

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное
образовательное учреждение высшего образования
“Национальный исследовательский университет ИТМО”

**ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ
И КОМПЬЮТЕРНОЙ ТЕХНИКИ**

ЛАБОРАТОРНАЯ РАБОТА №1

по дисциплине
“Информационные системы и Базы Данных”

Вариант: 336759.

выполнил:

Студент группы Р33311

Птицын Максим Евгеньевич

Преподаватель

Николаев Владимир Вячеславович

2023 г.

г. Санкт-Петербург

Содержание

1	Текст задания.	3
2	Описание предметной области.	3
3	Список и классификация сущностей.	3
3.1	Стержневые.	3
3.2	Ассоциативные.	3
3.3	Характеристические.	3
4	Инфологическая модель.	4
5	Даталогическая модель.	4
6	Реализация даталогической модели на SQL.	5
7	Выводы по работе.	7

1 Текст задания.

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.

2 Описание предметной области.

В сопровождении Элли и Малкольма, Грант обошел главное здание. Следом за ними шел мальчик. Грант любил детей. А как их можно не любить, когда они так непосредственно, так страстно интересуются динозаврами. Гранту приходилось видеть, как в музеях дети стояли с открытыми ртами, взирая на огромные скелеты, уходящие под самый потолок. Он часто спрашивал себя, почему вымершие ящеры производят такое сильное впечатление на детей. Но потом он понял, что дети любят динозавров потому, что эти гигантские создания воплощают в себе управляемую силу неограниченной власти. Динозавры символизируют родителей, которых дети обожают, но боятся. Дети любят динозавров точно так же, как они любят своих родителей.

3 Список и классификация сущностей.

3.1 Стержневые.

1. actors
2. actions
3. feels
4. actors_categories_history

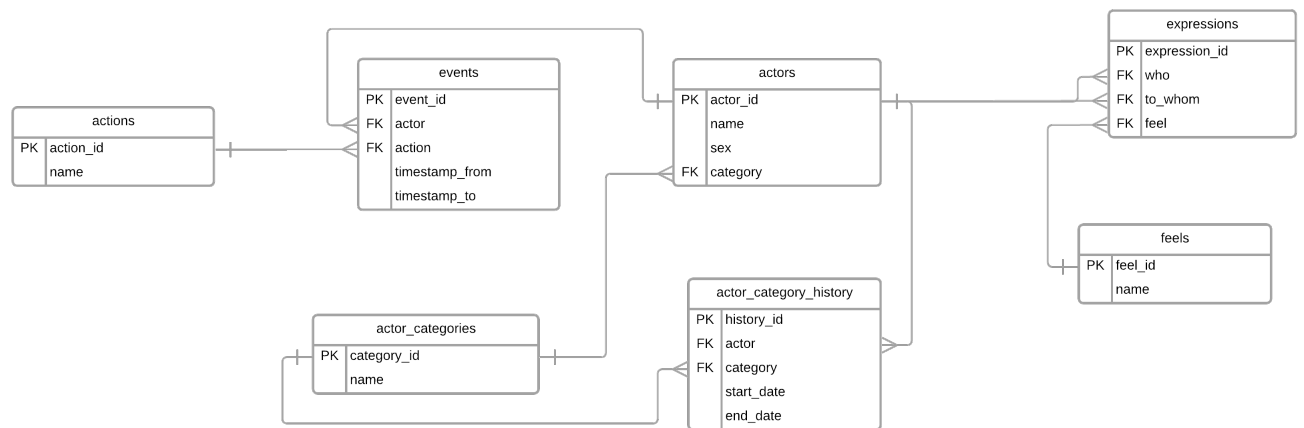
3.2 Ассоциативные.

