

**Федеральное государственное автономное образовательное
учреждение высшего образования
Университет ИТМО**

Дисциплина: Информационные системы и базы данных
Лабораторная работа 1
Вариант 1236

Выполнил:
ФИО: Энох Асанте Ларби
Группа: Р3131
Преподаватель: Коновало Арсений Антонович

2025 г. Санкт-Петербург

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.

Для создания объектов базы данных у каждого студента есть своя схема. Название схемы соответствует имени пользователя в базе studs (sXXXXXX). Команда для подключения к базе studs:

```
psql -h pg -d studs
```

Каждый студент должен использовать свою схему при работе над лабораторной работой №1 (а также в рамках выполнения 2, 3 и 4 этапа курсовой работы).

Отчёт по лабораторной работе должен содержать:

1. Текст задания.
2. Описание предметной области.
3. Список сущностей и их классификацию (стержневая, ассоциация, характеристика).
4. Инфологическая модель (ER-диаграмма в расширенном виде - с атрибутами, ключами...).
5. Даталогическая модель (должна содержать типы атрибутов, вспомогательные таблицы для отображения связей "многие-ко-многим").
6. Реализация даталогической модели на SQL.
7. Выводы по работе.

Темы для подготовки к защите лабораторной работы:

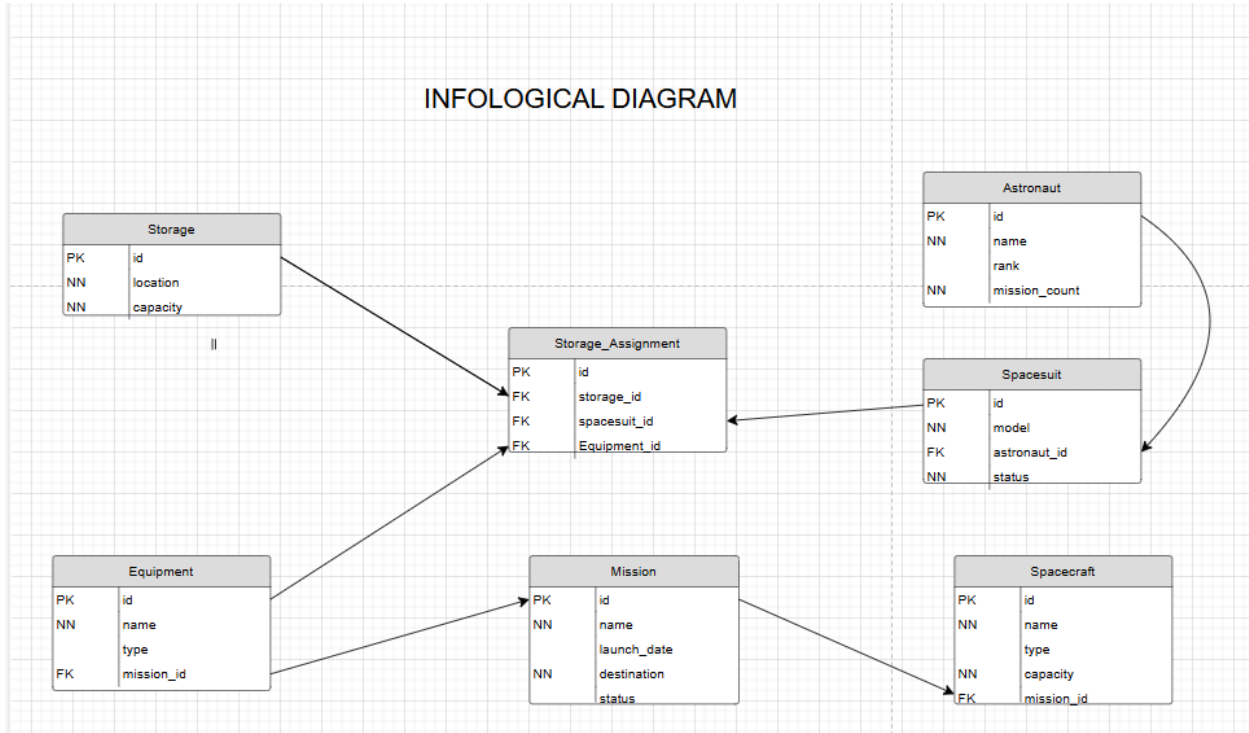
1. Архитектура ANSI-SPARC
2. Модель "Сущность-Связь". Классификация сущностей. Виды связей. Ограничения целостности.
3. DDL
4. DML

