

Университет ИТМО

Факультет программной инженерии и компьютерной техники

Лабораторная работа №1

по «Информационным системам и базам данных»

Выполнил:

Студент группы Р32312

Хайкин О.И.

Преподаватель:

Наумова Н. А.

Санкт-Петербург

2023

Текст задания

Для выполнения лабораторной работы №1 необходимо:

- 1 На основе предложенной предметной области (текста) составить ее описание. Из полученного описания выделить сущности, их атрибуты и связи.
- 2 Составить инфологическую модель.
- 3 Составить даталогическую модель. При описании типов данных для атрибутов должны использоваться типы из СУБД PostgreSQL.
- 4 Реализовать даталогическую модель в PostgreSQL. При описании и реализации даталогической модели должны учитываться ограничения целостности, которые характерны для полученной предметной области.
- 5 Заполнить созданные таблицы тестовыми данными.

Для создания объектов базы данных у каждого студента есть своя схема. Название схемы соответствует имени пользователя в базе studs (sXXXXXX). Команда для подключения к базе studs:

```
psql -h pg -d studs
```

Каждый студент должен использовать свою схему при работе над лабораторной работой №1 (а также в рамках выполнения 2, 3 и 4 этапа курсовой работы).

Отчёт по лабораторной работе должен содержать:

- 1 Текст задания.
- 2 Описание предметной области.
- 3 Список сущностей и их классификацию (стержневая, ассоциация, характеристика).
- 4 Инфологическая модель (ER-диаграмма в расширенном виде - с атрибутами, ключами...).
- 5 Даталогическая модель (должна содержать типы атрибутов, вспомогательные таблицы для отображения связей "многие-ко-многим").
- 6 Реализация даталогической модели на SQL.
- 7 Выводы по работе.

Текст варианта

Зато Юпитер надежно удерживал вторую четверку спутников, вдвое более близкую. Орбиты Эллары, Лиситеи, Гималии и Леды похожи и лежат почти в одной плоскости. Существует гипотеза, что они - части одного распавшегося небесного тела; если так, то его диаметр не превышал ста километров.

Описание предметной области

Предметная область состоит из небесных тел солнечной системы. В рамках предметной области будем считать, что небесные объекты могут находиться на орбите вокруг другого небесного объекта (любая планета, любой спутник) или могут не иметь своей орбиты (Солнце).

Орбиты характеризуются своими реальными орбитальными элементами, которые позволяют показать их положение в пространстве.

Кроме небесных тел, их типов и орбит, в предметной области существуют гипотетические небесные тела и гипотезы, их задающие. Гипотеза задаётся своим описанием и списком реальных небесных тел, которые связаны с ней.

Добавим в эту предметную область информацию об открытиях планет и людях, открывших их.

Список сущностей и их классификация

Стержневые сущности

Celestial Body/Небесное тело – представляет одно существующее небесное тело (Земля, Солнце, Юпитер и т. д.)

Атрибуты:

- ID: первичный ключ, идентификатор небесного тела
- Name: название небесного тела
- Mass: масса небесного тела
- Celestial body type ID: вторичный ключ на Тип небесного тела

Celestial body type/Тип небесного тела – представляет тип небесного тела (планета, луна, звезда и т. д.)

Атрибуты:

- ID: первичный ключ, идентификатор типа небесного тела
- Name: сам тип небесного тела

Hypothetical celestial Body/Гипотетическое небесное тело –

представляет одно возможно когда-то существовавшее небесное тело. Вынесено в отдельную от Небесного тела сущность из-за необходимости представлять гипотетические данные о массе, диаметре и подобном.

Атрибуты:

- ID: первичный ключ, идентификатор гипотетического небесного тела
- Name: название гипотетического небесного тела
- Diameter lower bound: нижняя граница диаметра гипотетического небесного тела
- Diameter higher bound: верхняя граница диаметра гипотетического небесного тела
- Mass lower bound: нижняя граница массы гипотетического небесного тела
- Mass higher bound: верхняя граница массы гипотетического небесного тела
- Celestial body type ID: вторичный ключ на Тип небесного тела
- Hypothesis ID: вторичный ключ на гипотезу, задающую это небесное тело

Hypothesis/Гипотеза – представляет гипотезу, связанную с небесными телами.