1. events
2. expressions

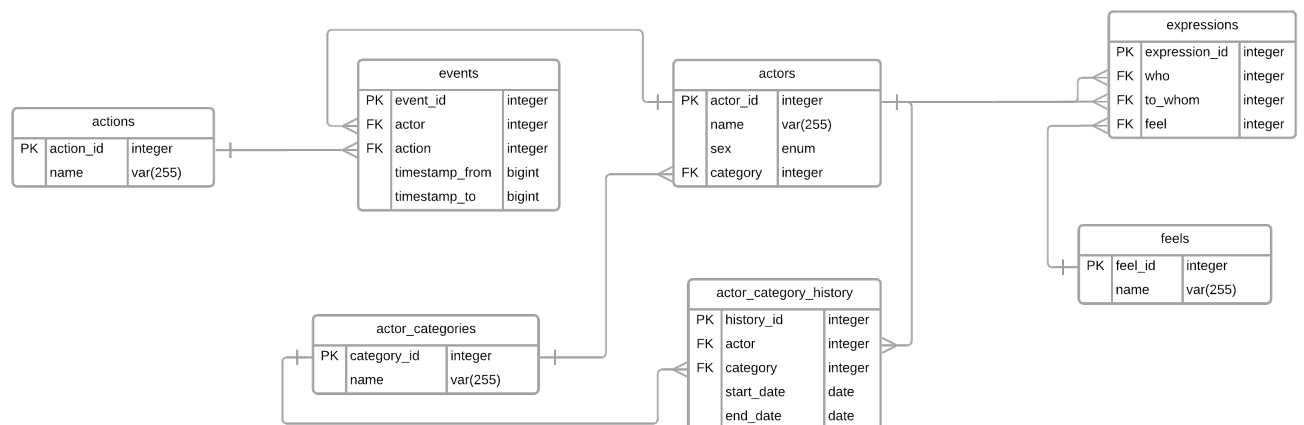
3.3 Характеристические.

1. actors_categories

4 Инфологическая модель.



5 Даталогическая модель.



