

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»
(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных
систем

Лабораторная работа №1
по дисциплине: Базы данных
тема: **Разработка структуры базы данных**

Выполнил: студент ПВ-233
Мороз Роман Алексеевич

Проверил:
Панченко Максим Владимирович

Белгород 2025 г.

Цель работы: изучение способов задания инфологической модели данных и создания структуры базы данных в заданной предметной области.

Вариант 9. Гарантийный ремонт. База данных должна содержать следующие данные: информацию о выпускаемых производителем товарах, гарантийных мастерских в разных городах и товарах, ремонт которых они могут производить, данные о выполненных ремонтах. Предусмотреть возможность анализа следующих показателей: рейтинг убыточности товаров, обеспеченность каждого города мастерскими по разным группам товаров и всему ассортименту в целом.

Задания к работе

1. Выполнить анализ предметной области, выделить основные сущности, атрибуты и связи.

Основные сущности и их атрибуты:

1. Производитель (Manufacturer)

- *manufacturer_id* — Уникальный идентификатор производителя (Первичный ключ).
- *name* — Наименование производителя.
- *contact_info* — Контактная информация (адрес, телефон и т.д.).

2. Товар (Product)

- *product_id* — Уникальный идентификатор товара (Первичный ключ).
- *manufacturer_id* — Идентификатор производителя (Внешний ключ).
- *model_name* — Наименование модели товара.
- *category* — Категория/группа товара (например, "смартфоны", "холодильники").
- *warranty_period* — Гарантийный период (в днях или месяцах).
- *base_repair_cost* — Базовая стоимость ремонта (для расчета убыточности).

3. Город (City)

- *city_id* — Уникальный идентификатор города (Первичный ключ).
- *name* — Название города.

4. Мастерская (RepairShop)

- *shop_id* — Уникальный идентификатор мастерской (Первичный ключ).
- *city_id* — Идентификатор города, где находится мастерская (Внешний ключ).
- *address* — Адрес мастерской.
- *contact_phone* — Контактный телефон.

5. Специализация мастерской (ShopSpecialization)

- *specialization_id* — Уникальный идентификатор специализации (Первичный ключ).
- *shop_id* — Идентификатор мастерской (Внешний ключ).
- *product_id* — Идентификатор товара (группы товаров), который может ремонтировать мастерская (Внешний ключ).

6. Ремонт (Repair)

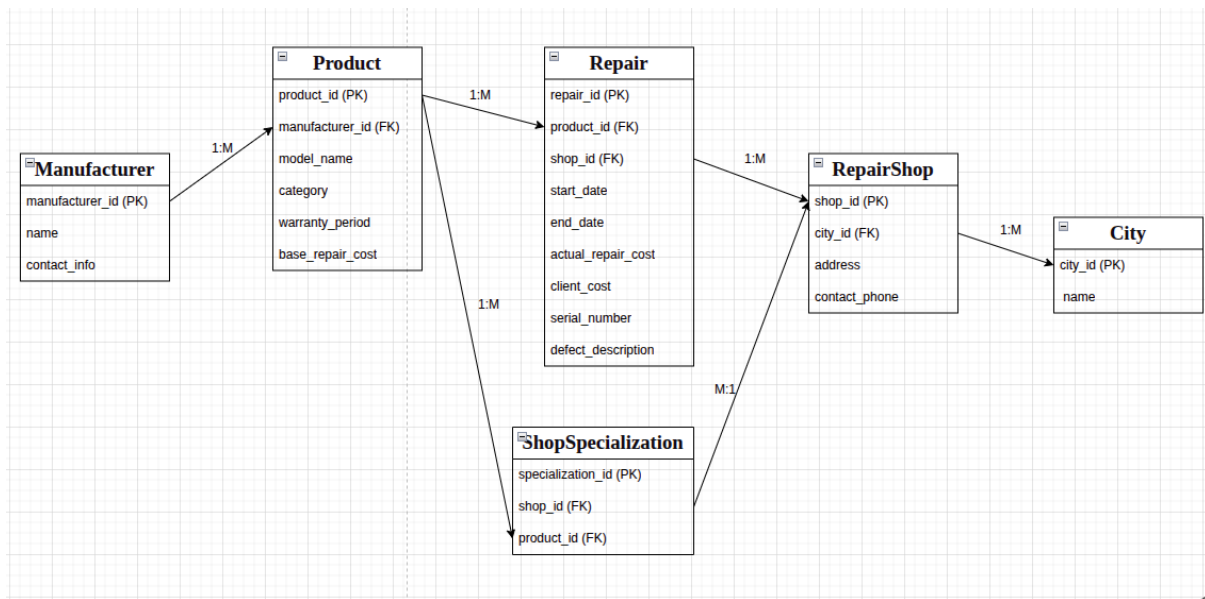
- *repair_id* — Уникальный идентификатор ремонта (Первичный ключ).
- *product_id* — Идентификатор товара, который ремонтировался (Внешний ключ).
- *shop_id* — Идентификатор мастерской, выполнившей ремонт (Внешний ключ).
- *start_date* — Дата начала ремонта.
- *end_date* — Дата завершения ремонта (может быть NULL, если ремонт еще не завершен).
- *actual_repair_cost* — Фактическая стоимость ремонта (затраты мастерской). Если ремонт по гарантии, может быть 0.