Атрибуты:

- ID: первичный ключ, идентификатор гипотезы
- Name: название гипотезы
- Description: описание гипотезы
- Related celestial bodies: связанные небесные тела

Human/Человек – представляет человек

Атрибуты:

- ID: первичный ключ, идентификатор человека
- Name: имя человека
- Surname: фамилия человека

- Age: возраст человека
- Gender: гендер человека

Discovery/Открытие – представляет открытие небесного тела

Атрибуты:

- Celestial body ID: первичный ключ, вторичный ключ на Небесное тело, которые было открыто
- Discovery date: дата открытия
- Discovered by: вторичный ключ на человека, совершившего открытие

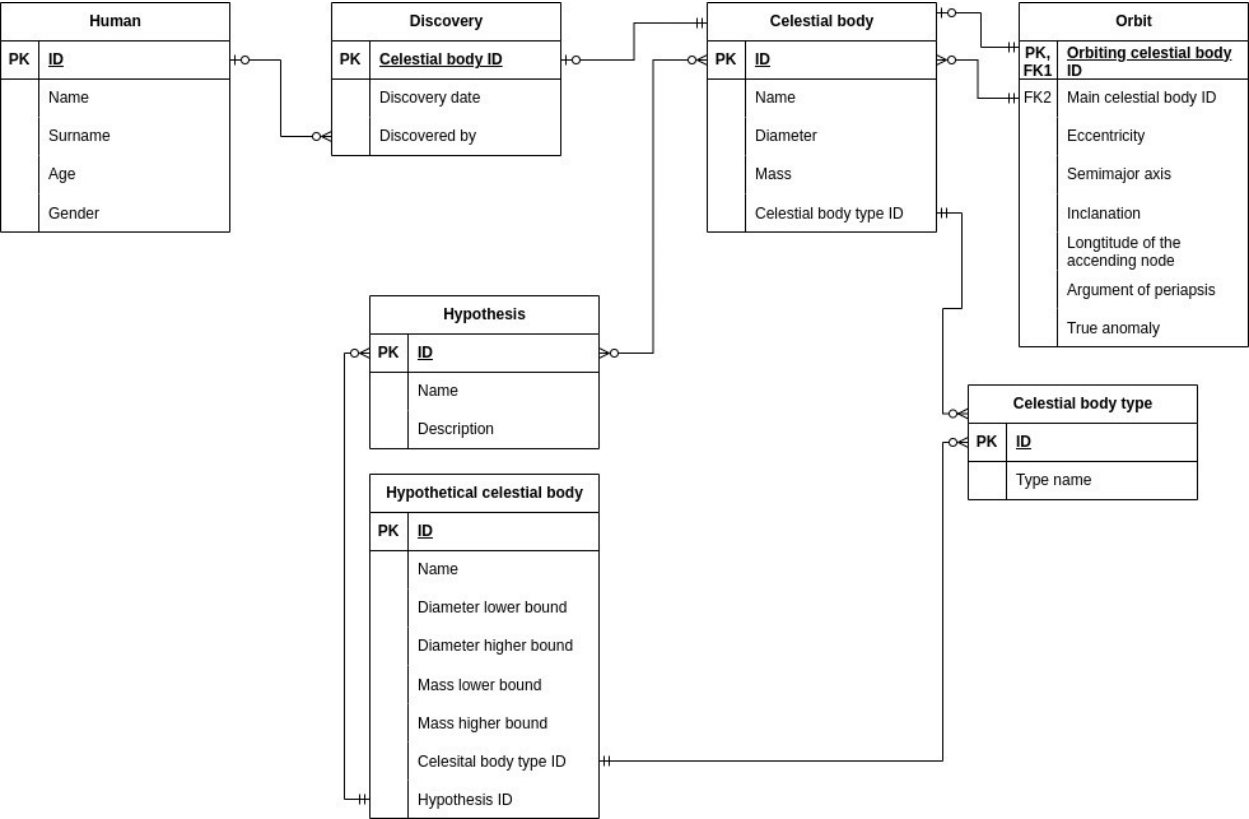
