

**Министерство науки и высшего образования Российской Федерации  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**ОТЧЕТ  
ПО ЛАБОРАТОРНОЙ РАБОТЕ №2  
«АНАЛИЗ ДАННЫХ. ПОСТРОЕНИЕ ИНФОЛОГИЧЕСКОЙ  
МОДЕЛИ ДАННЫХ БД»**

По дисциплине «Проектирование и реализация баз данных»

Вариант 11

**Обучающиеся:** Востров Илья Анатольевич

Титор Матвей Андреевич

**Факультет:** ФПИИ

**Группа:** К3239

**Направление подготовки:** 09.03.03 Прикладная информатика

**Образовательная программа:** Мобильные и сетевые технологии

**Преподаватель:** Говорова Марина Михайловна

Санкт-Петербург

2025 г.

## Содержание отчета

Содержание отчета	2
Цель работы	3
Практическое задание	3
Индивидуальное задание	3
Выполнение	5
Название создаваемой БД	5
Состав реквизитов сущностей	5
Схема инфологической модели данных (ER-диаграмма)	6
Схема инфологической модели в нотации IDEF1X	7
Описание атрибутов сущностей и ограничений	8
Алгоритмические связи для вычисляемых данных	13
Выводы	14

## Цель работы

Целью данной лабораторной работы является освоение практических навыков анализа предметной области и построения инфологической модели базы данных (БД) методом «сущность-связь», а также преобразование этой модели в логическую модель в нотации IDEF1X.

## Практическое задание

1. Проанализировать предметную область согласно варианту задания.
2. Выполнить инфологическое моделирование базы данных по заданной предметной области с использованием метода ER-диаграмм («сущность-связь») в комбинированной нотации Питера Чена - Кириллова (задание 1.1 варианта).
3. Реализовать разработанную ИЛМ в нотации IDEF1X.

## Индивидуальное задание

### БД «Автомастерская»

Описание предметной области: Сеть автомастерских осуществляет ремонт автомобилей, используя для этих целей штат мастеров и свои мастерские. Стоимость ремонта включает цену деталей и стоимость работы.

Заработная плата мастеров составляет 50% стоимости работы.

С клиентом заключается договор на выполнение авторемонтных и профилактических работ, который сопровождается администратором. В каждом договоре может быть несколько видов услуг. Для выполнения видов работ могут требоваться детали или расходные материалы, которые предоставляет либо клиент, либо автомастерская. Если детали предоставляет автомастерская, то их стоимость включается в смету по договору.

Каждый вид работ могут выполнять разные мастера, в зависимости от их специализации. Распределение мастеров выполняет администратор.

БД должна содержать следующий минимальный набор сведений: Табельный номер сотрудника. ФИО сотрудника. Должность. Разряд мастера. Специализация. Адрес автомастерской. Дата заказа. Гос. Номер автомобиля. Марка. Мощность автомобиля. Год выпуска. Цвет автомобиля. Дата принятия в ремонт. Плановая дата окончания ремонта. Фактическая дата окончания ремонта. Вид ремонта. Стоимость вида ремонта. Название детали. Цена детали. Марка и модель автомобиля. Страна производителя. Госномер автомобиля. ФИО владельца. Номер телефона владельца. E-mail владельца. Должность сотрудника. Количество ставок (по штатному расписанию).

Дополните состав атрибутов на основе анализа предметной области.

**Задание 1.1.** Выполните инфологическое моделирование базы данных системы. (Ограничения задать самостоятельно.)

**Задание 1.2.** Создайте логическую модель БД, используя ИЛМ (задание 1.1). Используйте необходимые средства поддержки целостности данных в СУБД.