Введите вариант:

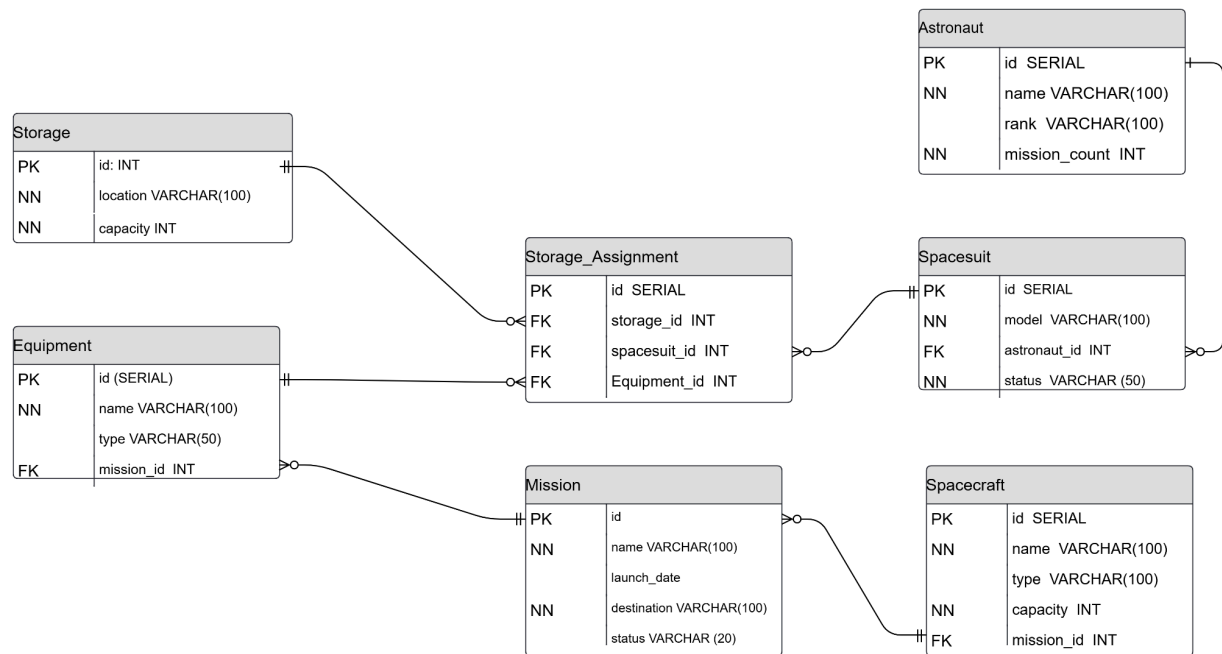
Описание предметной области, по которой должна быть построена доменная модель:

Боумен бросил шлем на кровать, весело, хотя и весьма неуклюже, начал стаскивать с себя скафандр. Покончив с этим, он потянулся, сделал несколько глубоких вдохов и бережно повесил скафандр в стеной шкаф рядом с другой, более обычной одеждой. В соседстве с пиджаками скафандр выглядел не особенно уместным, но свойственная всем астронавтам аккуратность не позволила Боумену бросить его где попало.