- *client_cost* — Стоимость, оплаченная клиентом (если ремонт не по гарантии).
- *serial_number* — Серийный номер отремонтированного устройства.
- *defect_description* — Описание дефекта.

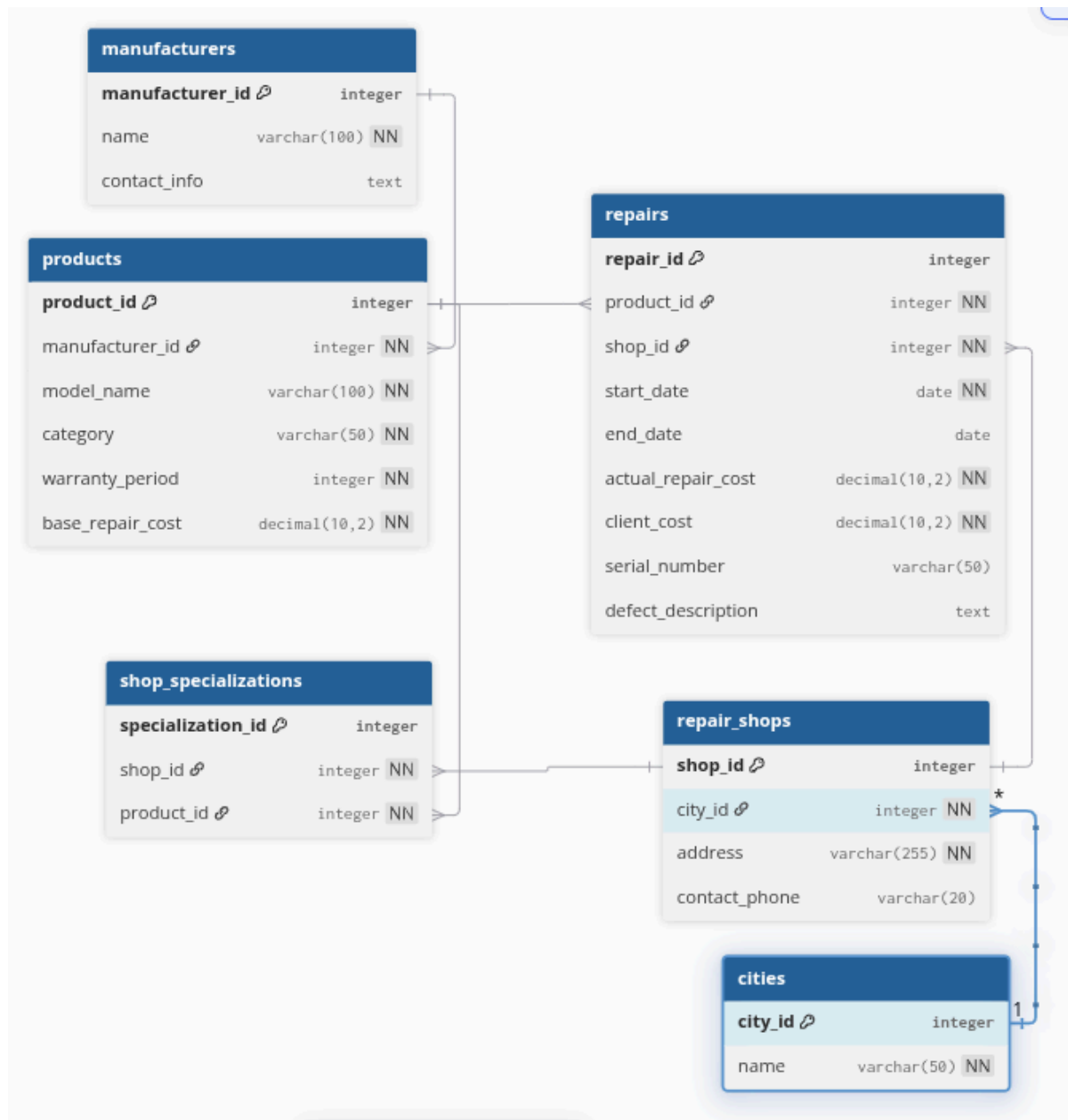
Связи между сущностями:

- **Производитель — Товар (1:M)**: Один производитель выпускает множество товаров. Каждый товар производится одним производителем.
- **Город — Мастерская (1:M)**: В одном городе может быть множество мастерских. Каждая мастерская находится в одном городе.
- **Мастерская — Специализация (1:M)**: Одна мастерская может иметь несколько специализаций (по разным группам товаров). Каждая специализация принадлежит одной мастерской.
- **Товар — Специализация (1:M)**: Один тип товара может быть в списке специализации многих мастерских. Каждая специализация соответствует одному типу товара.
- **Мастерская — Ремонт (1:M)**: Одна мастерская выполняет множество ремонтов. Каждый ремонт выполняется одной мастерской.
- **Товар — Ремонт (1:M)**: Один тип товара может фигурировать во многих ремонтах. Каждый ремонт связан с одним типом товара.

