

Федеральное государственное автономное образовательное учреждение высшего
образования «Национальный исследовательский университет ИТМО»

Факультет Программной Инженерии и Компьютерной Техники

Лабораторная работа №1

По дисциплине
“Базы данных”

Вариант 10102

Выполнил:

Ахроров Кароматуллохон Фирдавсович

Группа Р3110

Проверил:

Гаврилов Антон Валерьевич

Преподаватель практики

Содержание

<i>Задание.....</i>	<i>3</i>
<i>Описание предметной области.....</i>	<i>3</i>
<i>Список сущностей и их классификация.....</i>	<i>3</i>
<i>Инфологическая модель.....</i>	<i>3</i>
<i>Даталогическая модель.....</i>	<i>4</i>
<i>Реализация даталогической модели на SQL.....</i>	<i>4</i>
<i>Вывод.....</i>	<i>6</i>

Задание

На основе предложенной предметной области (текста) составить ее описание. Из полученного описания выделить сущности, их атрибуты и связи.

Составить инфологическую модель.

Составить даталогическую модель. При описании типов данных для атрибутов должны использоваться типы из СУБД PostgreSQL.

Реализовать даталогическую модель в PostgreSQL. При описании и реализации даталогической модели должны учитываться ограничения целостности, которые характерны для полученной предметной области.

Заполнить созданные таблицы тестовыми данными.

Описание предметной области

Зоопарк с дилофозаврами представляет собой комплексную систему, где каждый элемент – от заданий и инфраструктурных объектов до сотрудников и самих животных – играет важную роль в обеспечении качественного ухода и безопасности обитателей.

В зоопарке каждое мероприятие фиксируется через уникальные задания (tasks), в рамках которых проводятся работы по обслуживанию территории и уходу за обитателями.

Задания могут включать мероприятия по уходу за зелёными насаждениями – обработка деревьев (trees) и устранение старых или повреждённых металлических креплений (metal_fastenings), что позволяет поддерживать вид и безопасность вольеров. Для обозначения завершения отдельных этапов работ используются метки (markers), которые служат сигналом для дальнейших действий, таких как сооружение подпорок (supports) для укрепления конструкций в зоопарке. Эти работы могут выполняться работниками по озеленению (greenkeepers), координируемыми через специальное распределение обязанностей (support_builders).

Безопасность зоопарка обеспечивается за счёт системы оград (fences), по разные стороны которых располагаются рабочие (workers), выполняющие свои обязанности по уходу за животными. Каждый сотрудник зоопарка имеет уникальный идентификатор, что упрощает управление персоналом и позволяет оперативно реагировать на любые изменения в состоянии объектов.

Особое внимание уделяется обитателям – дилофозаврам (dilofozavr), которые находятся в естественной среде, зачастую вблизи водных объектов (rivers). Их присутствие и поведение, включая инциденты с плевыми слюной (spit_events), тщательно фиксируются для анализа и обеспечения безопасности как животных, так и персонала.

Наблюдение за выполнением задач и состоянием зоопарка осуществляется наблюдателями (observers), которые с помощью связей (observer_tasks) контролируют выполнение работ и могут оперативно реагировать на возникшие инциденты. Такая интегрированная система позволяет не только поддерживать высокий уровень ухода за животными, но и обеспечивает эффективное управление процессами в зоопарке.

Основные сущности:

- tasks
- trees
- metal_fastenings
- markers
- supports
- greenkeepers

- fences
- workers
- rivers
- dilofoavrs
- observers

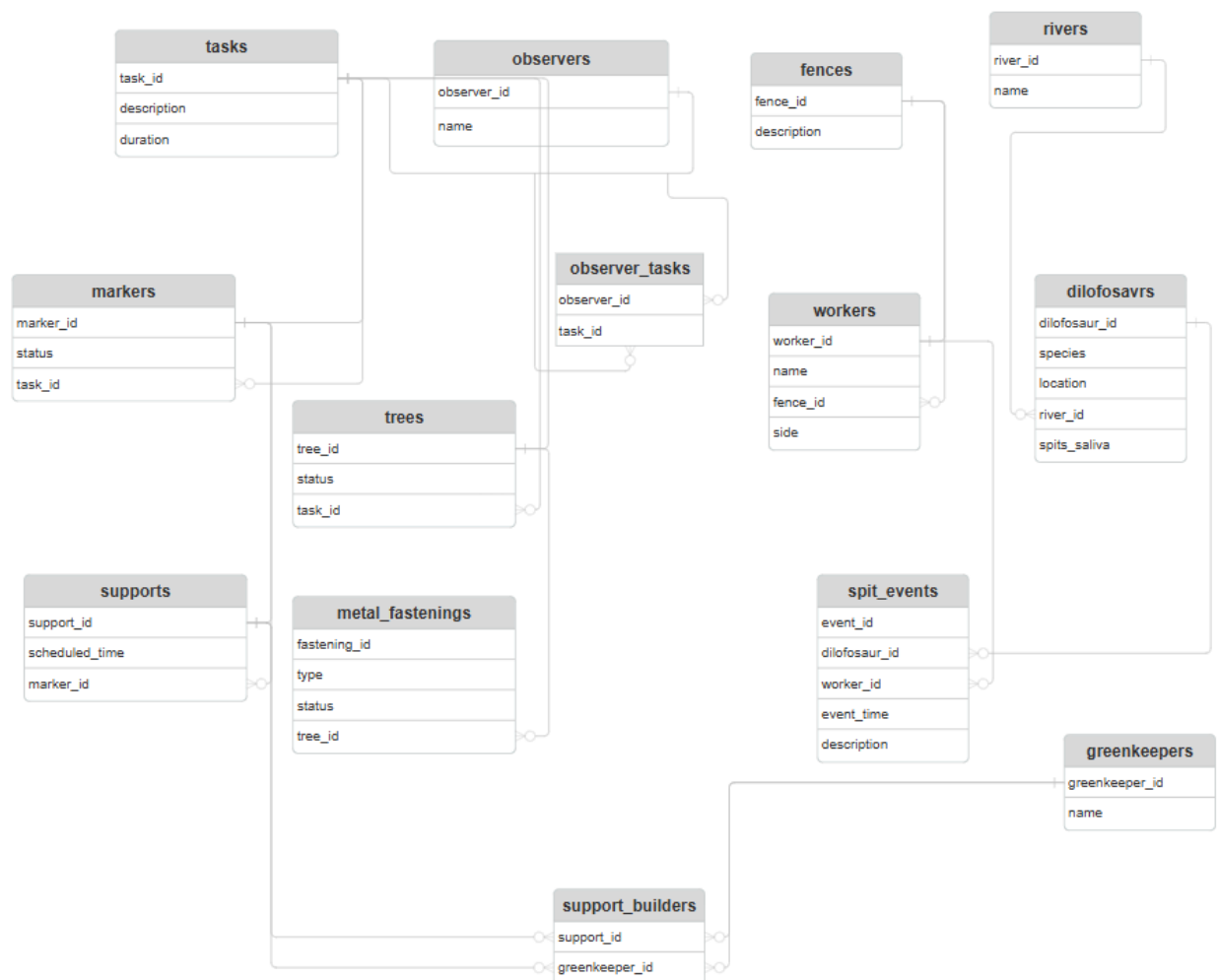
Ассоциативные сущности:

- support_builders
- observer_tasks

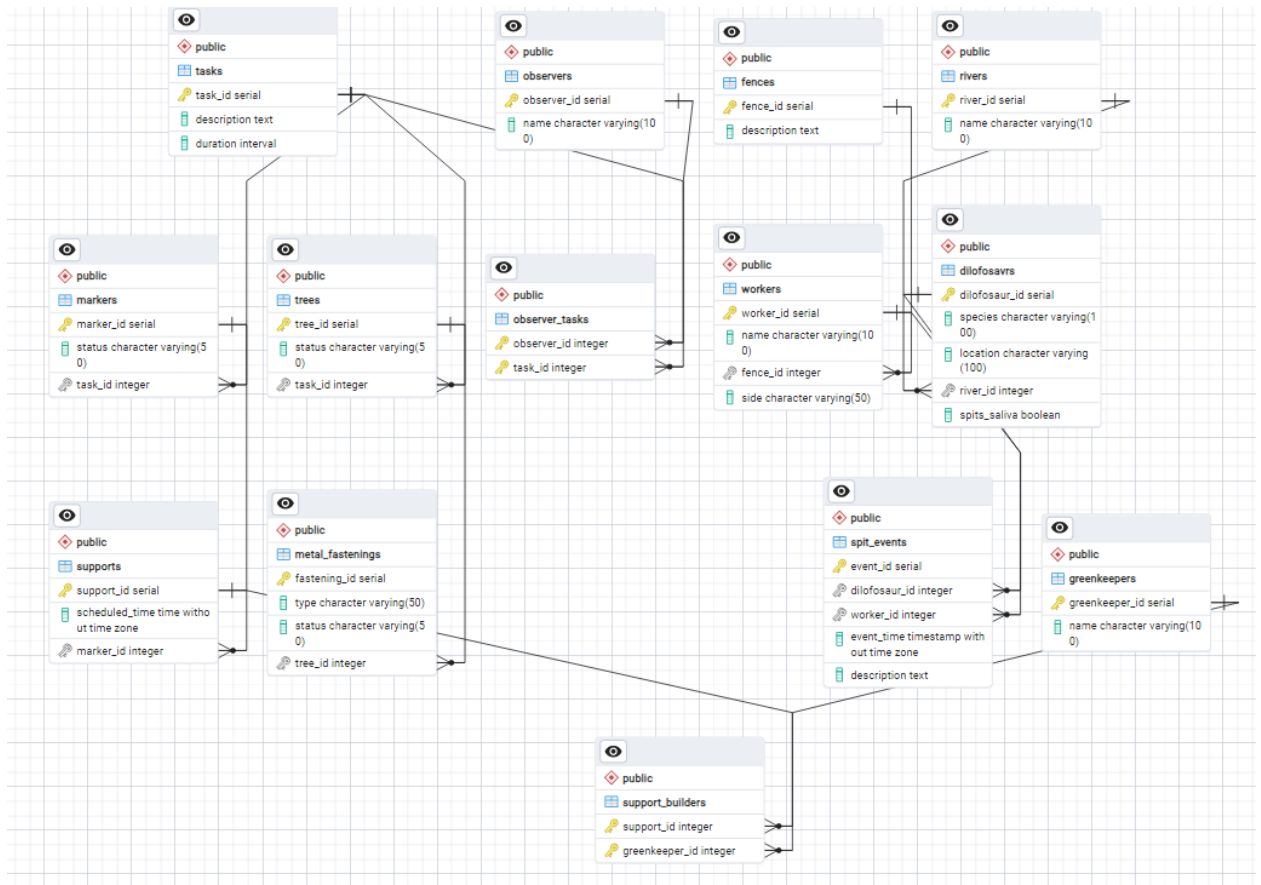
Событийные сущности:

- spit_events

Инфологическая модель



Даталогическая модель



Реализация даталогической модели на SQL

Git code - https://github.com/Ahrorovk/ITMO_SWE/tree/main/SEM_2/DB/lab_1

```

CREATE TABLE tasks (
    task_id SERIAL PRIMARY KEY,
    description TEXT NOT NULL,
    duration INTERVAL NOT NULL CHECK (duration > INTERVAL '0 minutes')
);

CREATE TABLE trees (
    tree_id SERIAL PRIMARY KEY,
    status VARCHAR(50) NOT NULL,
    task_id INTEGER REFERENCES tasks(task_id) ON DELETE CASCADE
);

CREATE TABLE metal_fastenings (
    fastening_id SERIAL PRIMARY KEY,
    type VARCHAR(50) DEFAULT 'metal',
    status VARCHAR(50) NOT NULL,
    tree_id INTEGER REFERENCES trees(tree_id) ON DELETE CASCADE
);

CREATE TABLE markers (
    marker_id SERIAL PRIMARY KEY,
    status VARCHAR(50) NOT NULL,

