

УНИВЕРСИТЕТ ИТМО

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.04 Программная инженерия

Дисциплина «Базы данных»

**Лабораторная работа №2**

**Вариант 2937**

Студент

*Сафонова А. О., Р33201*

Преподаватель

*Машина Е. А.*

Санкт-Петербург, 2022 г.

## **Задание**

Для выполнения лабораторной работы №2 необходимо:

- 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**

Поначалу это удивляло и даже пугало, потом все объяснилось. Любое торможение ведет к нагреву; мощные токи превращали корабль в своеобразную электропечь. "Дискавери" бросало то в жар, то в холод на протяжении нескольких лет - неудивительно, что продукты на борту не проявили достаточной стойкости.

### **Описание:**

Космический корабль - это транспортное средство или машина, предназначенная для полетов в космическом пространстве.

Персонал - весь личный состав работников (включая постоянных и временных), состоящих с организацией как юридическим лицом в отношениях, регулируемых договором о найме.

Пассажир - это лицо, которое путешествует в транспортном средстве, но не несет или не несет никакой ответственности за выполнение задач, необходимых для того, чтобы это транспортное средство прибыло в пункт назначения или иным образом управляло транспортным средством.

Билет - документ, удостоверяющий наличие некоего права у какого-либо определённого лица или у предъявителя билета.

Оборудование - это совокупность связанных между собой частей или устройств.

Продукты - продукты в натуральном или переработанном виде, употребляемые человеком в пищу, бутилированная питьевая вода, алкогольные и безалкогольные напитки, жевательная резинка, а также продовольственное сырьё, пищевые добавки и биологически активные добавки.

Температура - физическая величина, характеризующая термодинамическую систему и количественно выражающая интуитивное понятие о различной степени нагретости тел.

Скорость - векторная физическая величина, характеризующая быстроту перемещения и направление движения материальной точки относительно выбранной системы отсчёта.

### **Бизнес-процессы:**

С данной информационной системой должны работать следующие группы пользователей:

1. Администратор;
2. Персонал.

При работе с системой Администратор должен иметь возможность решать следующие задачи:

1. Добавление, редактирование и удаление кораблей из базы;
2. Добавление, редактирование и удаление оборудования;
3. Добавление, редактирование и удаление персонала;
4. Добавление, редактирование и удаление продуктов;
5. Просмотр данных об изменении скорости;
6. Просмотр данных об изменении температуры;
7. Просмотр данных о билетах;
8. Просмотр и редактирование данных о пассажирах;
9. Оформление и согласование заявок на ремонт оборудования корабля.

При работе с системой Персонал должен иметь возможность решать следующие задачи:

1. Добавление и удаление продуктов;
2. Просмотр данных о билетах;
3. Просмотр данных о пассажирах;
4. Просмотр данных об изменении скорости;
5. Просмотр данных об изменении температуры;
6. Оформление заявок на ремонт оборудования корабля.

### **Сущности и атрибуты**

Космический корабль (стержневая). Атрибуты:

1. id корабля,
2. название корабля,
3. дата вылета.

Персонал (характеристика). Атрибуты:

1. id сотрудника,
2. фамилия,

3. имя,
4. отчество,
5. дата рождения,
6. дата найма,
7. должность,
8. id корабля, на котором он работает.

Пассажир (характеристика). Атрибуты:

1. id пассажира,
2. фамилия,
3. имя,
4. отчество,
5. дата рождения.

Билет (характеристика). Атрибуты:

1. id билета,
2. id пассажира,
3. id корабля,
4. пункт отправления,
5. пункт прибытия,
6. дата покупки,
7. стоимость.

Оборудование (характеристика). Атрибуты:

1. id оборудования,
2. id корабля,
3. название,
4. дата последнего обслуживания,
5. состояние.

Продукты (характеристика). Атрибуты:

1. id продукта,
2. id корабля,
3. название,
4. дата изготовления,
5. срок годности,
6. температура хранения.