2. Создать диаграмму «сущность — связь» в нотации Чена.



3. Самостоятельно изучить нотацию IDEF1X для представления диаграммы «сущность-связь». Создать схему базы данных в нотации IDEF1X.



```

// Use DBML to define your database structure
// Docs: https://dbml.dbdiagram.io/docs

```

```

Table manufacturers {
  manufacturer_id integer [primary key]
  name varchar(100) [not null]
  contact_info text
}

```

```
Table cities {  
  city_id integer [primary key]  
  name varchar(50) [not null]  
}
```

```
Table products {  
  product_id integer [primary key]  
  manufacturer_id integer [not null]  
  model_name varchar(100) [not null]  
  category varchar(50) [not null]  
  warranty_period integer [not null]  
  base_repair_cost decimal(10,2) [not null]  
}
```

```
Table repair_shops {  
  shop_id integer [primary key]  
  city_id integer [not null]  
  address varchar(255) [not null]  
  contact_phone varchar(20)  
}
```

```
Table shop_specializations {  
  specialization_id integer [primary key]  
  shop_id integer [not null]  
  product_id integer [not null]  
}
```

```
Table repairs {  
  repair_id integer [primary key]
```

```

product_id integer [not null]

shop_id integer [not null]

start_date date [not null]

end_date date

actual_repair_cost decimal(10,2) [not null]

client_cost decimal(10,2) [not null]

serial_number varchar(50)

defect_description text
}

Ref: products.manufacturer_id > manufacturers.manufacturer_id
Ref: repair_shops.city_id > cities.city_id
Ref: shop_specializations.shop_id > repair_shops.shop_id
Ref: shop_specializations.product_id > products.product_id
Ref: repairs.product_id > products.product_id
Ref: repairs.shop_id > repair_shops.shop_id

```

4. Разработать структуру базы данных и составить описание столбцов таблиц базы данных, включающее: имя столбца, назначение (какие данные хранятся), тип данных, допускает ли столбец пустые значения.

Таблица: manufacturers (Производители)

- *manufacturer_id*: INT, не допускает NULL. Уникальный идентификатор производителя.
- *name*: VARCHAR(100), не допускает NULL. Наименование производителя.
- *contact_info*: TEXT, допускает NULL. Контактная информация (адрес, телефон).

Таблица: cities (Города)

- *city_id*: INT, не допускает NULL. Уникальный идентификатор города.
- *name*: VARCHAR(50), не допускает NULL. Название города.

Таблица: products (Товары)

- *product_id*: INT, не допускает NULL. Уникальный идентификатор товара.
- *manufacturer_id*: INT, не допускает NULL. Идентификатор производителя (внешний ключ).
- *model_name*: VARCHAR(100), не допускает NULL. Наименование модели товара.
- *category*: VARCHAR(50), не допускает NULL. Категория/группа товара.
- *warranty_period*: INT, не допускает NULL. Гарантийный период в днях.
- *base_repair_cost*: DECIMAL(10,2), не допускает NULL. Базовая стоимость ремонта.

Таблица: repair_shops (Мастерские)

- *shop_id*: INT, не допускает NULL. Уникальный идентификатор мастерской.
- *city_id*: INT, не допускает NULL. Идентификатор города (внешний ключ).
- *address*: VARCHAR(255), не допускает NULL. Адрес мастерской.
- *contact_phone*: VARCHAR(20), допускает NULL. Контактный телефон мастерской.

Таблица: shop_specializations (Специализации мастерских)

- *specialization_id*: INT, не допускает NULL. Уникальный идентификатор специализации.
- *shop_id*: INT, не допускает NULL. Идентификатор мастерской (внешний ключ).
- *product_id*: INT, не допускает NULL. Идентификатор товара (внешний ключ).

Таблица: repairs (Ремонты)

- *repair_id*: INT, не допускает NULL. Уникальный идентификатор ремонта.
- *product_id*: INT, не допускает NULL. Идентификатор товара (внешний ключ).

- *shop_id*: INT, не допускает NULL. Идентификатор мастерской (внешний ключ).
- *start_date*: DATE, не допускает NULL. Дата начала ремонта.
- *end_date*: DATE, допускает NULL. Дата завершения ремонта.
- *actual_repair_cost*: DECIMAL(10,2), не допускает NULL. Фактическая стоимость ремонта.
- *client_cost*: DECIMAL(10,2), не допускает NULL. Стоимость, оплаченная клиентом.
- *serial_number*: VARCHAR(50), допускает NULL. Серийный номер устройства.
- *defect_description*: TEXT, допускает NULL. Описание дефекта.

Вывод: изучили способы задания инфологической модели данных и создания структуры базы данных в заданной предметной области.