

Университет ИТМО
Факультет программной инженерии и компьютерной техники

**Базы данных.
Лабораторная работа №1.**

Группа: Р33131
Студент: Смирнов Виктор Игоревич
Вариант: 310963

2023

Ключевые слова

База данных, PostgreSQL, даталогическая модель, инфологическая модель.

Содержание

1	Цель работы	1
2	Текст задания	1
3	Описание предметной области	1
4	Классификация сущностей	1
5	Инфологическая модель	2
6	Даталогическая модель	2
7	Схема данных на PostgreSQL	2
8	Инициализация БД PostgreSQL	3
9	Запрос к сущностей из БД	4
10	Вывод	5

1 Цель работы

Научиться проектировать базы данных, составлять инфологические и даталогические модели данных, реализовывать их в БД PostgreSQL, научиться выполнять запросы.

2 Текст задания

Как бы там ни было, вид спускающихся с дерева загадочных существ произвел слишком тягостное впечатление на динозавриху. Загоготовав на прощание, животное подтолкнуло малыша и медленно поплелось прочь.

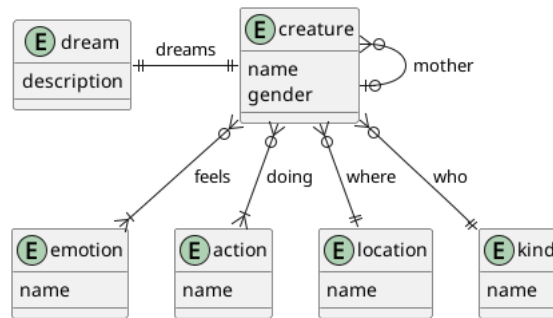
3 Описание предметной области

Из текста сразу выделяем действующие лица: загадочное существо, динозавриха, малыш – их можно назвать одним словом – существа. У существ есть имя, пол, они что-то чувствуют, что-то делают и где-то находятся. Чувств, действий и местоположений может быть немеренное количество, поэтому их целесообразно выделить в отдельные таблицы для гибкости схемы данных.

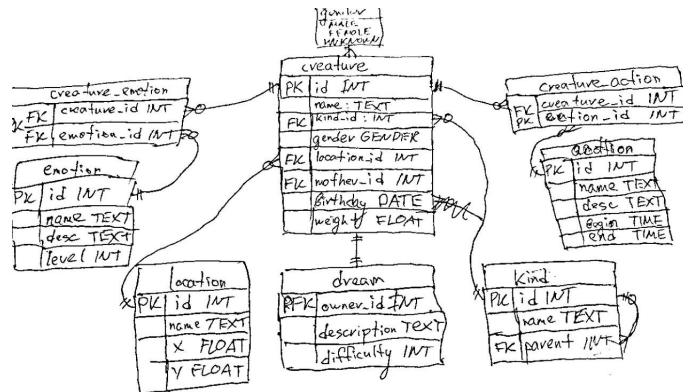
4 Классификация сущностей

1. `creature` – стержневая сущность
2. `dream` – характеристическая сущность
3. `kind` – характеристическая сущность
4. `location` – характеристическая сущность
5. `emotion` – ассоциативная сущность
6. `action` – ассоциативная сущность

5 Инфологическая модель



6 Даталогическая модель



7 Схема данных на PostgreSQL

```

1  -- Scheme Migration #1: Initialize Tables
2  -- Diagram: doc/lab-1/entity.puml
3
4  CREATE DOMAIN nametext
5  AS varchar(70)
6  CHECK (VALUE ~ '[a-zA-z ,.'-]{2,70}');
7
8  CREATE DOMAIN locationtext
9  AS varchar(130)
10 CHECK (VALUE ~ '[a-zA-z ,.'-]{2,130}');
11
12 CREATE SEQUENCE seq_kind_id START 1;
13
14 CREATE TABLE kind (
15     id int PRIMARY KEY DEFAULT nextval('seq_kind_id'),
16     name nametext NOT NULL UNIQUE,
17     parent_id int references kind(id)
18 );
19
20 CREATE SEQUENCE seq_location_id START 1;
21
22 CREATE TABLE location (
23     id int PRIMARY KEY DEFAULT nextval('seq_location_id'),
24     name locationtext NOT NULL UNIQUE,
25     x real NOT NULL,
26     y real NOT NULL,
27     z real NOT NULL
28 );
    
```

```

29
30 CREATE SEQUENCE seq_action_id START 1;
31
32 CREATE TABLE action (
33     id int PRIMARY KEY DEFAULT nextval('seq_action_id'),
34     name nametext NOT NULL,
35     description text NOT NULL,
36     speed int NOT NULL CHECK(speed > 0)
37 );
38
39 CREATE SEQUENCE seq_emotion_id START 1;
40
41 CREATE TABLE emotion (
42     id int PRIMARY KEY DEFAULT nextval('seq_action_id'),
43     name nametext NOT NULL,
44     description text NOT NULL,
45     level int NOT NULL CHECK(level > 0)
46 );
47
48 CREATE TYPE gender AS ENUM ('male', 'female', 'unknown');
49
50 CREATE SEQUENCE seq_creature_id START 1;
51
52 CREATE TABLE creature (
53     id int PRIMARY KEY DEFAULT nextval('seq_creature_id'),
54     name nametext NOT NULL,
55     gender gender NOT NULL,
56     kind_id int NOT NULL references kind(id),
57     location_id int NOT NULL references location(id),
58     mother_id int references creature(id),
59     birthday timestamp NOT NULL DEFAULT NOW(),
60     weight int NOT NULL CHECK(weight > 0)
61 );
62
63 CREATE TABLE dream (
64     owner_id int PRIMARY KEY references creature(id),
65     description text NOT NULL,
66     difficulty int NOT NULL CHECK(difficulty > 0)
67 );
68
69 CREATE TABLE creature_emotion (
70     creature_id int references creature(id),
71     emotion_id int references emotion(id),
72     PRIMARY KEY (creature_id, emotion_id)
73 );
74
75 CREATE TABLE creature_action (
76     creature_id int references creature(id),
77     action_id int references action(id),
78     PRIMARY KEY (creature_id, action_id)
79 );