**Задание 2.** Создать запросы:

- Выбрать фамилию того механика, который чаще всех работает с автомобилями марки "Тойота".
- Определить тех владельцев автомобилей, которых всегда обслуживает один и тот же мастер (в зависимости от вида работ). Вывести фамилию механика и его постоянных клиентов.
- Вывести фамилии механиков, которые не выполняли работы в срок и количество дней просрочки выполнения заказа.
- Вывести данные клиентов, которые максимально часто посещали автосервис за прошедший год.
- Сколько заработал каждый мастер за прошедший месяц?
- Вывести данные владельцев автомобилей, которые обращались в ремонт больше одного раза.
- За каждый день просрочки выполнения заказа механику назначается штраф в размере 5%. Рассчитать штраф каждого механика за прошедший месяц.

**Задание 3.** Создать представление:

- для заказчиков (фамилию механика и модель автомобиля, которую он ремонтирует чаще всего);
- для менеджеров (рассчитать премию все механикам, которые за прошедший месяц все свои заказы выполнили своевременно - 10% от зарплаты).

**Задание 4.** Создать хранимые процедуры:

- Повышения цены деталей для автомобиля "Ford" на 10 %.
- Для повышения разряда тех мастеров, которые отремонтировали больше 3 автомобилей.
- Сколько автомобилей отремонтировал каждый механик за истекший квартал.

**Задание 5.** Создать необходимые триггеры.

## Выполнение

Название создаваемой БД

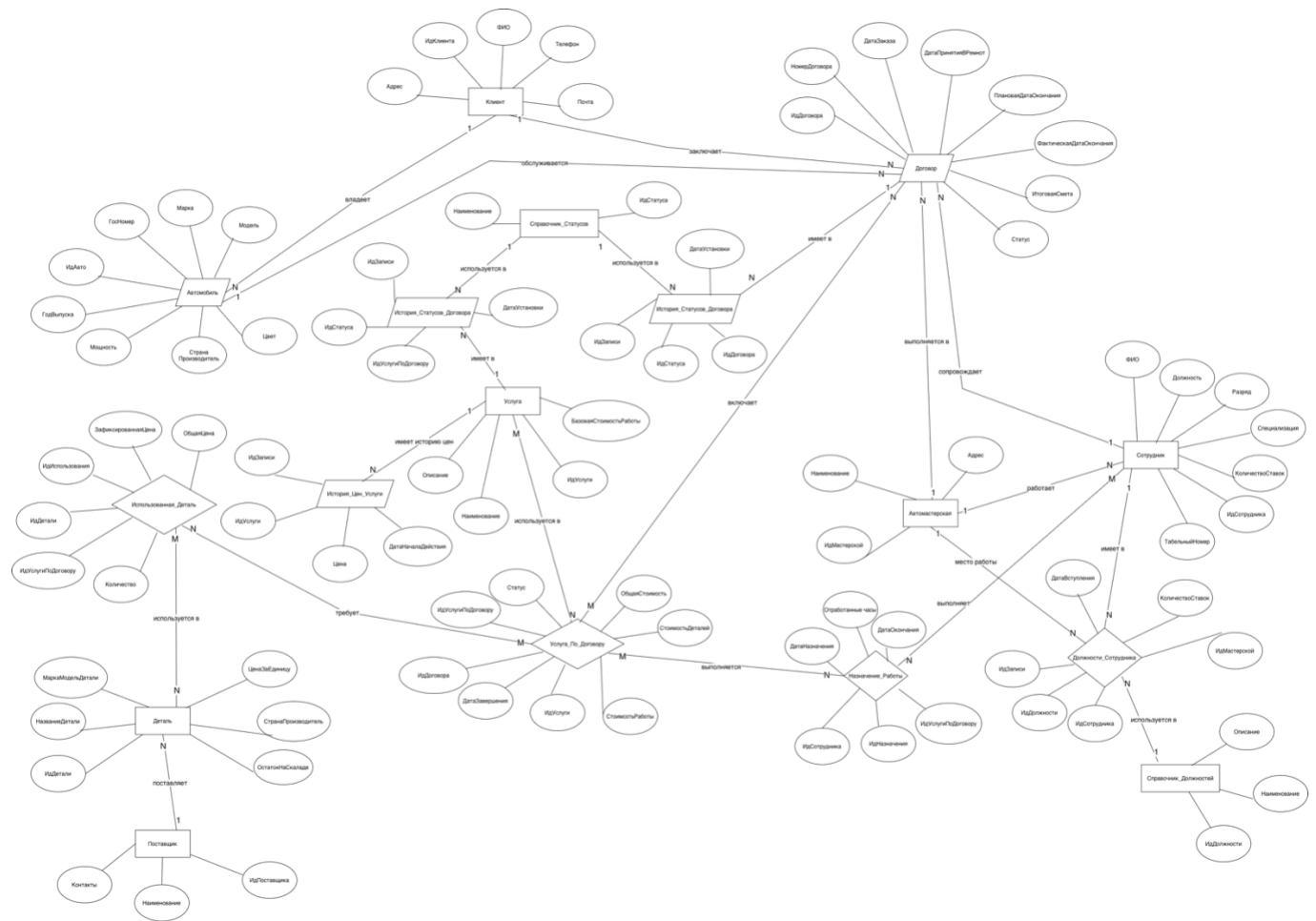
База данных называется **AutoRepairShop**

