Федеральное государственное бюджетное образовательное учреждение высшего образования «Российская академия народного хозяйства и государственной службы при Президенте Российской Федерации».

## Кафедра «Прикладная Информатика»

ОТЧЕТ

О ПРОДЕЛАННОЙ ЛАБОРАТОРНОЙ РАБОТЕ №9

по курсу «Построение логической модели данных предметной области»

Выполнил: студент группы

ИК-731

**(наименование группы)**

***Соколов Дмитрий Александрович***

**(Ф.И.О.)**

## Нижний Новгород

**2025 г.**

ЛАБОРАТОРНАЯ РАБОТА № 9

«Построение логической модели данных предметной области»

Цель работы:

1. Построить логическую модель данных выбранной предметной области в

Сущность (Entity) - реальный либо воображаемый объект, имеющий существенное значение для рассматриваемой предметной области. Каждая сущность должна иметь наименование, выраженное существительным в единственном числе. Каждая сущность

должна обладать уникальным идентификатором. Каждый экземпляр сущности должен однозначно идентифицироваться и отличаться от всех других экземпляров данного типа сущности.

Атрибут (Attribute) - любая характеристика сущности, значимая для рассматриваемой предметной области и предназначенная для квалификации, идентификации, классификации, количественной характеристики или выражения

состояния сущности. Наименование атрибута должно быть выражено существительным в единственном числе.

Связь (Relationship) - поименованная ассоциация между двумя сущностями, значимая для рассматриваемой предметной области.

В методе IDEF1X все сущности делятся на зависимые и независимые от идентификаторов. Сущность является независимой от идентификаторов или просто независимой, если каждый экземпляр сущности может быть однозначно идентифицирован без определения его отношений с другими сущностями. Сущность называется зависимой от идентификаторов или просто зависимой, если однозначная идентификация экземпляра сущности зависит от его отношения к другой сущности. Независимая сущность изображается в виде обычного прямоугольника, зависимая - в виде прямоугольника с

закругленными углами.

Построение логической модели

1 Выбор сущностей

Для лабораторной работы я выбрал 6 сущностей, которые покрывают основные бизнес-процессы автосервиса:

1. client - Клиент
2. vehicle - Автомобиль
3. job\_order - Заказ-наряд (обращение клиента)
4. job\_service - Строка услуги в заказе (конкретная услуга в заказе)
5. employee - Сотрудник (мастер, администратор, кладовщик и т.п.)
6. part - Запчасть / материал

2 Описание сущностей и атрибутов

**Client (Клиент) - независимая сущность (прямоугольник)**

1. PK: client\_id (integer / surrogate)
2. last\_name (varchar) NOT NULL
3. first\_name (varchar) NOT NULL
4. middle\_name (varchar) NULL
5. is\_company (boolean) NOT NULL DEFAULT FALSE
6. company\_name (varchar) NULL
7. phone (varchar) NOT NULL
8. email (varchar) NULL
9. address (text) NULL
10. notes (text) NULL

**Бизнес-правило:** Email уникален, если используется.

**Vehicle (Автомобиль) - независимая сущность**

1. PK: vehicle\_id
2. FK: client\_id → client.client\_id (NOT NULL)
3. vin (varchar) UNIQUE NULL
4. reg\_number (varchar) NULL
5. make (varchar) NOT NULL
6. model (varchar) NOT NULL
7. year (int) NULL
8. mileage (int) NULL
9. color (varchar) NULL
10. notes (text) NULL

Связь: каждый **vehicle** принадлежит одному **client** (1:N). Связь- неидентифицирующая (в ERwin ставим пунктир). Оставил независимой - Vehicle имеет собственный PK.

**JobOrder (Заказ-наряд)**

1. PK: order\_id
2. FK: client\_id → client.client\_id (NOT NULL)
3. FK: vehicle\_id → vehicle.vehicle\_id (NULL)
4. order\_number (varchar) UNIQUE NOT NULL
5. date\_created (datetime) NOT NULL
6. date\_requested (date) NULL
7. status (varchar) NOT NULL
8. estimated\_amount (decimal) NULL
9. actual\_amount (decimal) NULL
10. notes (text)

Связи и мощности:

client – job\_order (идентифицирующая). (В IDEF1X: client родитель, job\_order потомок, связь сплошная, мощность P с точки зрения job\_order - заказ должен иметь клиента; с точки **client** - может иметь 0..N заказов;

vehicle - неидентифицирующая (job\_order может ссылаться на vehicle, но порядок может существовать и без привязки к конкретному vehicle).

