



Федеральное государственное автономное образовательное

учреждение высшего образования

«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Отчёт по лабораторной работе №1

Дисциплина: Базы Данных

Вариант 53302

Выполнил: студент группы Р3115 Матвеева П.П.

Проверил: Райла Мартин

Дата сдачи:

г. Санкт-Петербург 2024

Содержание	
Задание.....	3
Описание предметной области.....	3
Сущности и их классификация	3
Инфологическая модель	4
Даталогическая модель	5
Вывод	5

Задание

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

Деннис Недри зевнул. Он давно уже понял, чем примерно должны заниматься в "ИнДжин". Несколько лет назад, когда Недри наняли в "ИнДжин" для того, чтобы он разработал системы управления Парком, он удивился, увидев, что согласно техническому заданию длина групп разрядов числовых данных должна достигать не менее 3×10^9 . Сначала Недри подумал, что тут какая-то ошибка, и позвонил в Пало-Альто. Однако там ему сказали, что требование правильное, действительно, три миллиарда разрядов чисел.

Существует сотрудник, у него есть имя (**Деннис Недри**), организация, в которой он работает («**ИнДжин**»), **должность**. Существует **проект**, которым он занимается (разработка систем управления Парком), по которому есть техническое задание с требованиями.

Существует город (**Пало-Альто**), в нем также есть организации. В «ИнДжин» также работают люди, которые наняли Денниса.

Существует **техническое задание**, которое дали Деннису люди из Пало-Альто.

Сущности и их классификация

Стержневые

- Сотрудник – имя, место работы, должность
- Город – название, организации
- Организация - название, работники, место
- Проект – название, описание, тз
- Динозавры – вид, описание, место жительства

Характеристические

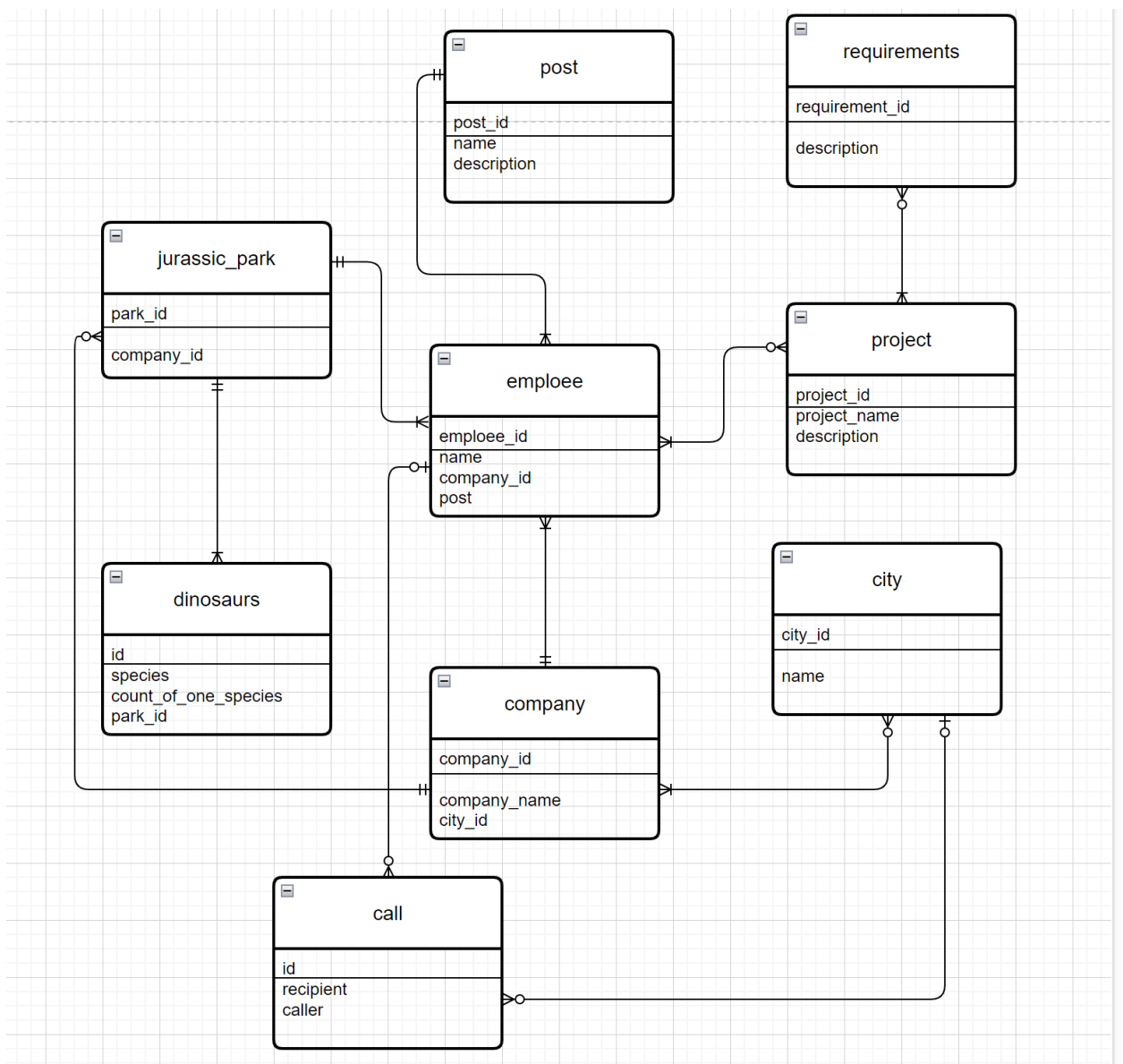
- Должность (характеристика) – название, описание
- Требования (описание технического задания)

