

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

Факультет Программной Инженерии и Компьютерной Техники

Лабораторная работа №1

Вариант 31063

Выполнил:

Бен Шамех Абделаиз

Группа Р3106

Преподаватели:

Вербавой Александр

Санкт-Петербург 2025

Содержание

Текст задания	3
Описание предметной области.....	3
Список сущностей и их классификация.....	3
Инфологическая модель.....	4
Даталогическая модель.....	4
Реализация даталогической модели на SQL.....	5
Выводы по работе.....	7

Текст задания

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

Прямоугольник впереди посветлел. Яркие черточки звезд тускнели на фоне млечного неба, которое сияло все сильнее. Казалось, капсула летит к скоплению облаков, равномерно освещенному лучами невидимого солнца.

Жители Диаспара забыли прошлое, но не осознали этого. Их мир был совершенным, самодостаточным и неизменным — так и задумано. За стенами города в их сознании ничего не существовало; звезды, когда-то доступные человечеству, теперь были просто мифами. Но в глубинах банков памяти Диаспара сохранилось скрытое знание — ожидающее, когда кто-то вспомнит.

Список сущностей и их классификация

Стержневые:

- Объект(Object) – название, тип, форма

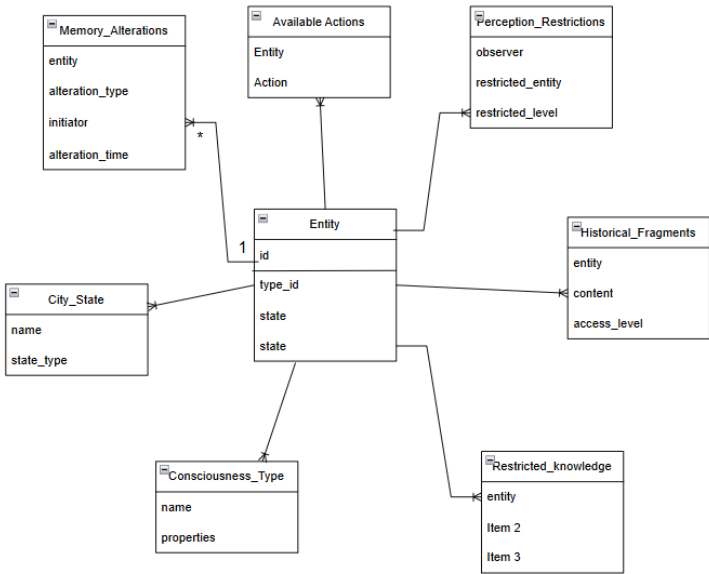
Ассоциативные:

- Доступные действия (Available actions) – *объект, действие*
- Изменения памяти (Memory Alternations) – сущность, тип изменения, инициатор, время изменения
- Ограниченные знания (Restricted Knowledge) – сущность, тип ограничения

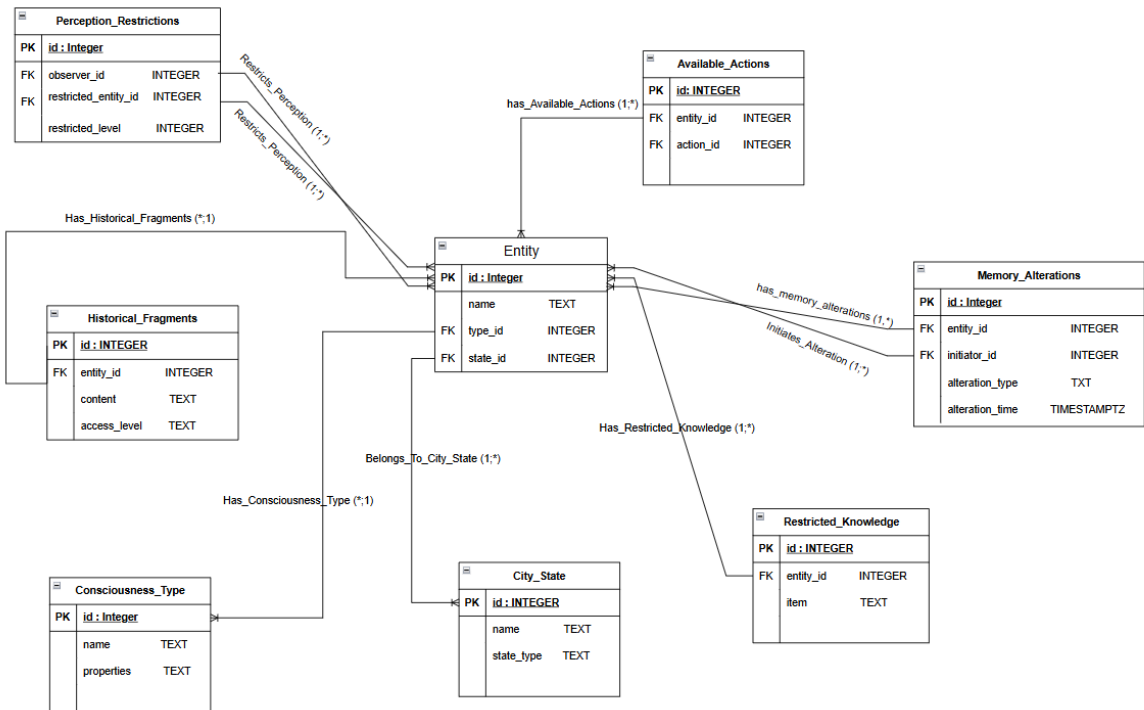
Характеристические:

- город состояние (City State) – *название, тип города*
- Тип сознания (Consciousness Type) – название, свойства
- Ограничения восприятия (Perception Restrictions) – наблюдатель, ограниченная сущность, уровень ограничения
- Исторические фрагменты (Historical Fragments) – *сущность, содержание, уровень доступа*

Инфологическая модель



Даталогическая модель



Реализация даталогической модели на SQL

BEGIN;

-- Table for Entity Types (Consciousness Type)

```
CREATE TABLE IF NOT EXISTS Consciousness_Type (  
  id SERIAL PRIMARY KEY,  
  name TEXT UNIQUE NOT NULL,  
  properties TEXT  
);
```

-- Table for City States

```
CREATE TABLE IF NOT EXISTS City_State (  
  id SERIAL PRIMARY KEY,  
  name TEXT UNIQUE NOT NULL,  
  state_type TEXT  
);
```

-- Main Entity Table

```
CREATE TABLE IF NOT EXISTS Entity (  
  id SERIAL PRIMARY KEY,  
  name TEXT NOT NULL DEFAULT 'Unknown',  
  type_id INTEGER REFERENCES Consciousness_Type(id),  
  state_id INTEGER REFERENCES City_State(id)  
);
```

-- Table for Memory Alterations

```
CREATE TABLE IF NOT EXISTS Memory_Alterations (  
  id SERIAL PRIMARY KEY,  
  entity_id INTEGER NOT NULL REFERENCES Entity(id),  
  initiator_id INTEGER REFERENCES Entity(id),  
  alteration_type TEXT,  
  alteration_time TIMESTAMPTZ NOT NULL DEFAULT NOW()  
);
```

-- Table for Available Actions

```
CREATE TABLE IF NOT EXISTS Available_Actions (  
  entity_id INTEGER NOT NULL REFERENCES Entity(id),  
  action_id INTEGER NOT NULL,  
  PRIMARY KEY(entity_id, action_id)  
);
```

-- Table for Perception Restrictions

```
CREATE TABLE IF NOT EXISTS Perception_Restrictions (  
  id SERIAL PRIMARY KEY,  
  observer_id INTEGER NOT NULL REFERENCES Entity(id),  
  restricted_entity_id INTEGER NOT NULL REFERENCES Entity(id),  
  restricted_level TEXT
```

);

-- Table for Historical Fragments

```
CREATE TABLE IF NOT EXISTS Historical_Fragments (  
  id SERIAL PRIMARY KEY,  
  entity_id INTEGER NOT NULL REFERENCES Entity(id),  
  content TEXT,  
  access_level TEXT  
);
```

-- Table for Restricted Knowledge

```
CREATE TABLE IF NOT EXISTS Restricted_Knowledge (  
  id SERIAL PRIMARY KEY,  
  entity_id INTEGER NOT NULL REFERENCES Entity(id),  
  item TEXT  
);
```

-- INSERT SAMPLE DATA

-- Insert City States

```
INSERT INTO City_State (name, state_type)  
VALUES  
  ('Metropolis', 'Urban'),  
  ('Outpost 12', 'Space Station'),  
  ('Ancient Ruins', 'Historical');
```

-- Insert Consciousness Types

```
INSERT INTO Consciousness_Type (name, properties)  
VALUES  
  ('Artificial Intelligence', 'Self-learning, logical'),  
  ('Human', 'Emotional, social'),  
  ('Alien', 'Unknown properties');
```

-- Insert Entities

```
INSERT INTO Entity (name, type_id, state_id)  
VALUES  
  ('ECHO-1', 1, 2),  
  ('Dr. Leon', 2, 1),  
  ('Zyraxian Ambassador', 3, 3);
```

-- Insert Memory Alterations

```
INSERT INTO Memory_Alterations (entity_id, initiator_id, alteration_type)  
VALUES  
  (1, 2, 'Memory Implant'),  
  (2, 1, 'Memory Erasure');
```

-- Insert Perception Restrictions

```
INSERT INTO Perception_Restrictions (observer_id, restricted_entity_id, restricted_level)  
VALUES
```

```
(1, 3, 'Classified');
```

```
-- Insert Historical Fragments
```

```
INSERT INTO Historical_Fragments (entity_id, content, access_level)
```

```
VALUES
```

```
(3, 'Ancient Zyraxian war records', 'Restricted');
```

```
-- Insert Restricted Knowledge
```

```
INSERT INTO Restricted_Knowledge (entity_id, item)
```

```
VALUES
```

```
(2, 'Quantum Drive Blueprints');
```

```
END;
```

Выводы по работе

В ходе данной работы я научился проектировать базу данных по принципу “Up-Down”, научился составлять инфологическую и даталогическую модели сущностей, по которым реализовал базу данных с помощью SQL.