Состав реквизитов сущностей

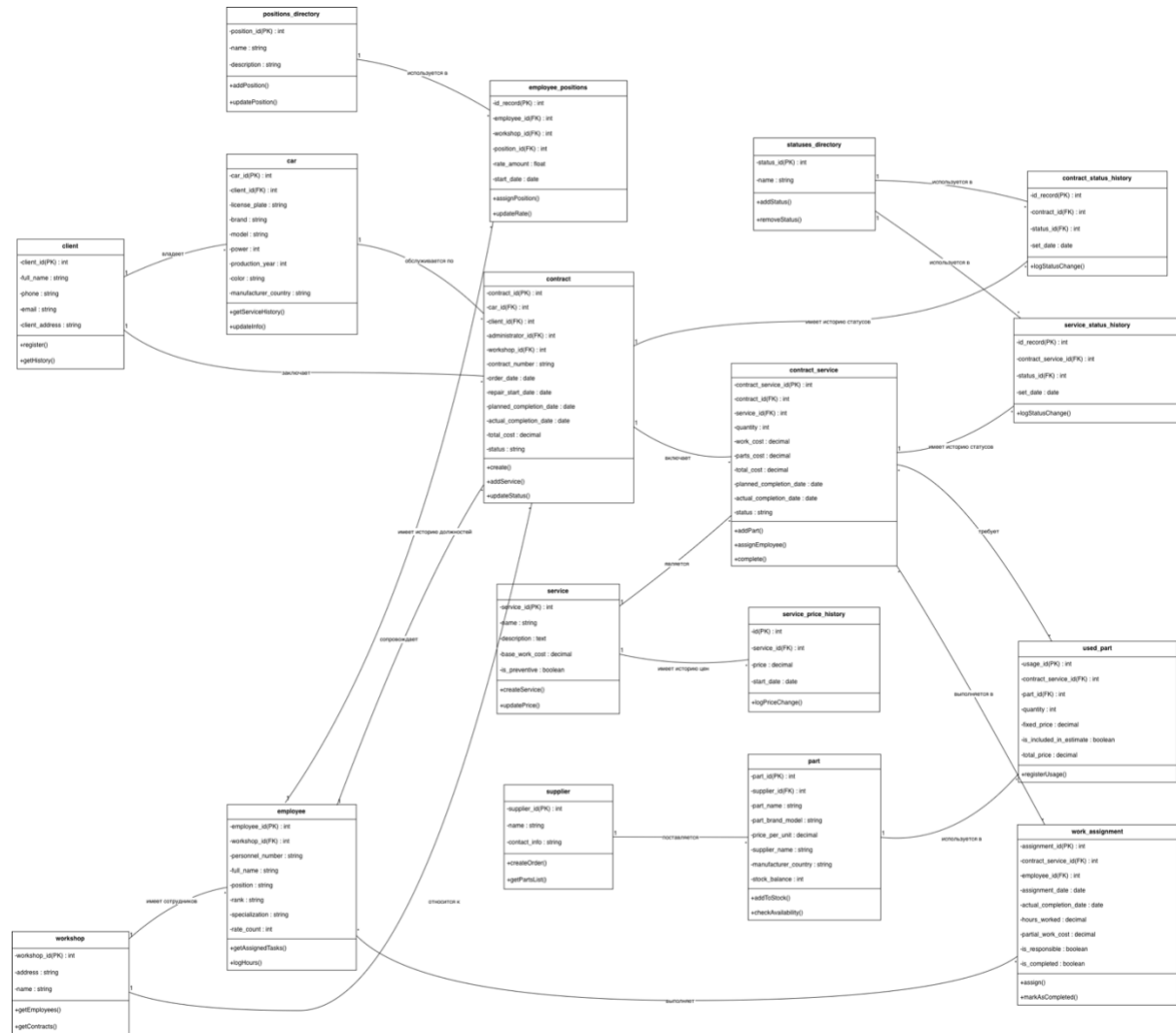
Вот сущности и их реквизиты:

- **Клиент (Client):** client\_id, full\_name, phone, email, client\_address
- **Автомобиль (Car):** car\_id, client\_id, license\_plate, brand, model, power, production\_year, color, manufacturer\_country
- **Мастерская (Workshop):** workshop\_id, address, name
- **Сотрудник (Employee):** employee\_id, workshop\_id, personnel\_number, full\_name, position, rank, specialization, salary\_rate
- **Справочник должностей (Positions\_Directory):** position\_id, name, description
- **Запись должности сотрудника (Employee\_Positions):** id\_record, employee\_id, workshop\_id, position\_id, rate\_amount, start\_date
- **Договор (Contract):** contract\_id, car\_id, client\_id, administrator\_id, workshop\_id, contract\_number, order\_date, repair\_start\_date, planned\_completion\_date, actual\_completion\_date, total\_cost, status
- **Услуга / Сервис (Service):** service\_id, name, description, base\_work\_cost, is\_preventive
- **Услуга в договоре (Contract\_Service):** contract\_service\_id, contract\_id, service\_id, quantity, work\_cost, parts\_cost, total\_cost, planned\_completion\_date, actual\_completion\_date, status
- **Состояние договора — история (Contract\_Status\_History):** id\_record, contract\_id, status\_id, set\_date
- **Справочник статусов (Statuses\_Directory):** status\_id, name
- **Состояние услуги — история (Service\_Status\_History):** id\_record, contract\_service\_id, status\_id, set\_date
- **История цен услуг (Service\_Price\_History):** id\_record, service\_id, price, start\_date
- **Запчасть (Part):** part\_id, supplier\_id, part\_name, part\_brand\_model, price\_per\_unit, supplier\_name, manufacturer\_country, stock\_balance
- **Поставщик (Supplier):** supplier\_id, name, contact\_info
- **Используемая запчасть (Used\_Part):** usage\_id, contract\_service\_id, part\_id, quantity, fixed\_price, is\_included\_in\_estimate, total\_price
- **Назначение работы мастеру (Work\_Assignment):** assignment\_id, contract\_service\_id, employee\_id, assignment\_date, actual\_completion\_date, hours\_worked, partial\_work\_cost, is\_responsible, is\_completed

### Схема инфологической модели данных (ER-диаграмма)



# Схема инфологической модели в нотации IDEF1X



## Описание атрибутов сущностей и ограничений

Таблица 1 – Описание атрибутов сущностей

Наименование атрибута	Тип	Первичный ключ		Внешний ключ	Обязательность	Ограничения целостности
		Собственный атрибут	Внешний ключ			
CLIENT						
client_id	INTEGER	+			+	Уникален, автоинкремент
full_name	VARCHAR(255)				+	Длина ≤ 255
phone	VARCHAR(20)				+	Формат телефонного номера
email	VARCHAR(255)					Формат email
client_address	VARCHAR(255)					—
CAR						
car_id	INTEGER	+			+	Уникален, автоинкремент
client_id	INTEGER		+	+	+	Значение соответствует client.client_id
license_plate	VARCHAR(20)				+	Уникальный гос. номер
brand	VARCHAR(50)				+	—
model	VARCHAR(50)				+	—
power	INTEGER					> 0
production_year	INTEGER					≥ 1900 и ≤ текущий год
color	VARCHAR(30)					—
manufacturer_country	VARCHAR(50)					—
WORKSHOP						
workshop_id	INTEGER	+			+	Уникален, автоинкремент