**JobService (Строка услуги в заказе) - зависимая (идентифицирующая) сущность**

PK: составной (order\_id, line\_no)

FK: order\_id → job\_order.order\_id (NOT NULL)

line\_no (int) NOT NULL

service\_code (varchar) NULL

service\_name (varchar) NOT NULL

quantity (int) DEFAULT 1

unit\_price (decimal) NOT NULL

FK: assigned\_employee\_id → employee.employee\_id NULL

status (varchar)

notes (text)

Это строка, описывающая одну конкретную услугу в заказе - реализует отношение «заказ → несколько услуг».

Связи: job\_order – job\_service (идентифицирующая). job\_service идентифицируется через order\_id+line\_no.

**Employee (Сотрудник) - независимая сущность**

1. PK: employee\_id
2. last\_name (varchar) NOT NULL
3. first\_name (varchar) NOT NULL
4. middle\_name (varchar) NOT NULL
5. position (varchar)
6. phone (varchar) NULL
7. email (varchar) NULL
8. hire\_date (date)
9. notes (text)

Связи: employee – job\_service.

**Part (Запчасть / материал) - независимая сущность**

1. PK: part\_id
2. part\_code (varchar) UNIQUE NULL
3. name (varchar) NOT NULL
4. unit (varchar) NOT NULL
5. price (decimal) NULL
6. reorder\_level (int) NULL
7. notes (text)

**3 Связи и их типы**

**client → job\_order**

Тип: идентифицирующая (сплошная) - заказ не существует без клиента.

Мощность: client: N (0-N заказов), job\_order: P (каждый заказ обязательно связан с одним клиентом).

**job\_order → job\_service**

Тип: идентифицирующая (сплошная) - строка услуги существует только в контексте заказа.

Мощность: job\_order: P (заказ содержит минимум одну услугу).

**vehicle → job\_order**

Тип: неидентифицирующая (пунктир) - заказ может ссылаться на автомобиль, но заказ не идентифицируется через автомобиль.

Мощность: vehicle: N (один автомобиль может иметь много заказов), job\_order: Z (каждый заказ может иметь 0 или 1 привязанный автомобиль).

**employee → job\_service**

Тип: неидентифицирующая - сотрудник выполняет/назначен на строку услуги.

Мощность: Employee: N (1 сотрудник выполняет много job\_service), job\_service: Z (необязательно назначен).

На основе данного логического описания модели, была реализована логическая модель базы данных на 6 таблиц.

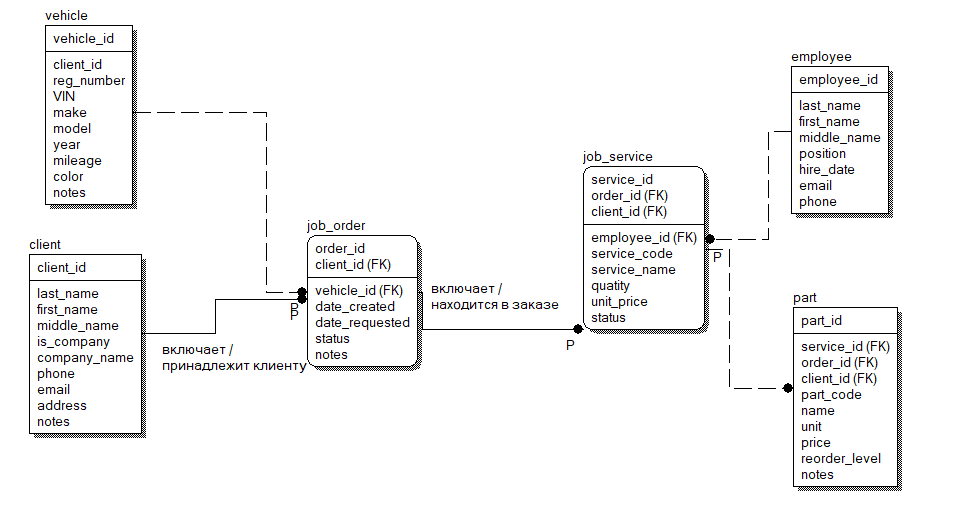


Рисунок 1 – Базовая логическая модель базы данных «Авто-сервис»

Вывод

В ходе выполнения лабораторной работы была построена логическая модель данных предметной области «Автосервис» в нотации IDEF1X с использованием CASE-средства ERwin Data Modeler. В процессе работы были выделены основные сущности, определены их атрибуты и установлены связи между ними с указанием мощностей и типов связей.

Построенная модель отражает структуру данных, необходимую для хранения информации об автомобилях, клиентах, заказах и услугах автосервиса. Полученная логическая схема послужит основой для создания физической модели базы данных и дальнейшей реализации информационной системы автосервиса.