Характеристические сущности

Orbit/Орбита – представляет информацию об орбите небесного тела

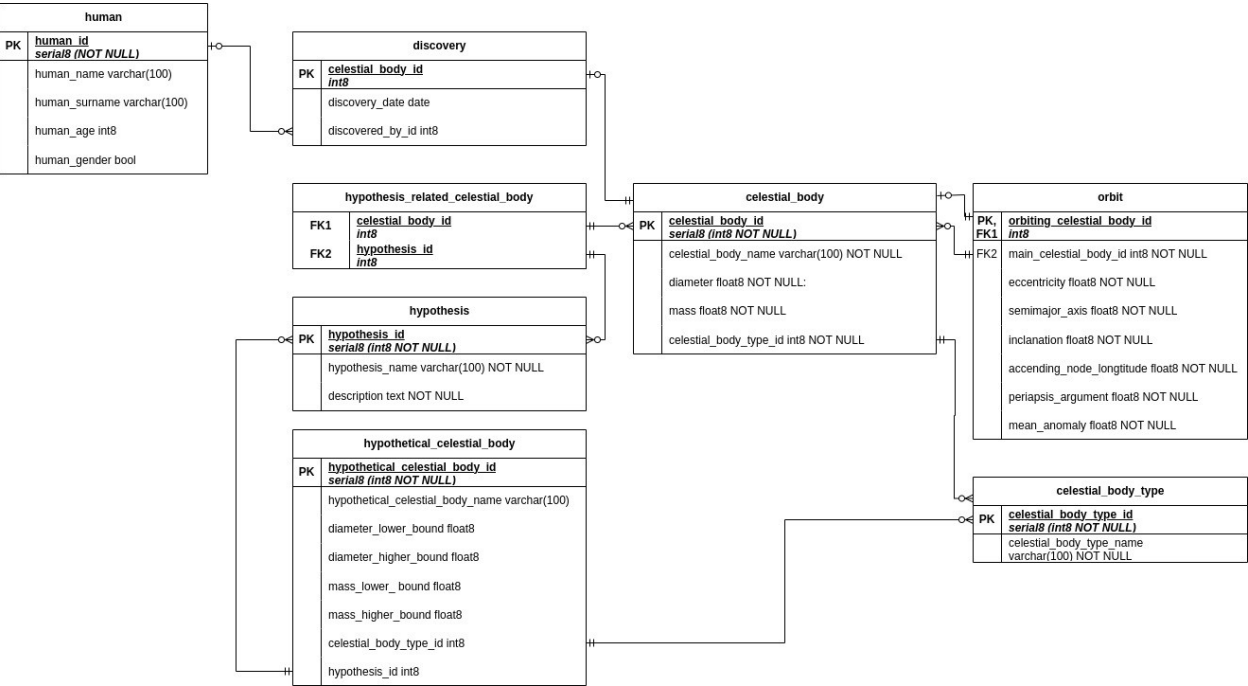
Атрибуты:

- Orbiting celestial body ID: первичный ключ, вторичный ключ на Небесное тело, орбита которого описывается
- Main celestial body ID: вторичный ключ на Небесное тело, вокруг которого находится орбитальными
- Eccentricity: эксцентриситет орбиты
- Semimajor Axis: Большая полуось (в астрономических единицах)
- Inclination: наклонение (в градусах относительно экватора)
- Ascending node longitude: долгота восходящего узла (в градусах)
- Periapsis argument: аргумент перицентра (в градусах)
- Mean anomaly: средняя аномалия (в градусах)

