passports (series, **number**, country) **VALUES** (1144, 91034, 'brazil');  
**insert into** passports (series, **number**, country) **VALUES** (1381, 1414, 'france');  
  
  
**insert into** humans (**name**, surname, age, eyes\_color, passport\_id) **VALUES** ('Artem', 'Lysenko', 19, 1, 1);  
**insert into** humans (**name**, surname, age, eyes\_color, passport\_id) **VALUES** ('Ivan', 'Petrov', 42, 4, 2);  
**insert into** humans (**name**, surname, age, eyes\_color, passport\_id) **VALUES** ('Leon', 'Maxwell', 9, 1, 4);  
**insert into** humans (**name**, surname, age, eyes\_color, passport\_id) **VALUES** ('Petya', 'Bondar', 24, 1, 3);  
  
**insert into** locations (x, y, z, availability) **VALUES** (3, 5, 9, **true**);  
**insert into** locations (x, y, z, availability) **VALUES** (18.141, 41.41, 41.514, **false**);  
**insert into** locations (x, y, z, availability) **VALUES** (75.14151, 1.515, 43.4151, **null**);  
**insert into** locations (x, y, z, availability) **VALUES** (13.41, 41.531531, 41.114, **true**);  
  
**insert into** location\_visits (human\_id, location\_id, was\_visited) **VALUES** (1, 1, **true**);  
**insert into** location\_visits (human\_id, location\_id, was\_visited) **VALUES** (1, 2, **true**);  
**insert into** location\_visits (human\_id, location\_id, was\_visited) **VALUES** (1, 3, **true**);  
**insert into** location\_visits (human\_id, location\_id, was\_visited) **VALUES** (1, 4, **true**);  
**insert into** location\_visits (human\_id, location\_id, was\_visited) **VALUES** (2, 2, **true**);  
**insert into** location\_visits (human\_id, location\_id, was\_visited) **VALUES** (2, 3, **true**);  
**insert into** location\_visits (human\_id, location\_id, was\_visited) **VALUES** (2, 4, **false**);  
**insert into** location\_visits (human\_id, location\_id, was\_visited) **VALUES** (3, 4, **false**);  
  
**insert into** landmarks (**name**, country, rating, description, location\_id) **VALUES** ('landmark1', 'ru', 4, **null**, 1);  
**insert into** landmarks (**name**, country, rating, description, location\_id) **VALUES** ('landmark2', 'ru', 4, **null**, 2);  
  
**insert into** mountains (**name**, description, location\_id) **VALUES** ('mountain1', 'wow', 3);  
  
**insert into** caves (**name**, description, mountain\_id, location\_id) **VALUES** ('cave1', **null**, 1, 4);  
  
  
**insert into** roads (**length**, **condition**, cave\_id) **VALUES** (141, 3, 1);  
**insert into** roads (**length**, **condition**, cave\_id) **VALUES** (13, 1, 1);  
**insert into** roads (**length**, **condition**, cave\_id) **VALUES** (100, 5, 1);

**Вывод**

В ходе выполнения лабораторной работы я научился различать инфологическую и даталогическую модель данных, научился создавать таблицы в базе данных, добавлять данные в таблице и получать данные из таблиц посредством запросов. Я познакомился с DDL, DML и СУБД PostgreSQL.