address	VARCHAR(255)				+	—
name	VARCHAR(100)				+	—
<b>EMPLOYEE</b>						
employee_id	INTEGER	+			+	Уникален, автоинкремент
workshop_id	INTEGER		+	+	+	FK → workshop.workshop_id
personnel_number	VARCHAR(20)				+	Уникален
full_name	VARCHAR(255)				+	—
position	VARCHAR(100)					—
rank	VARCHAR(50)					—
specialization	VARCHAR(100)					—
salary_rate	DECIMAL(10,2)				+	≥ 0
<b>POSITIONS_DIRECTORY</b>						
position_id	INTEGER	+			+	Уникален, автоинкремент
name	VARCHAR(100)				+	—
description	VARCHAR(255)					—
<b>EMPLOYEE_POSITIONS</b>						
id_record	INTEGER	+			+	Уникален, автоинкремент
employee_id	INTEGER		+	+	+	FK employee.employee_id
workshop_id	INTEGER		+	+	+	FK workshop.workshop_id
position_id	INTEGER		+	+	+	FK positions_directory.position_id
rate_amount	DECIMAL(5,2)				+	> 0

start_date	DATE				+	—
<b>CONTRACT</b>						
contract_id	INTEGER	+			+	Уникален, автоинкремент
car_id	INTEGER		+	+	+	FK car.car_id
client_id	INTEGER		+	+	+	FK client.client_id
administrator_id	INTEGER		+	+	+	FK employee.employee_id
workshop_id	INTEGER		+	+	+	FK workshop.workshop_id
contract_number	VARCHAR(30)				+	Уникален
order_date	DATE				+	—
repair_start_date	DATE					≥ order_date
planned_completion_date	DATE					≥ repair_start_date
actual_completion_date	DATE					≥ repair_start_date
total_cost	DECIMAL(10,2)				+	≥ 0
status	VARCHAR(50)				+	—
<b>SERVICE</b>						
service_id	INTEGER	+			+	Уникален, автоинкремент
name	VARCHAR(100)				+	—
description	VARCHAR(255)					—
base_work_cost	DECIMAL(10,2)				+	≥ 0
is_preventive	BOOLEAN				+	—
<b>CONTRACT_SERVICE</b>						
contract_service_id	INTEGER	+			+	Уникален

contract_id	INTEGER		+	+	+	FK contract.contract_id
service_id	INTEGER		+	+	+	FK service.service_id
quantity	INTEGER				+	≥ 1
work_cost	DECIMAL(10,2)				+	≥ 0
parts_cost	DECIMAL(10,2)				+	≥ 0
total_cost	DECIMAL(10,2)				+	= work_cost + parts_cost
planned_completion_date	DATE					—
actual_completion_date	DATE					≥ planned_completion_date
status	VARCHAR(50)				+	—
<b>STATUSES_DIRECTORY</b>						
status_id	INTEGER	+			+	автоинкремент
name	VARCHAR(50)				+	—
<b>CONTRACT_STATUS_HISTORY</b>						
id_record	INTEGER	+			+	автоинкремент
contract_id	INTEGER		+	+	+	FK contract.contract_id
status_id	INTEGER		+	+	+	FK statuses_directory.status_id
set_date	DATE				+	—
<b>SERVICE_STATUS_HISTORY</b>						
id_record	INTEGER	+			+	автоинкремент
contract_service_id	INTEGER		+	+	+	FK contract_service.contract_service_id
status_id	INTEGER		+	+	+	FK statuses_directory.status_id
set_date	DATE				+	—