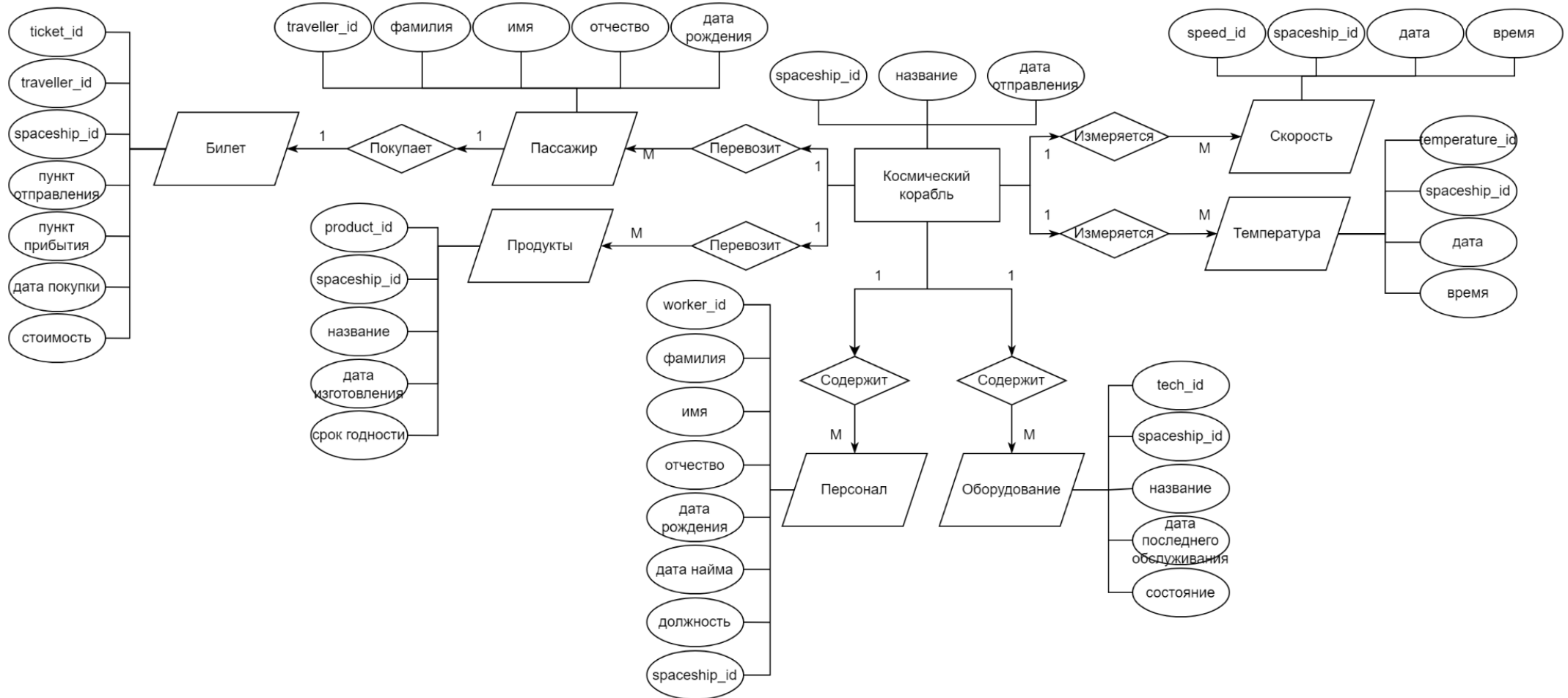
Температура (характеристика). Атрибуты:

1. id измерения,
2. id корабля,
3. дата,
4. время.

