# Титульный лист

Предмет: Теория баз данных

Практическая работа №3

Название работы: Создание и Нормализация Базы Данных для Бизнес-Процесса Доставки

Группа: БИСО-02-23

ФИО: [Ваше Имя Фамилия]

Дата: [Текущая дата]

# Введение

Цель работы: Целью данной практической работы является создание базы данных для бизнес-процесса доставки крупногабаритного товара и приведение её к 3-ей нормальной форме. Задачи работы включают анализ существующей таблицы, построение ER-диаграммы, проверку и нормализацию таблицы.

# Описание Бизнес-Процесса

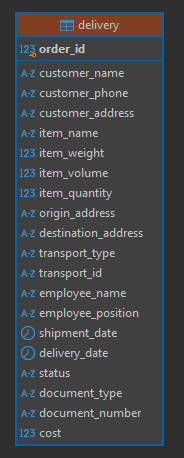
В данной работе рассматривается процесс доставки крупногабаритного товара. Основные этапы бизнес-процесса включают:  
- Получение заказа от клиента.  
- Обработка заказа транспортной компанией.  
- Подготовка товара к отправке (упаковка, оформление документов).  
- Транспортировка.  
- Получение товара клиентом.

Информационные объекты и атрибуты:  
- Заказ (номер заказа, дата заказа, статус).  
- Товар (наименование, вес, объем, количество).  
- Клиент (ФИО, контактные данные, адрес доставки).  
- Транспортное средство (тип, регистрационный номер).  
- Сотрудник (ФИО, должность).  
- Документы (накладная, счет, договор).

# Практическая Часть

## Задание 1: Построение ER-Диаграммы

Описание:  
Использовал конструктор ER-диаграмм Visual Paradigm Online для создания начальной ER-диаграммы таблицы `delivery`.

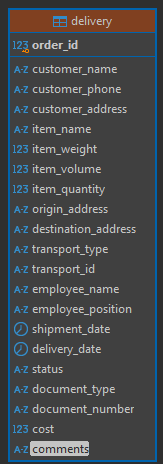
Скриншот ER-Диаграммы:  


## Задание 2: Проверка 1-ой Нормальной Формы

Описание:  
Таблица `delivery` соответствует 1NF, так как все значения атрибутов атомарные и не содержат множественных значений.

## Задание 3: Добавление Атрибута `comments`

Описание:  
Добавлен атрибут `comments` для хранения информации о сотруднике, вносящем запись.  
SQL Команда: ALTER TABLE delivery ADD COLUMN comments VARCHAR(100);

Скриншот Измененной Таблицы:  


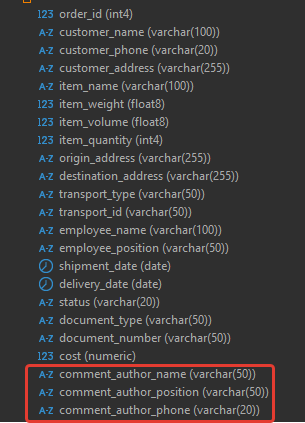
Анализ:  
Добавление атрибута `comments` нарушает атомарность, так как один атрибут содержит несколько данных.

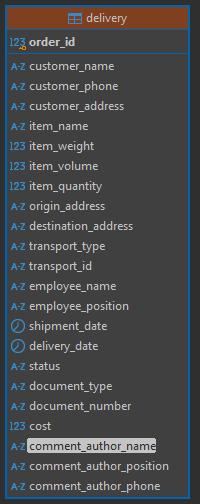
## Задание 4: Приведение Таблицы к 1NF

Описание:  
Разделил атрибут `comments` на три отдельных атрибута: `comment\_author\_name`, `comment\_author\_position`, `comment\_author\_phone`.

SQL Команды:  
ALTER TABLE delivery ADD COLUMN comment\_author\_name VARCHAR(50);  
ALTER TABLE delivery ADD COLUMN comment\_author\_position VARCHAR(50);  
ALTER TABLE delivery ADD COLUMN comment\_author\_phone VARCHAR(20);  
ALTER TABLE delivery DROP COLUMN comments;

Скриншот Обновленной Таблицы:



ER-Диаграмма после Изменений:  


## Задание 5: Проверка и Приведение к 2NF

Описание:  
Таблица `delivery` находится в 2NF, так как все неключевые атрибуты полностью зависят от первичного ключа `order\_id`.

## Задание 6: Проверка и Приведение к 3NF

Описание:  
В таблице `delivery` обнаружены транзитивные зависимости, такие как зависимость `customer\_phone` от `customer\_name` и `employee\_position` от `employee\_name`. Для устранения этих зависимостей были созданы отдельные таблицы `Customer` и `Employee`.

CREATE TABLE Customer (

customer\_id SERIAL PRIMARY KEY,

customer\_name VARCHAR(100),

customer\_phone VARCHAR(20),

customer\_address VARCHAR(255)

);

CREATE TABLE Employee (

employee\_id SERIAL PRIMARY KEY,

employee\_name VARCHAR(100),

employee\_position VARCHAR(50)

);

CREATE TABLE Delivery (

order\_id SERIAL PRIMARY KEY,

customer\_id INT REFERENCES Customer(customer\_id),

item\_name VARCHAR(100),

item\_weight FLOAT,

item\_volume FLOAT,

item\_quantity INT,

origin\_address VARCHAR(255),

destination\_address VARCHAR(255),

transport\_type VARCHAR(50),

transport\_id VARCHAR(50),

employee\_id INT REFERENCES Employee(employee\_id),

shipment\_date DATE,

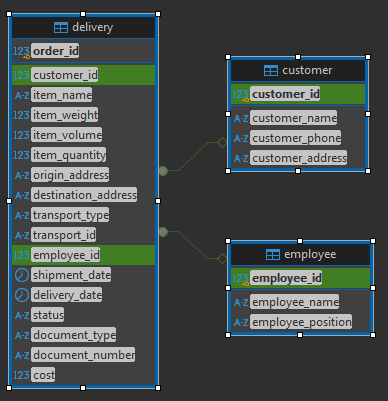
delivery\_date DATE,

status VARCHAR(20),

document\_type VARCHAR(50),

document\_number VARCHAR(50),

cost DECIMAL

);  
  
Скриншот Обновленной ER-Диаграммы:  


Обоснование Разделения:  
Разделение таблицы на `Customer`, `Employee` и `Delivery` устраняет транзитивные зависимости, обеспечивая целостность и избегая избыточности данных.

# Заключение

В ходе выполнения практической работы была создана база данных для бизнес-процесса доставки крупногабаритного товара. Таблица `delivery` была проанализирована и приведена к 3-ей нормальной форме, что обеспечило целостность данных и уменьшило избыточность. Были созданы дополнительные таблицы `Customer` и `Employee` для устранения транзитивных зависимостей.