Summary of Relationships		
Relationship Type	Entities Involved	Description
One-to-One	Astronaut ↔ Spacesuit	Each astronaut has one spacesuit, and each spacesuit is assigned to one astronaut.
One-to-Many	Mission ↔ Spacecraft	One mission can have one spacecraft, and each spacecraft is assigned to one mission.
One-to-Many	Mission ↔ Equipment	One mission can have many pieces of equipment, and each piece of equipment is assigned to one mission.
Many-to-Many	Storage ↔ Spacesuit	A storage location can hold many spacesuits, and a spacesuit can be stored in many locations.
Many-to-Many	Storage ↔ Equipment	A storage location can hold many pieces of equipment, and a piece of equipment can be stored in many locations.



DATALOAGICAL DIAGRAM



```
-- Step 1: Drop Tables if They Exist
```

```
DROP TABLE IF EXISTS Storage_Assignment;
```

```
DROP TABLE IF EXISTS Storage;
```

```
DROP TABLE IF EXISTS Equipment;
```

```
DROP TABLE IF EXISTS Spacecraft;
```

```
DROP TABLE IF EXISTS Spacesuit;
```

```
DROP TABLE IF EXISTS Mission;
```

```
DROP TABLE IF EXISTS Astronaut;
```

```
-- Step 2: Create Tables
```

```
-- Create Astronaut Table
```

```
CREATE TABLE Astronaut (
    id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    rank VARCHAR(50),
    mission_count INT DEFAULT 0
);
```

```

-- Create Mission Table
CREATE TABLE Mission (
    id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    launch_date DATE,
    destination VARCHAR(100),
    status VARCHAR(20) CHECK (status IN ('active', 'completed', 'failed'))
);

-- Create Spacesuit Table (One-to-One with Astronaut)
CREATE TABLE Spacesuit (
    id SERIAL PRIMARY KEY,
    model VARCHAR(50),
    assigned_astronaut_id INT UNIQUE REFERENCES Astronaut(id) ON DELETE
SET NULL, -- 1:1 relationship
    condition_status VARCHAR(50) CHECK (condition_status IN ('new',
'used', 'damaged'))
);

-- Create Spacecraft Table (One-to-Many with Mission)
CREATE TABLE Spacecraft (
    id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    type VARCHAR(50),
    capacity INT CHECK (capacity > 0),
    assigned_mission_id INT REFERENCES Mission(id) ON DELETE SET NULL --
1:N relationship
);

-- Create Equipment Table (One-to-Many with Mission)
CREATE TABLE Equipment (
    id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    type VARCHAR(50),
    assigned_mission_id INT REFERENCES Mission(id) ON DELETE SET NULL --
1:N relationship
);

-- Create Storage Table
CREATE TABLE Storage (

```

```

        id SERIAL PRIMARY KEY,
        location VARCHAR(100),
        capacity INT CHECK (capacity > 0)
    );

-- Create Storage Assignment Table (Many-to-Many relationship)
CREATE TABLE Storage_Assignment (
    id SERIAL PRIMARY KEY,
    storage_id INT REFERENCES Storage(id) ON DELETE CASCADE,
    spacesuit_id INT REFERENCES Spacesuit(id) ON DELETE SET NULL,
    equipment_id INT REFERENCES Equipment(id) ON DELETE SET NULL
);

-- Step 3: Insert Sample Data

-- Insert into Astronaut
INSERT INTO Astronaut (name, rank, mission_count) VALUES
('David Bowman', 'Commander', 3),
('Frank Poole', 'Lieutenant', 2);

-- Insert into Mission
INSERT INTO Mission (name, launch_date, destination, status) VALUES
('Jupiter Mission', '2001-06-12', 'Jupiter', 'active'),
('Mars Exploration', '2025-04-15', 'Mars', 'completed');

-- Insert into Spacesuit (One-to-One with Astronaut)
INSERT INTO Spacesuit (model, assigned_astronaut_id, condition_status)
VALUES
('MK-1', 1, 'new'), -- Assigned to David Bowman
('MK-2', 2, 'used'); -- Assigned to Frank Poole

-- Insert into Spacecraft (One-to-Many with Mission)
INSERT INTO Spacecraft (name, type, capacity, assigned_mission_id) VALUES
('Discovery One', 'Deep Space Vessel', 5, 1), -- Assigned to Jupiter
Mission
('Odyssey', 'Mars Rover', 2, 2); -- Assigned to Mars Exploration

-- Insert into Equipment (One-to-Many with Mission)
INSERT INTO Equipment (name, type, assigned_mission_id) VALUES
('Oxygen Tank', 'Life Support', 1), -- Assigned to Jupiter Mission