- Звонок (звонок в Пало-Альто) – звонящий, получатель

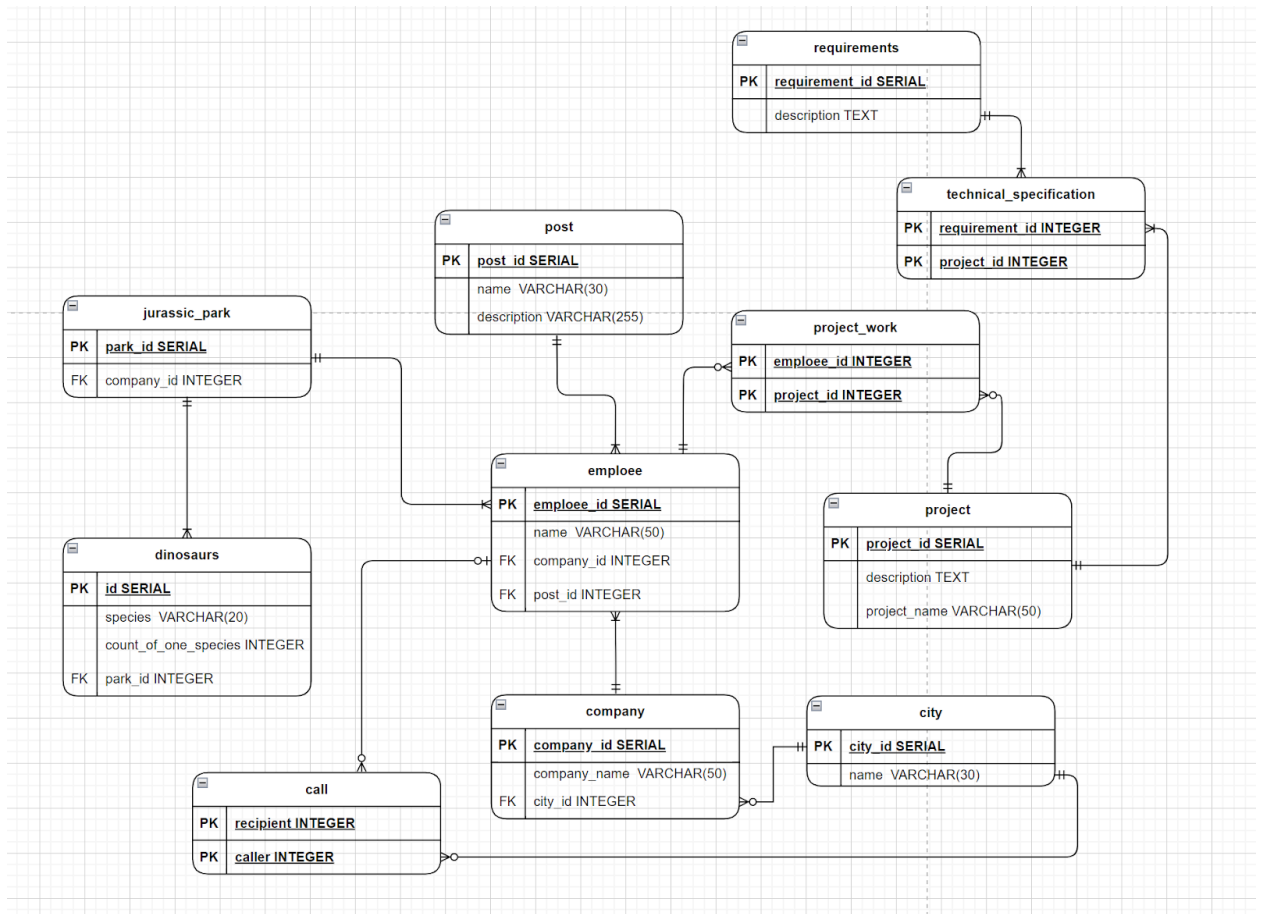
Ассоциативные

- Парк Юрского периода (динозавры, работники)
- Работа над проектом (сотрудник, проект).
- Техническое задание (проект, требования).
- Расположение компании(город, организация)

Инфологическая модель



Даталогическая модель



Реализация даталогической модели на языке SQL

BEGIN;

```
CREATE TYPE species AS ENUM ('velociraptor', 'spinosaurus', 'apatosaurus', 'triceratops', 'pentaceratops');
```

```
CREATE TABLE IF NOT EXISTS city(  
    city_id SERIAL PRIMARY KEY,  
    name VARCHAR(30) UNIQUE NOT NULL  
);
```

```
CREATE TABLE IF NOT EXISTS company(  
    company_id SERIAL PRIMARY KEY,  
    name VARCHAR(50) NOT NULL,  
    city_id INTEGER REFERENCES city(city_id)  
);
```

```
CREATE TABLE IF NOT EXISTS post(  
    post_id SERIAL PRIMARY KEY,  
    name VARCHAR(30) NOT NULL,  
    description VARCHAR(255)  
);
```

```
CREATE TABLE IF NOT EXISTS employee(  
    employee_id SERIAL PRIMARY KEY,  
    name VARCHAR(50) NOT NULL,  
    company_id INTEGER REFERENCES company(company_id) NOT NULL,  
    post_id INTEGER REFERENCES post(post_id) NOT NULL  
);
```

```
CREATE TABLE IF NOT EXISTS jurassic_park(  
    park_id SERIAL PRIMARY KEY,  
    company_id INTEGER REFERENCES company(company_id)  
);