SERVICE_PRICE_HISTORY						
id_record	INTEGER	+			+	автоинкремент
service_id	INTEGER		+	+	+	FK service.service_id
price	DECIMAL(10,2)				+	$\geq 0$
start_date	DATE				+	—
SUPPLIER						
supplier_id	INTEGER	+			+	автоинкремент
name	VARCHAR(100)				+	—
contact_info	VARCHAR(255)					—
PART						
part_id	INTEGER	+			+	автоинкремент
supplier_id	INTEGER		+	+	+	FK supplier.supplier_id
part_name	VARCHAR(100)				+	—
part_brand_model	VARCHAR(100)					—
price_per_unit	DECIMAL(10,2)				+	$\geq 0$
supplier_name	VARCHAR(100)					—
manufacturer_country	VARCHAR(50)					—
stock_balance	INTEGER				+	$\geq 0$
USED_PART						
usage_id	INTEGER	+			+	автоинкремент
contract_service_id	INTEGER		+	+	+	FK contract_service.contract_service_id
part_id	INTEGER		+	+	+	FK part.part_id
quantity	INTEGER				+	$\geq 1$

fixed_price	DECIMAL(10,2)				+	≥ 0
is_included_in_estimate	BOOLEAN				+	—
total_price	DECIMAL(10,2)				+	= quantity * fixed_price
<b>WORK_ASSIGNMENT</b>						
assignment_id	INTEGER	+			+	автоинкремент
contract_service_id	INTEGER		+	+	+	FK contract_service.contract_service_id
employee_id	INTEGER		+	+	+	FK employee.employee_id
assignment_date	DATE				+	—
actual_completion_date	DATE					≥ assignment_date
hours_worked	DECIMAL(5,2)					≥ 0
partial_work_cost	DECIMAL(10,2)					≥ 0
is_responsible	BOOLEAN				+	—
is_completed	BOOLEAN				+	—

### Алгоритмические связи для вычисляемых данных

#### 1. Заработная плата мастера:

- Для каждой записи Work\_Assignment мастер получает вознаграждение, равное 50% от стоимости выполненной работы (work\_cost).
- Формула: salary = 0.5 \* work\_cost.

#### 2. Общая стоимость договора (Contract.total\_cost):

- Сумма всех типов затрат, связанных с услугами в договоре:  
total\_cost =  $\Sigma$  (Contract\_Service.total\_cost) для всех Contract\_Service, связанных с данным контрактом.

#### 3. Стоимость запчастей в услуге (Contract\_Service.parts\_cost):

- Если Used\_Part.is\_included\_in\_estimate = true, то:  
parts\_cost =  $\Sigma$  (quantity \* fixed\_price) для всех использованных запчастей, включённых в смету.

#### 4. Общая стоимость услуги (Contract\_Service.total\_cost):

- $total\_cost = work\_cost + parts\_cost$ , где  $work\_cost$  и  $parts\_cost$  рассчитываются или хранятся.

## Выводы

- Построенная инфологическая модель охватывает все ключевые аспекты предметной области автомастерской: клиенты, автомобили, мастера, услуги, детали, договоры и мастерские.
- История статусов для услуг и договоров (таблицы `Contract_Status_History` и `Service_Status_History`) обеспечивает трассируемость изменений — это важно для ведения учета и аудита.
- Таблица `Service_Price_History` позволяет хранить динамику изменений цен на услуги, что даёт гибкость для анализа и коррекции ценовой политики.
- Использование вычисляемых атрибутов (например, зарплата мастера, общая стоимость договора) уменьшает дублирование данных и повышает целостность, если расчёты выполняются корректно.
- Преобразование в логическую модель IDEF1X позволяет задать чёткие ограничения (РК, FK, обязательность), что особенно важно для последующей реализации в СУБД.
- В целом модель достаточно хорошо отражает бизнес-логику автомастерской и может быть использована как основа для физической реализации