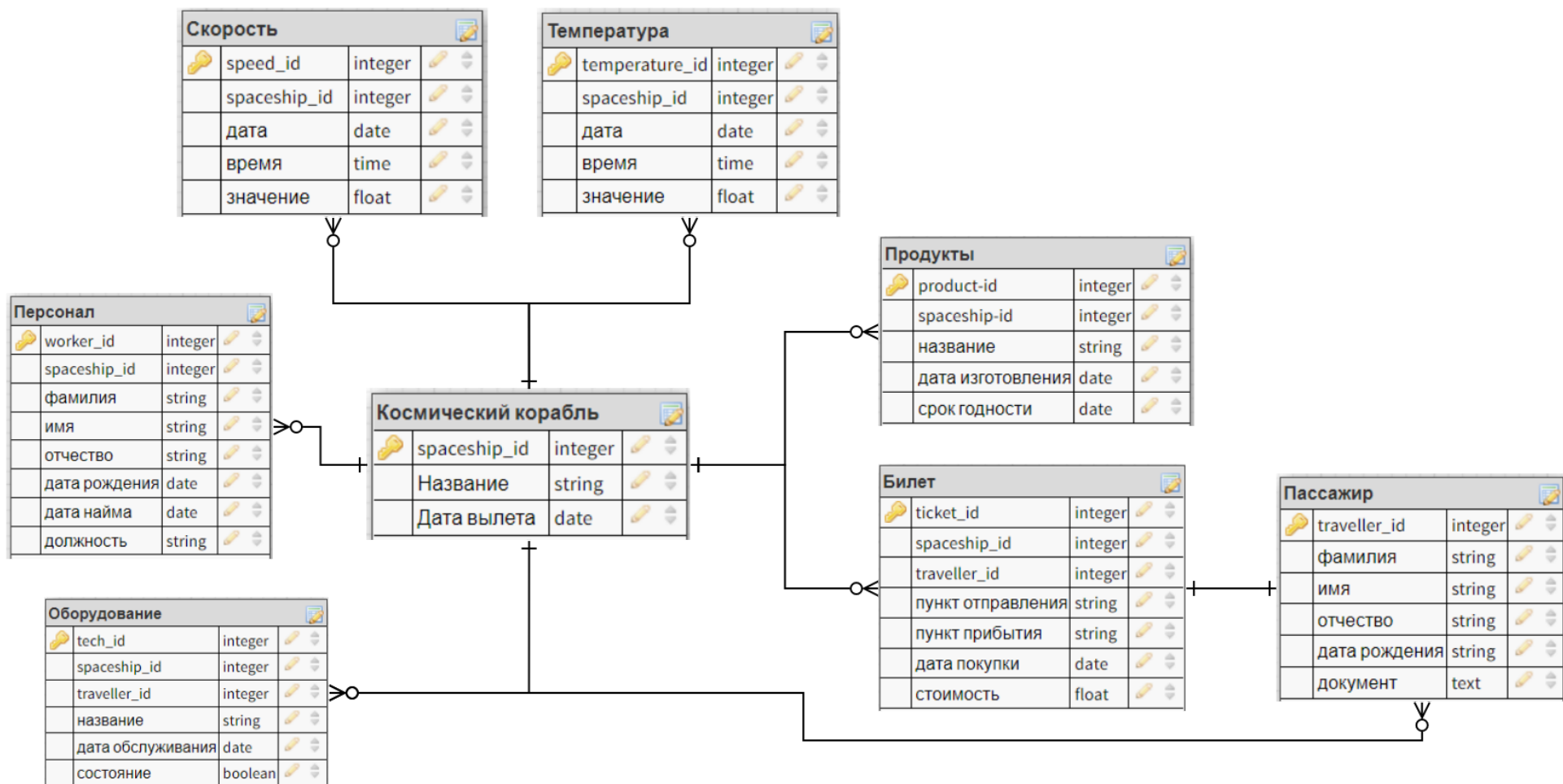
Скорость (характеристика). Атрибуты:

1. id измерения,
2. id корабля,
3. дата,
4. время,
5. изменение температуры.