6 Реализация даталогической модели на SQL.

```
CREATE TYPE sex AS ENUM ('male', 'female', 'both');

CREATE TABLE actions (
  action_id SERIAL PRIMARY KEY,
  name VARCHAR(255) NOT NULL
);

CREATE TABLE feels (
  feel_id SERIAL PRIMARY KEY,
  name VARCHAR(255) NOT NULL
);

CREATE TABLE actor_categories (
  category_id SERIAL PRIMARY KEY,
  name VARCHAR(255) NOT NULL
);

CREATE TABLE actors (
  actor_id SERIAL PRIMARY KEY,
  name VARCHAR(255),
  sex sex,
  category INT REFERENCES actor_categories(category_id)
);

CREATE TABLE actor_category_history (
  history_id SERIAL PRIMARY KEY,
  actor INT NOT NULL,
  category INT NOT NULL,
  start_date DATE,
  end_date DATE,
  FOREIGN KEY (actor) REFERENCES actors(actor_id),
  FOREIGN KEY (category) REFERENCES actor_categories(category_id)
);

CREATE TABLE events (
  event_id SERIAL PRIMARY KEY,
  actor INT REFERENCES actors(actor_id) NOT NULL,
  action INT REFERENCES actions(action_id) NOT NULL,
  timestamp_from BIGINT,
  timestamp_to BIGINT
);

CREATE TABLE expressions (
  expression_id SERIAL PRIMARY KEY,
  who INT REFERENCES actors(actor_id) NOT NULL,
  to_whom INT REFERENCES actors(actor_id) NOT NULL,
  feel INT REFERENCES feels(feel_id) NOT NULL
);
```

```

INSERT INTO actor_categories (name)
VALUES
    ('взрослые'),
    ('дети'),
    ('динозавры'),
    ('экспонаты');

INSERT INTO actors (name, sex, category)
VALUES
    ('элли', 'female', (SELECT category_id FROM actor_categories WHERE name = 'взрослые')),
    ('малкольм', 'male', (SELECT category_id FROM actor_categories WHERE name = 'взрослые')),
    ('грант', 'male', (SELECT category_id FROM actor_categories WHERE name = 'взрослые')),
    ('мальчик', 'male', (SELECT category_id FROM actor_categories WHERE name = 'дети')),
    ('ребенок', 'both', (SELECT category_id FROM actor_categories WHERE name = 'дети')),
    ('родитель', 'both', (SELECT category_id FROM actor_categories WHERE name = 'взрослые')),
    ('динозавр', 'both', (SELECT category_id FROM actor_categories WHERE name = 'экспонаты'));

INSERT INTO actor_category_history (actor, category, start_date, end_date)
VALUES
    ( (SELECT actor_id FROM actors WHERE name = 'элли'), (SELECT category_id FROM actor_categories WHERE name = 'дети'), '1970-02-03', '1988-02-03'),
    ( (SELECT actor_id FROM actors WHERE name = 'элли'), (SELECT category_id FROM actor_categories WHERE name = 'взрослые'), '1988-02-03', NULL),
    ( (SELECT actor_id FROM actors WHERE name = 'малкольм'), (SELECT category_id FROM actor_categories WHERE name = 'дети'), '1978-03-12', '1996-03-12'),
    ( (SELECT actor_id FROM actors WHERE name = 'малкольм'), (SELECT category_id FROM actor_categories WHERE name = 'взрослые'), '1996-03-12', NULL),
    ( (SELECT actor_id FROM actors WHERE name = 'грант'), (SELECT category_id FROM actor_categories WHERE name = 'дети'), '1974-12-10', '1992-12-10'),
    ( (SELECT actor_id FROM actors WHERE name = 'грант'), (SELECT category_id FROM actor_categories WHERE name = 'взрослые'), '1992-12-10', NULL),
    ( (SELECT actor_id FROM actors WHERE name = 'мальчик'), (SELECT category_id FROM actor_categories WHERE name = 'дети'), '2000-12-10', NULL),
    ( (SELECT actor_id FROM actors WHERE name = 'ребенок'), (SELECT category_id FROM actor_categories WHERE name = 'дети'), NULL, NULL),
    ( (SELECT actor_id FROM actors WHERE name = 'родитель'), (SELECT category_id FROM actor_categories WHERE name = 'взрослые'), NULL, NULL),
    ( (SELECT actor_id FROM actors WHERE name = 'динозавр'), (SELECT category_id FROM actor_categories WHERE name = 'динозавры'), NULL, NULL),
    ( (SELECT actor_id FROM actors WHERE name = 'динозавр'), (SELECT category_id FROM actor_categories WHERE name = 'экспонаты'), '1980-03-05', NULL);

INSERT INTO feels (name)
VALUES
    ('любовь'),
    ('интерес'),
    ('впечатляться'),
    ('обожание'),
    ('страх');

INSERT INTO actions (name)
VALUES
    ('обойти здание'),
    ('следовать'),
    ('смотреть'),
    ('стоять');

INSERT INTO expressions (who, to_whom, feel)
VALUES
    ((SELECT actor_id FROM actors WHERE name = 'грант'), (SELECT actor_id FROM actors WHERE name = 'ребенок'), (SELECT feel_id FROM feels WHERE name = 'любовь')),
    ((SELECT actor_id FROM actors WHERE name = 'ребенок'), (SELECT actor_id FROM actors WHERE name = 'динозавр'), (SELECT feel_id FROM feels WHERE name = 'интерес')),
    ((SELECT actor_id FROM actors WHERE name = 'динозавр'), (SELECT actor_id FROM actors WHERE name = 'ребенок'), (SELECT feel_id FROM feels WHERE name = 'впечатляться')),
    ((SELECT actor_id FROM actors WHERE name = 'ребенок'), (SELECT actor_id FROM actors WHERE name = 'динозавр'), (SELECT feel_id FROM feels WHERE name = 'любовь')),
    ((SELECT actor_id FROM actors WHERE name = 'ребенок'), (SELECT actor_id FROM actors WHERE name = 'родитель'), (SELECT feel_id FROM feels WHERE name = 'обожание')),
    ((SELECT actor_id FROM actors WHERE name = 'ребенок'), (SELECT actor_id FROM actors WHERE name = 'родитель'), (SELECT feel_id FROM feels WHERE name = 'страх')),
    ((SELECT actor_id FROM actors WHERE name = 'ребенок'), (SELECT actor_id FROM actors WHERE name = 'родитель'), (SELECT feel_id FROM feels WHERE name = 'любовь'));

INSERT INTO events (actor, action, timestamp_from, timestamp_to)
VALUES
    ((SELECT actor_id FROM actors WHERE name = 'элли'), (SELECT action_id FROM actions WHERE name = 'обойти здание'), 1000, 2000),
    ((SELECT actor_id FROM actors WHERE name = 'малкольм'), (SELECT action_id FROM actions WHERE name = 'обойти здание'), 1000, 2000),
    ((SELECT actor_id FROM actors WHERE name = 'грант'), (SELECT action_id FROM actions WHERE name = 'обойти здание'), 1000, 2000),
    ((SELECT actor_id FROM actors WHERE name = 'мальчик'), (SELECT action_id FROM actions WHERE name = 'следовать'), 2000, 3000),
    ((SELECT actor_id FROM actors WHERE name = 'ребенок'), (SELECT action_id FROM actions WHERE name = 'стоять'), 3000, 4000),
    ((SELECT actor_id FROM actors WHERE name = 'ребенок'), (SELECT action_id FROM actions WHERE name = 'смотреть'), 4000, 5000);

```

7 Выводы по работе.

В результате выполнения лабораторной работы были применены навыки выявления сущностей по описанию предметной области, создана инфологическая и даталогическая модель, получены навыки написания DDL и DML запросов на языке SQL для базы данных PostgreSQL.