```

```

    task_id INTEGER REFERENCES tasks(task_id) ON DELETE CASCADE
);

CREATE TABLE supports (
    support_id SERIAL PRIMARY KEY,
    scheduled_time TIME NOT NULL,
    marker_id INTEGER REFERENCES markers(marker_id) ON DELETE SET NULL
);

CREATE TABLE greenkeepers (
    greenkeeper_id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL
);

CREATE TABLE support_builders (
    support_id INTEGER REFERENCES supports(support_id) ON DELETE CASCADE,
    greenkeeper_id INTEGER REFERENCES greenkeepers(greenkeeper_id) ON DELETE CASCADE,
    PRIMARY KEY (support_id, greenkeeper_id)
);

CREATE TABLE fences (
    fence_id SERIAL PRIMARY KEY,
    description TEXT NOT NULL
);

CREATE TABLE workers (
    worker_id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    fence_id INTEGER REFERENCES fences(fence_id) ON DELETE SET NULL,
    side VARCHAR(50) NOT NULL
);

CREATE TABLE rivers (
    river_id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL
);

CREATE TABLE dilofosavrs (
    dilofosaur_id SERIAL PRIMARY KEY,
    species VARCHAR(100) DEFAULT 'Дилофозавры',
    location VARCHAR(100),
    river_id INTEGER REFERENCES rivers(river_id) ON DELETE SET NULL,
    spits_saliva BOOLEAN DEFAULT TRUE
);

CREATE TABLE observers (
    observer_id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL
);

```

```

CREATE TABLE observer_tasks (
  observer_id INTEGER REFERENCES observers(observer_id) ON DELETE CASCADE,
  task_id INTEGER REFERENCES tasks(task_id) ON DELETE CASCADE,
  PRIMARY KEY (observer_id, task_id)
);

CREATE TABLE spit_events (
  event_id SERIAL PRIMARY KEY,
  dilofosaur_id INTEGER REFERENCES dilofosavrs(dilofosaur_id) ON DELETE CASCADE,
  worker_id INTEGER REFERENCES workers(worker_id) ON DELETE CASCADE,
  event_time TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
  description TEXT
);

INSERT INTO tasks (description, duration)
VALUES ('Выпрямить деревце, убрать металлические крепления и поставить метку для подпорки', INTERVAL '20 minutes');

INSERT INTO trees (status, task_id)
VALUES ('выпрямлено', 1);

INSERT INTO metal_fastenings (status, tree_id)
VALUES ('удалено', 1);

INSERT INTO markers (status, task_id)
VALUES ('установлена', 1);

INSERT INTO supports (scheduled_time, marker_id)
VALUES ('08:00:00', 1);

INSERT INTO greenkeepers (name) VALUES ('Иван Иванов');
INSERT INTO greenkeepers (name) VALUES ('Пётр Петров');

INSERT INTO support_builders (support_id, greenkeeper_id)
VALUES (1, 1), (1, 2);

INSERT INTO fences (description)
VALUES ('Ограда участка');

INSERT INTO workers (name, fence_id, side)
VALUES ('Алексей', 1, 'северная'),
      ('Борис', 1, 'южная');

INSERT INTO rivers (name)
VALUES ('Река Дила');
INSERT INTO dilofosavrs (location, river_id)
VALUES ('недалеко от реки', 1);

```

```
INSERT INTO observers (name)
VALUES ('Малдун');
```

```
INSERT INTO observer_tasks (observer_id, task_id)
VALUES (1, 1);
```

```
INSERT INTO spit_events (dilofoosaur_id, worker_id, description)
VALUES (1, 1, 'Плевкок ослепляющей слюной');
```

Вывод

В ходе работы я обновил свои знания использования базы данных. Попрактиковался с PostgreSQL. Ознакомился с архитектурой ANSI-SPARC а так же с DDL и DML. Изучил новые команды как CHECK, REFERENCES и тп. Обучился тому, как из текста составить описание предметной области.