```

```

('Communication Device', 'Electronics', 2); -- Assigned to Mars
Exploration

-- Insert into Storage
INSERT INTO Storage (location, capacity) VALUES
('Cabin A', 10),
('Cabin B', 5);

-- Insert into Storage_Assignment (Many-to-Many relationship)
INSERT INTO Storage_Assignment (storage_id, spacesuit_id, equipment_id)
VALUES
(1, 1, NULL), -- Spacesuit MK-1 stored in Cabin A
(2, NULL, 1); -- Oxygen Tank stored in Cabin B

-- Step 3: Example Queries

-- Query 1: One-to-One (Get astronauts and their spacesuits)
SELECT a.name AS astronaut, s.model AS spacesuit
FROM Astronaut a
JOIN Spacesuit s ON a.id = s.assigned_astronaut_id;

-- Query 2: One-to-Many (Get missions and their spacecraft)
SELECT m.name AS mission, sc.name AS spacecraft
FROM Mission m
JOIN Spacecraft sc ON m.id = sc.assigned_mission_id;

-- Query 3: One-to-Many (Get missions and their equipment)
SELECT m.name AS mission, e.name AS equipment
FROM Mission m
JOIN Equipment e ON m.id = e.assigned_mission_id;

-- Query 4: Many-to-Many (Get storage locations and their contents)
SELECT st.location AS storage,
       COALESCE(s.model, 'No Spacesuit') AS spacesuit,
       COALESCE(e.name, 'No Equipment') AS equipment
FROM Storage_Assignment sa
JOIN Storage st ON sa.storage_id = st.id
LEFT JOIN Spacesuit s ON sa.spacesuit_id = s.id
LEFT JOIN Equipment e ON sa.equipment_id = e.id;

```

```
-- Query 5: Get all equipment stored in Cabin A
SELECT e.name AS equipment
FROM Storage_Assignment sa
JOIN Equipment e ON sa.equipment_id = e.id
JOIN Storage st ON sa.storage_id = st.id
WHERE st.location = 'Cabin A';
```

Results 👍

```
[s407885@helios ~]$ psql -h pg -d studs -U s407885 -f PgDBScript.sql
```

```
DROP TABLE
DROP TABLE
DROP TABLE
DROP TABLE
DROP TABLE
DROP TABLE
DROP TABLE
DROP TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
```

```
INSERT 0 2
INSERT 0 2
INSERT 0 2
INSERT 0 2
INSERT 0 2
INSERT 0 2
INSERT 0 2
astronaut | spacesuit
```

```
-----+-----
David Bowman | MK-1
Frank Poole | MK-2
(2 строки)
```

```
mission | spacecraft
-----+-----
Jupiter Mission | Discovery One
Mars Exploration | Odyssey
(2 строки)
```

```
mission | equipment
-----+-----
Jupiter Mission | Oxygen Tank
Mars Exploration | Communication Device
(2 строки)
```

```
storage | spacesuit | equipment
-----+-----+-----
```


Cabin A | MK-1 | No Equipment
Cabin B | No Spacesuit | Oxygen Tank
(2 строки)

equipment

(0 строк)

[s407885@helios ~]\$

Вывод

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.