Инфологическая модель



Даталогическая модель



Реализация на SQL

Инициализация таблиц

```
CREATE TABLE celesital_body_type (  
    celestial_body_type_id serial8 NOT NULL,  
    celestial_body_type_name varchar(100) NOT NULL,  
    CONSTRAINT celesital_body_type_pk PRIMARY KEY (celestial_body_type_id),  
    CONSTRAINT celesital_body_type_un UNIQUE (celestial_body_type_name),  
    CONSTRAINT name_check CHECK (  
        (  
            (celestial_body_type_name):: text <> '' :: text  
        )  
    )  
);  
  
CREATE TABLE hypothesis (  
    hypothesis_id serial8 NOT NULL,  
    hypothesis_name varchar(100) NOT NULL,  
    description text NOT NULL,  
    CONSTRAINT hypothesis_pk PRIMARY KEY (hypothesis_id),  
    CONSTRAINT hypothesis_un UNIQUE (hypothesis_name),  
    CONSTRAINT name_check CHECK (  
        (  
            (hypothesis_name):: text <> '' :: text  
        )  
    )  
);  
  
CREATE TABLE celestial_body (  
    celestial_body_id serial8 NOT NULL,  
    celestial_body_name varchar(100) NOT NULL,  
    diameter float8 NOT NULL,  
    mass float8 NOT NULL,  
    celestial_body_type_id int8 NOT NULL,  
    CONSTRAINT celestial_body_pk PRIMARY KEY (celestial_body_id),  
    CONSTRAINT celestial_body_un UNIQUE (celestial_body_name),  
    CONSTRAINT diameter_check CHECK (  
        (  
            diameter > (0):: double precision  
        )  
    ),  
    CONSTRAINT mass_check CHECK (  
        (  
            mass > (0):: double precision  
        )  
    ),  
    CONSTRAINT name_check CHECK (  
        (  
            (celestial_body_name):: text <> '' :: text  
        )  
    ),  
    CONSTRAINT celestial_body_fk FOREIGN KEY (celestial_body_type_id) REFER-  
ENCES celesital_body_type(celestial_body_type_id)  
);  
  
CREATE TABLE hypothesis_related_celestial_body (  
    hypothesis_id int8 NOT NULL,  
    celestial_body_id int8 NOT NULL,  
    CONSTRAINT hypothesis_celestial_body_pk PRIMARY KEY (  
        hypothesis_id, celestial_body_id  
    ),  
);
```



```

    CONSTRAINT hypothesis_celestial_body_fk FOREIGN KEY (hypothesis_id) REFERENCES hypothesis(hypothesis_id),
    CONSTRAINT hypothesis_celestial_body_fk_1 FOREIGN KEY (celestial_body_id) REFERENCES celestial_body(celestial_body_id)
);

CREATE TABLE hypothetical_celestial_body (
    hypothetical_celestial_body_id serial8 NOT NULL,
    hypothetical_celestial_body_name varchar(100) NULL,
    diameter_lower_bound float8 NULL,
    diameter_higher_bound float8 NULL,
    mass_lower_bound float8 NULL,
    mass_higher_bound float8 NULL,
    celestial_body_type_id int8 NOT NULL,
    hypothesis_id int8 NOT NULL,
    CONSTRAINT theoretical_celestial_body_pk PRIMARY KEY (hypothetical_celestial_body_id),
    CONSTRAINT theoretical_celestial_body_fk FOREIGN KEY (celestial_body_type_id) REFERENCES celestial_body_type(celestial_body_type_id),
    CONSTRAINT theoretical_celestial_body_fk_1 FOREIGN KEY (hypothesis_id) REFERENCES hypothesis(hypothesis_id)
);

CREATE TABLE orbit (
    orbiting_celestial_body_id int8 NOT NULL,
    main_celestial_body_id int8 NOT NULL,
    eccentricity float8 NOT NULL,
    semimajor_axis float8 NOT NULL,
    inclination float8 NOT NULL,
    ascending_node_longitude float8 NOT NULL,
    periapsis_argument float8 NOT NULL,
    mean_anomaly float8 NOT NULL,
    CONSTRAINT ascending_node_longitude_check CHECK (
        (
            ascending_node_longitude >= (0):: double precision
        )
        AND (
            ascending_node_longitude < (360):: double precision
        )
    ),
    CONSTRAINT eccentricity_check CHECK (
        (
            eccentricity > (0):: double precision
        )
        AND (
            eccentricity < (1):: double precision
        )
    ),
    CONSTRAINT inclination_check CHECK (
        (
            inclination >= (0):: double precision
        )
        AND (
            inclination < (360):: double precision
        )
    ),
    CONSTRAINT mean_anomaly_check CHECK (
        (

