

Национальный исследовательский университет информационных  
технологий, механики и оптики  
Факультет ПИиКТ

Информационные системы и базы данных  
Лабораторная работа №1  
Вариант №314929

Работу выполнил:  
Конаныхина А.А.

Группа:  
Р33102

Преподаватель:  
Гаврилов А.В.

Санкт-Петербург  
2022

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.

### **Описание предметной области:**

#### **Текст варианта:**

Затем я увидел, что на ветвях образуются бутоны. Это напоминало ускоренный показ кадров с распускающимися цветами. Я действительно решил, что это цветы - каждый величиной с человеческую голову.

#### **Описание:**

Имеется набор кадров, которые проигрываются стороннему наблюдателю. Кадры имеют разрешение и глубину цвета. На кадрах есть бутоны, распускающиеся на ветвях. Бутоны могут быть в различных состояниях – раскрытые и закрытые, иметь различный размер. Бутоны растут на ветвях, у каждой из которых есть тип древесины и размер.

#### **Список сущностей:**

##### **Стержневые:**

- Ветка – размер, тип древесины.
- Кадр – разрешение кадра, глубина цвета.
- Человек – возраст, имя.

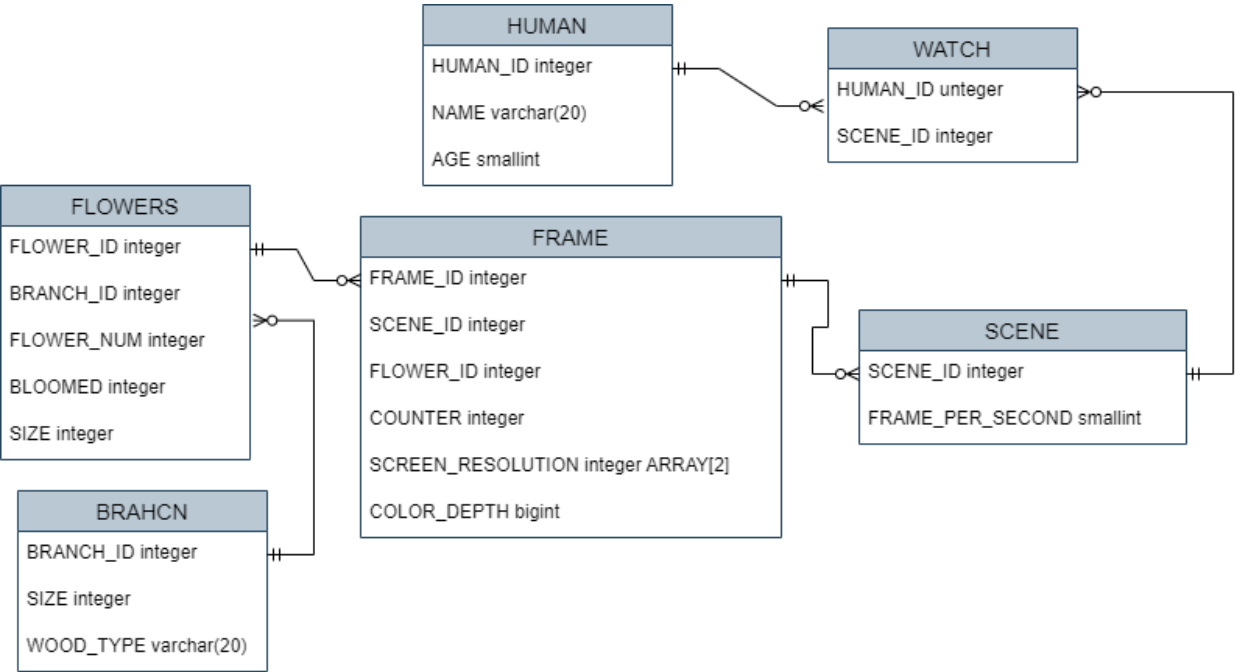
##### **Характеристические:**

- Цветок – состояние (в % раскрытости на кадре), номер цветка на кадре.