```

8 Инициализация БД PostgreSQL

```

1 -- Scheme Migration #1: Inserting Testing Data
2
3 INSERT INTO kind (name) VALUES
4     ('Dinosaur'),
5     ('Mysterious Creature');
6
7 INSERT INTO location (name, x, y, z) VALUES
8     ('Tree in Jungles', 100, 100, 50),
9     ('Somewhere in Jungles', 100, 100, 0);
10
11 INSERT INTO action (name, description, speed) VALUES
12     ('Terrify', 'To make too painful an impression', 100),
13     ('Climbing down', 'Climbing down', 5),
14     ('Cackle', 'Shouting, making guffaws', 50),
15     ('Push', 'Push', 10),
16     ('Slowly trudge away', 'Slowly trudge away', 5),
17     ('Sitting', 'Do nothing', 1);
18

```

```

19 INSERT INTO emotion (name, description, level) VALUES
20     ('Fear', 'Fear', 5),
21     ('Confidence', 'Confidence', 3),
22     ('Love', 'Love', 10);
23
24 INSERT INTO creature (name, gender, kind_id, location_id, mother_id, weight)
25     SELECT 'Kostyan', 'unknown', kind.id, location.id, NULL, 55
26     FROM kind, location
27     WHERE kind.name = 'Mysterious Creature'
28           AND location.name = 'Tree in Jungles';
29
30 INSERT INTO creature (name, gender, kind_id, location_id, mother_id, weight)
31     SELECT 'Dino-Mother', 'female', kind.id, location.id, NULL, 500
32     FROM kind, location
33     WHERE kind.name = 'Dinosaur'
34           AND location.name = 'Somewhere in Jungles';
35
36 INSERT INTO creature (name, gender, kind_id, location_id, mother_id, weight)
37     SELECT 'Dino-Child', 'male', kind.id, location.id, mother.id, 250
38     FROM kind, location, creature as mother
39     WHERE kind.name = 'Dinosaur'
40           AND location.name = 'Somewhere in Jungles'
41           AND mother.name = 'Dino-Mother';
42
43 INSERT INTO creature_emotion (creature_id, emotion_id)
44     SELECT creature.id, emotion.id FROM creature, emotion
45     WHERE creature.name = 'Kostyan'
46           AND emotion.name = 'Confidence';
47
48 INSERT INTO creature_emotion (creature_id, emotion_id)
49     SELECT creature.id, emotion.id FROM creature, emotion
50     WHERE creature.name IN ('Dino-Mother', 'Dino-Child')
51           AND emotion.name IN ('Fear', 'Love');
52
53 INSERT INTO creature_action (creature_id, action_id)
54     SELECT creature.id, action.id FROM creature, action
55     WHERE creature.name = 'Kostyan'
56           AND action.name IN ('Terrify', 'Climbing down');
57
58 INSERT INTO creature_action (creature_id, action_id)
59     SELECT creature.id, action.id FROM creature, action
60     WHERE creature.name = 'Dino-Mother'
61           AND action.name IN ('Cackle', 'Push', 'Slowly trudge away');
62
63 INSERT INTO creature_action (creature_id, action_id)
64     SELECT creature.id, action.id FROM creature, action
65     WHERE creature.name = 'Dino-Child'
66           AND action.name IN ('Sitting');
67
68 INSERT INTO dream (owner_id, description, difficulty)
69     SELECT id, 'To be happy and fall in love with some other creature', 999
70     FROM creature
71     WHERE creature.name = 'Kostyan';
72
73 INSERT INTO dream (owner_id, description, difficulty)
74     SELECT id, 'To rise up a decent and strong child', 500
75     FROM creature
76     WHERE creature.name = 'Dino-Mother';
77
78 INSERT INTO dream (owner_id, description, difficulty)
79     SELECT id, 'To pass databases course on se.ifmo.ru', 250
80     FROM creature
81     WHERE creature.name = 'Dino-Child';

```

9 Запрос к сущностям из БД

```

1 SELECT creature.id,
2        creature.name,
3        creature.gender,
4        creature.birthday,
5        kind.name as kind,
6        location.name as location,

```

```

7      location.x,
8      location.y,
9      location.z,
10     mother.name as mother,
11     dream.description as dream,
12     dream.difficulty
13 FROM creature
14 JOIN kind ON creature.kind_id = kind.id
15 JOIN location ON creature.location_id = location.id
16 LEFT JOIN creature as mother ON creature.mother_id = mother.id
17 JOIN dream ON dream.owner_id = creature.id;

```

10 Вывод

Проектирование БД – непростое занятие, которое лучше осуществлять итеративно. Сначала описать предметную область словами, чтобы лучше понять суть проблемы. Далее опуститься на уровень ниже и составить инфологическую модель данных, которая никак не связана с конкретной БД, а лишь выражает главные связи ваших данных. Когда инфологическая модель будет готова, по ней можно будет составить даталогическую модель данных – наиболее близкое к выбранной БД представление. И только после выполнения всех вышеперечисленных шагов можно приступить к реализации схемы БД на выбранном диалекте SQL, так удастся свести риски неудачного дизайна БД к минимуму.

Список литературы

- [1] PostgreSQL Home Page
- [2] ИТМО ВТ. Информационные системы и базы данных
- [3] Базы данных. Проектирование. R class Tech