## Инфологическая модель



## Даталогическая модель





## Реализация

```
CREATE TABLE "public.Космический корабль" (  
    "spaceship_id" serial NOT NULL,  
    "Название" VARCHAR(255) NOT NULL UNIQUE,  
    "Дата вылета" DATE NOT NULL,  
    CONSTRAINT "Космический корабль_pk" PRIMARY KEY  
    ("spaceship_id")  
    ) WITH (  
    OIDS=FALSE  
    );
```

```
CREATE TABLE "public.Оборудование" (  
    "tech_id" serial NOT NULL,  
    "spaceship_id" integer REFERENCES "public.Космический корабль"  
    ("spaceship_id"),  
    "название" VARCHAR(255) NOT NULL,  
    "дата обслуживания" DATE NOT NULL,  
    "состояние" BOOLEAN NOT NULL,  
    CONSTRAINT "Оборудование_pk" PRIMARY KEY ("tech_id")  
    ) WITH (  
    OIDS=FALSE  
    );
```

```
CREATE TABLE "public.Скорость" (  

```

```
"speed_id" serial NOT NULL,  
  
"spaceship_id" integer REFERENCES "public.Космический корабль"  
("spaceship_id"),  
  
"дата" DATE NOT NULL,  
  
"время" TIME NOT NULL,  
  
"значение" FLOAT,  
  
CONSTRAINT "Скорость_pk" PRIMARY KEY ("speed_id")  
  
) WITH (  
  
  OIDS=FALSE  
  
);
```

```
CREATE TABLE "public.Температура" (  
  
  "temperature_id" serial NOT NULL,  
  
  "spaceship_id" integer REFERENCES "public.Космический корабль"  
("spaceship_id"),  
  
  "дата" DATE NOT NULL,  
  
  "время" TIME NOT NULL,  
  
  "значение" FLOAT,  
  
  CONSTRAINT "Температура_pk" PRIMARY KEY ("temperature_id")  
  
) WITH (  
  
  OIDS=FALSE  
  
);
```

```
CREATE TABLE "public.Продукты" (  
  
  "product-id" serial NOT NULL,
```

```
        "spaceship_id" integer REFERENCES "public.Космический корабль"
("spaceship_id"),

        "название" VARCHAR(255) NOT NULL,

        "дата изготовления" DATE NOT NULL,

        "срок годности" DATE NOT NULL,

        CONSTRAINT "Продукты_pk" PRIMARY KEY ("product-id")

) WITH (

    OIDS=FALSE

);
```

```
CREATE TABLE "public.Персонал" (

    "worker_id" serial NOT NULL,

    "spaceship_id" integer REFERENCES "public.Космический корабль"
("spaceship_id"),

    "фамилия" VARCHAR(255) NOT NULL,

    "имя" VARCHAR(255) NOT NULL,

    "отчество" VARCHAR(255) NOT NULL,

    "дата рождения" DATE NOT NULL,

    "дата найма" DATE NOT NULL,

    "должность" VARCHAR(255) NOT NULL,

    CONSTRAINT "Персонал_pk" PRIMARY KEY ("worker_id")

) WITH (

    OIDS=FALSE

);
```

```
CREATE TABLE "public.Пассажир" (  
    "traveller_id" serial NOT NULL,  
    "фамилия" VARCHAR(255) NOT NULL,  
    "имя" VARCHAR(255) NOT NULL,  
    "отчество" VARCHAR(255) NOT NULL,  
    "дата рождения" DATE NOT NULL,  
    "документ" VARCHAR(255) NOT NULL,  
    CONSTRAINT "Пассажир_pk" PRIMARY KEY ("traveller_id")  
) WITH (  
    OIDS=FALSE  
);
```

```
CREATE TABLE "public.Билет" (  
    "ticket_id" serial NOT NULL,  
    "spaceship_id" serial REFERENCES "public.Космический корабль"  
("spaceship_id"),  
    "traveller_id" serial REFERENCES "public.Пассажир" ("traveller_id"),  
    "пункт отправления" VARCHAR(255) NOT NULL,  
    "пункт прибытия" VARCHAR(255) NOT NULL,  
    "дата покупки" DATE NOT NULL,  
    "стоимость" FLOAT NOT NULL,  
    CONSTRAINT "Билет_pk" PRIMARY KEY ("ticket_id")  
) WITH (  
    OIDS=FALSE
```