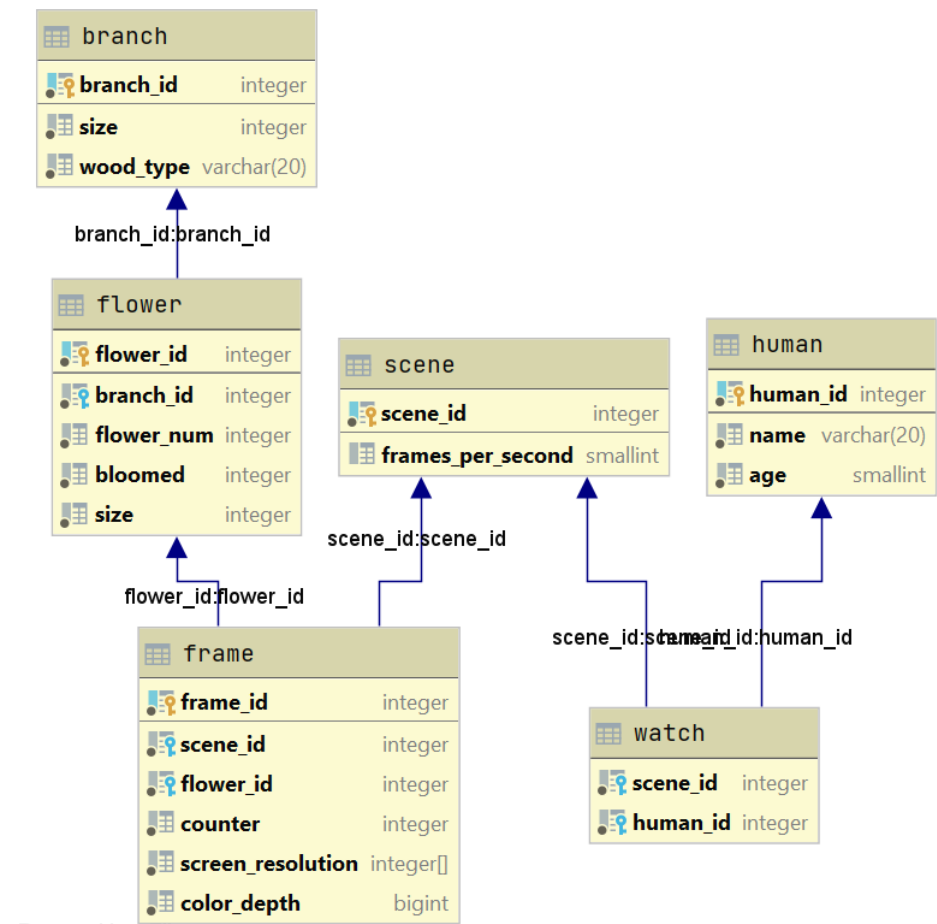
##### **Ассоциативные:**

- Сцена – наблюдатель, кадры, цветы, количество кадров в секунду.

Инфологическая модель:



Даталогическая модель:



## Реализация модели на SQL:

```
CREATE TABLE HUMAN
(
    HUMAN_ID      SERIAL PRIMARY KEY,
    NAME          VARCHAR(20) NOT NULL,
    AGE          SMALLINT NOT NULL
    CHECK (AGE >= 0)
);

CREATE TABLE BRANCH
(
    BRANCH_ID     SERIAL PRIMARY KEY,
    SIZE          INTEGER NOT NULL,
    WOOD_TYPE     VARCHAR(20) NOT NULL
    CHECK (SIZE >= 0)
);

CREATE TABLE FLOWER
(
    FLOWER_ID     SERIAL PRIMARY KEY,
    BRANCH_ID     INTEGER REFERENCES BRANCH ON DELETE CASCADE NOT NULL,
    FLOWER_NUM    INTEGER NOT NULL,
    BLOOMED       INTEGER NOT NULL,
    SIZE          INTEGER NOT NULL
    CHECK (SIZE >= 0 AND BLOOMED >= 0 AND BLOOMED <= 100)
);

CREATE TABLE SCENE
(
    SCENE_ID      SERIAL PRIMARY KEY,
    FRAMES_PER_SECOND SMALLINT
    CHECK (FRAMES_PER_SECOND >= 0)
);

CREATE TABLE WATCH
(
    SCENE_ID      INTEGER REFERENCES SCENE ON DELETE CASCADE NOT NULL,
    HUMAN_ID      INTEGER REFERENCES HUMAN ON DELETE CASCADE NOT NULL
);

CREATE TABLE FRAME
(
    FRAME_ID      SERIAL PRIMARY KEY,
    SCENE_ID      INTEGER REFERENCES SCENE ON DELETE CASCADE NOT NULL,
    FLOWER_ID     INTEGER REFERENCES FLOWER ON DELETE CASCADE NOT NULL,
    COUNTER       INTEGER NOT NULL,
    SCREEN_RESOLUTION INTEGER ARRAY[2] NOT NULL,
    COLOR_DEPTH   BIGINT NOT NULL
    CHECK ( SCREEN_RESOLUTION[0] >= 0 AND
            SCREEN_RESOLUTION[1] >= 0 AND
            COLOR_DEPTH >= 0)
);

INSERT INTO HUMAN (NAME, AGE)
VALUES ('ME', 3);

INSERT INTO BRANCH (SIZE, WOOD_TYPE)
VALUES (10, 'Поза'),
       (20, 'Сирень');

INSERT INTO FLOWER (FLOWER_NUM, BRANCH_ID, BLOOMED, SIZE)
VALUES (1, 1, 0, 10);
```

```

(1, 1, 20, 50),
(1, 1, 50, 70),
(1, 1, 90, 90),
(1, 1, 100, 150),
(2, 2, 20, 5),
(2, 2, 50, 20);

INSERT INTO SCENE (FRAMES_PER_SECOND)
VALUES (24),
(24),
(24);

INSERT INTO WATCH (SCENE_ID, HUMAN_ID)
VALUES (1, 1),
(2, 1),
(3, 1);

INSERT INTO FRAME (SCENE_ID, FLOWER_ID, COUNTER, SCREEN_RESOLUTION,
COLOR_DEPTH)
VALUES (1, 1, 0, '{1920, 1080}', 255),
(1, 2, 1, '{1920, 1080}', 255),
(1, 3, 2, '{1920, 1080}', 255),
(1, 4, 3, '{1920, 1080}', 255),
(1, 5, 4, '{1920, 1080}', 255),
(2, 6, 0, '{1920, 1080}', 255),
(2, 7, 1, '{1920, 1080}', 255);

```

## Вывод:

При выполнении лабораторной работы были изучены правила построения инфологической и даталогической моделей сущностей, по которым была реализована база данных с помощью SQL.

Запрос: вывести всех пользователей, которые смотрят на сцену, на которой есть цветок, максимально раскрывшийся за всю длительность сцены (т. е. разница в раскрытии между первым и последним кадром максимальна).

```

WITH DIFF_TAB AS (
    SELECT SCENE_ID, MAX(BLOOMED) - MIN(BLOOMED) AS DIFF
    FROM FRAME JOIN FLOWER F ON F.FLOWER_ID = FRAME.FLOWER_ID
    GROUP BY FRAME.SCENE_ID, F.FLOWER_NUM)
SELECT DISTINCT H.HUMAN_ID, H.NAME, H.AGE FROM
DIFF_TAB JOIN WATCH W ON W.SCENE_ID = DIFF_TAB.SCENE_ID JOIN HUMAN H ON
H.HUMAN_ID = W.HUMAN_ID
WHERE DIFF_TAB.DIFF = (SELECT MAX(DIFF_TAB.DIFF) FROM DIFF_TAB);

```