```

```

        mean_anomaly >= (0):: double precision
    )
    AND (
        mean_anomaly < (360):: double precision
    )
)
),
CONSTRAINT orbit_pk PRIMARY KEY (orbiting_celestial_body_id),
CONSTRAINT periapsis_argument_check CHECK (
    (
        periapsis_argument >= (0):: double precision
    )
    AND (
        periapsis_argument < (360):: double precision
    )
)
),
CONSTRAINT orbit_fk FOREIGN KEY (orbiting_celestial_body_id) REFERENCES ce-
lestial_body(celestial_body_id),
CONSTRAINT orbit_fk_1 FOREIGN KEY (main_celestial_body_id) REFERENCES ce-
lestial_body(celestial_body_id)
);

```

```

CREATE TABLE discovery (
    celestial_body_id int8 NOT NULL,
    discovery_date date NOT NULL,
    discovered_by_id int8 NOT NULL,
    CONSTRAINT discovery_fk FOREIGN KEY (celestial_body_id) REFERENCES celes-
tial_body(celestial_body_id),
    CONSTRAINT discovery_fk1 FOREIGN KEY (discovered_by_id) REFERENCES
human(human_id)
);

```

Заполнение данными

```

INSERT INTO celestial_body_type (celestial_body_type_name)
VALUES
    ('Star'),
    ('Planet'),
    ('Sattelite'),
    ('Asteroid');

```

```

INSERT INTO hypothesis (hypothesis_name, description)
VALUES
    (
        '4 sattelites hypothesis', 'Согласно этой гипотезе Элара, Лиситея,
Гималия и Леда являются частями одного распавшегося небесного тела'
    ),
    (
        'Giant impact hypothesis', 'Согласно этой гипотезе, Луна была
сформирована в результате столкновения гипотетической планеты Тейя с Землёй'
    );

```

```

INSERT INTO celestial_body (
    celestial_body_name, diameter, mass,
    celestial_body_type_id
)
VALUES
    ('Sun', 1391400.0, 33295.0, 1),
    ('Jupiter', 139822.0, 317.8, 2),

```

```

('Elara', 80.0, 0.000000145, 3),
('Lysithea', 42.0, 0.000000013, 3),
('Himalia', 140.0, 0.000000703, 3),
('Leda', 22.0, 0.00000000095, 3),
('Earth', 12742.0, 1.0, 2),
('Moon', 3474.0, 0.0123, 3);

```

```

INSERT INTO hypothesis_related_celestial_body (
  hypothesis_id, celestial_body_id
)
VALUES
  (1, 3),
  (1, 4),
  (1, 5),
  (1, 6),
  (2, 7),
  (2, 8);

```

```

INSERT INTO hypothetical_celestial_body (
  hypothetical_celestial_body_name,
  diameter_lower_bound, diameter_higher_bound,
  mass_lower_bound, mass_higher_bound,
  celestial_body_type_id, hypothesis_id
)
VALUES
  (NULL, NULL, 100.0, NULL, NULL, 4, 1),
  (
    'Theia', 5000.0, 6500.0, NULL, NULL,
    2, 2
  );

```

```

INSERT INTO orbit (
  orbiting_celestial_body_id, main_celestial_body_id,
  eccentricity, semimajor_axis, inclination,
  ascending_node_longitude, periapsis_argument,
  mean_anomaly
)
VALUES
  (
    2, 1, 0.0489, 5.2038, 6.09, 100.464,
    273.867, 20.02
  ),
  (
    3, 2, 0.22, 0.078, 30.66, 90.86, 191.199,
    10.93
  ),
  (
    4, 2, 0.148, 0.078, 26.29, 343.46, 94.8,
    27.19
  ),
  (
    5, 2, 0.154, 0.076, 29.9, 45.0, 21.6,
    94.3
  ),
  (
    6, 2, 0.165, 0.0748, 27.64, 190.18, 312.93,
    137.03
  ),
  (
    7, 1, 0.0167, 1.0, 7.155, 348.8, 114.2,
    358.6
  ),
  (

```

```
8, 7, 0.055, 0.00257, 5.145, 125.08,  
318.5, 135.27  
);
```

Вывод

В ходе выполнения данной лабораторной работы я повторил основы Sql-синтаксиса, его DDL и DML составляющие, и изучил модель “Сущность-Связь”.