);

### **Наполнение таблиц**

```
INSERT INTO "public.Космический корабль" VALUES
```

```
(1, 'Дискавери', '03.03.2023');
```

```
INSERT INTO "public.Космический корабль" VALUES
```

```
(2, 'Навуходоносор', '01.02.2012');
```

```
INSERT INTO "public.Оборудование" VALUES
```

```
(1, 1, 'Камера', '02.03.2010', TRUE);
```

```
INSERT INTO "public.Оборудование" VALUES
```

```
(2, 1, 'Холодильник', '02.03.2010', TRUE);
```

```
INSERT INTO "public.Оборудование" VALUES
```

```
(3, 1, 'Утюг', '02.03.2010', FALSE);
```

```
INSERT INTO "public.Скорость" VALUES
```

```
(1, 1, '03.03.2023', '12:00:02', 1234.00);
```

```
INSERT INTO "public.Температура" VALUES
```

```
(1, 1, '03.03.2023', '12:00:02', 273);
```

```
INSERT INTO "public.Продукты" VALUES
```

```
(1, 1, 'Вода', '01.03.2023', '01.03.2026');
```

```
INSERT INTO "public.Продукты" VALUES
```

```
(2, 1, 'Гречка', '01.03.2023', '01.03.2026');
```

```
INSERT INTO "public.Продукты" VALUES
```

```
(3, 1, 'Рис', '01.03.2023', '01.03.2026');
```

```
INSERT INTO "public.Персонал" VALUES
```

(1, 1, 'Иванов', 'Иван', 'Иванович', '01.01.1990', '16.01.2018', 'Пилот');

INSERT INTO "public.Персонал" VALUES

(2, 1, 'Петров', 'Петр', 'Петрович', '02.02.1991', '16.02.2018', 'Второй пилот');

INSERT INTO "public.Персонал" VALUES

(3, 1, 'Васильева', 'Мария', 'Викторовна', '03.03.1992', '19.01.2018', 'Повар');

INSERT INTO "public.Персонал" VALUES

(4, 1, 'Кимчи', 'Евгения', 'Васильевна', '04.04.1993', '16.03.2018', 'Слесарь');

INSERT INTO "public.Пассажир" VALUES

(1, 'Ди', 'Леонардо', 'Каприо', '02.08.1978', 'паспорт серия 0000 номер 123456');

INSERT INTO "public.Пассажир" VALUES

(2, 'Карлов', 'Карл', 'Карлович', '07.12.1989', 'паспорт серия 1010 номер 039827');

INSERT INTO "public.Билет" VALUES

(1, 1, 1, 'Земля', 'Юпитер', '02.08.2020', 1284903);

INSERT INTO "public.Билет" VALUES

(2, 1, 2, 'Земля', 'Юпитер', '07.12.2021', 2849031);

ALTER TABLE "public.Оборудование" ADD COLUMN traveller\_id serial  
REFERENCES "public.Пассажир" ("traveller\_id");

INSERT INTO "public.Оборудование" VALUES

(4, 1, 'Камера', '10.12.2023', TRUE, 2);

INSERT INTO "public.Оборудование" VALUES

(5, 1, 'Утюг', '11.12.2023', FALSE, 2);



## **Выводы**

В данной лабораторной работе были разработаны на основе текста описание предметной области, список сущностей, их классификация и их атрибуты, созданы инфологическая модель и даталогическая модель, которая реализована на SQL.