```

```
CREATE TABLE IF NOT EXISTS dinosaurs(  
    id SERIAL PRIMARY KEY,  
    type species UNIQUE NOT NULL,  
    count_of_one_species INTEGER,  
    park_id INTEGER REFERENCES jurassic_park(park_id) NOT NULL  
);
```

```
CREATE TABLE IF NOT EXISTS company_location (
```

```
        company_id INTEGER REFERENCES company(company_id),
        city_id INTEGER REFERENCES city(city_id) NOT NULL,
        PRIMARY KEY (city_id, company_id)
    );
```

```
CREATE TABLE IF NOT EXISTS project(
    project_name VARCHAR(50) PRIMARY KEY,
    description TEXT
);
```

```
CREATE TABLE IF NOT EXISTS project_work(
    employee_id INTEGER REFERENCES employee(employee_id),
    project_name VARCHAR(50) REFERENCES project(project_name),
    PRIMARY KEY (employee_id, project_name)
);
```

```
CREATE TABLE IF NOT EXISTS requirements(
    requirement_id SERIAL PRIMARY KEY,
    description TEXT NOT NULL
);
```

```
CREATE TABLE IF NOT EXISTS technical_specification(
    requirement_id INTEGER REFERENCES requirements(requirement_id),
    project_name VARCHAR(50) REFERENCES project(project_name),
    PRIMARY KEY (requirement_id, project_name)
);
```

```
CREATE TABLE IF NOT EXISTS call(
    recipient INTEGER REFERENCES city(city_id),
    caller INTEGER REFERENCES employee(employee_id),
    PRIMARY KEY (recipient, caller)
);
```

INSERT INTO city(name) VALUES ('San Jose');

INSERT INTO city(name) VALUES ('Palo Alto');

INSERT INTO city(name) VALUES ('Turrialba');

INSERT INTO company(name, city_id) VALUES ('InGen', 1);

INSERT INTO company(name, city_id) VALUES ('DinoWorld', 2);

INSERT INTO company(name, city_id) VALUES ('Genesis', 2);

INSERT INTO post(name, description) VALUES ('Owner of Jurassic Park', 'A disgruntled employee responsible for the parks computer systems.');

INSERT INTO post(name, description) VALUES ('Paleontologist', 'A renowned dinosaur specialist invited to the park to assess its safety and scientific value');

INSERT INTO post(name, description) VALUES ('Paleobotanist ', 'invited to the park to study its plants and ecosystem.');

INSERT INTO post(name, description) VALUES ('IT Specialist', 'A disgruntled employee responsible for the parks computer systems.');

INSERT INTO post(name, description) VALUES ('Chief Engineer', 'Responsible for the technical infrastructure of the park');

INSERT INTO employee (name, company_id, post_id) VALUES ('Dennis Nedry', 1, 4);

INSERT INTO employee (name, company_id, post_id) VALUES ('John Hammond', 1, 1);

INSERT INTO employee (name, company_id, post_id) VALUES ('Ray Arnold', 1, 3);

INSERT INTO employee (name, company_id, post_id) VALUES ('Alan Grant', 2, 2);

INSERT INTO employee (name, company_id, post_id) VALUES ('Ellie Sattler', 2, 4);

INSERT INTO employee (name, company_id, post_id) VALUES ('Ian Malcolm', 2, 5);

INSERT INTO jurassic_park(company_id) VALUES (1);

INSERT INTO jurassic_park(company_id) VALUES (2);

INSERT INTO dinosaurs(type, count_of_one_species, park_id) VALUES ('velociraptor', 67, 1);

INSERT INTO dinosaurs(type, count_of_one_species, park_id) VALUES ('triceratops', 129, 1);

INSERT INTO dinosaurs(type, count_of_one_species, park_id) VALUES ('pentaceratops', 4, 1);


```
INSERT INTO project (project_name, description) VALUES ('Jurassic Park Control Systems',  
'Development of control systems for Jurassic Park, including security and dinosaur  
management.');
```

```
INSERT INTO project (project_name, description) VALUES ('Genetic Research Integration',  
'Integration of genetic research data into the park's operational systems.');
```

```
INSERT INTO project_work(employee_id, project_id) VALUES (5, 2);
```

```
INSERT INTO project_work(employee_id, project_id) VALUES (3, 1);
```

```
INSERT INTO project_work(employee_id, project_id) VALUES (2, 2);
```

```
INSERT INTO project_work(employee_id, project_id) VALUES (4, 2);
```

```
INSERT INTO project_work(employee_id, project_id) VALUES (1, 1);
```

```
INSERT INTO requirements (description) VALUES ('Develop a control system capable of  
handling numeric data groups with a length of at least  $3 \times 10^9$  digits.');
```

```
INSERT INTO requirements (description) VALUES('Ensure the system can manage real-time  
data processing for dinosaur tracking.');
```

```
INSERT INTO requirements (description) VALUES('Integrate genetic data storage with a  
capacity of at least 1 PB.');
```

```
INSERT INTO requirements (description) VALUES('Implement security protocols to prevent  
unauthorized access to genetic data.');
```

```
COMMIT;
```

Доп

У динозавров через время отпадают части тела, на месте отпадающих частей тела начинают расти другие динозавры, у которых также могут отпадать части тел (они падают на землю = состояние)

Вывод

Научилась строить инфологическую и даталогическую модели, работать на SQL,