|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  | | --- | | МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ | | ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ | | **«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»** | | **(МОСКОВСКИЙ ПОЛИТЕХ)** | | Факультет информационных технологий | |
|  |

Кафедра «Прикладная информатика»

Форма обучения: очная

|  |  |  |
| --- | --- | --- |
| Лабораторная работа №2  по дисциплине  «Хранилища данных» | | |
| Тема: *«Реализация структуры Data Vault»* | | |
|  | | |
| Группа | *211-365* |  |
| Студент | *\_\_\_\_\_\_\_\_\_\_\_\_\_* | Д.М. Рахманова |
|  |  |  |
|  |  |  |
|  |  |  |

МОСКВА 2024

# Реализуйте структуру базы данных в соответствии с моделью 3NF

Вариант 9: Система должна описывать работу одного из участков автосалона.

Для создания структуры базы данных автосалона в третьей нормальной форме (3NF) в MySQL Workbench, необходимо следовать принципам нормализации, чтобы избежать избыточности данных и обеспечить целостность.

Даталогическая модель (Рис. 1).

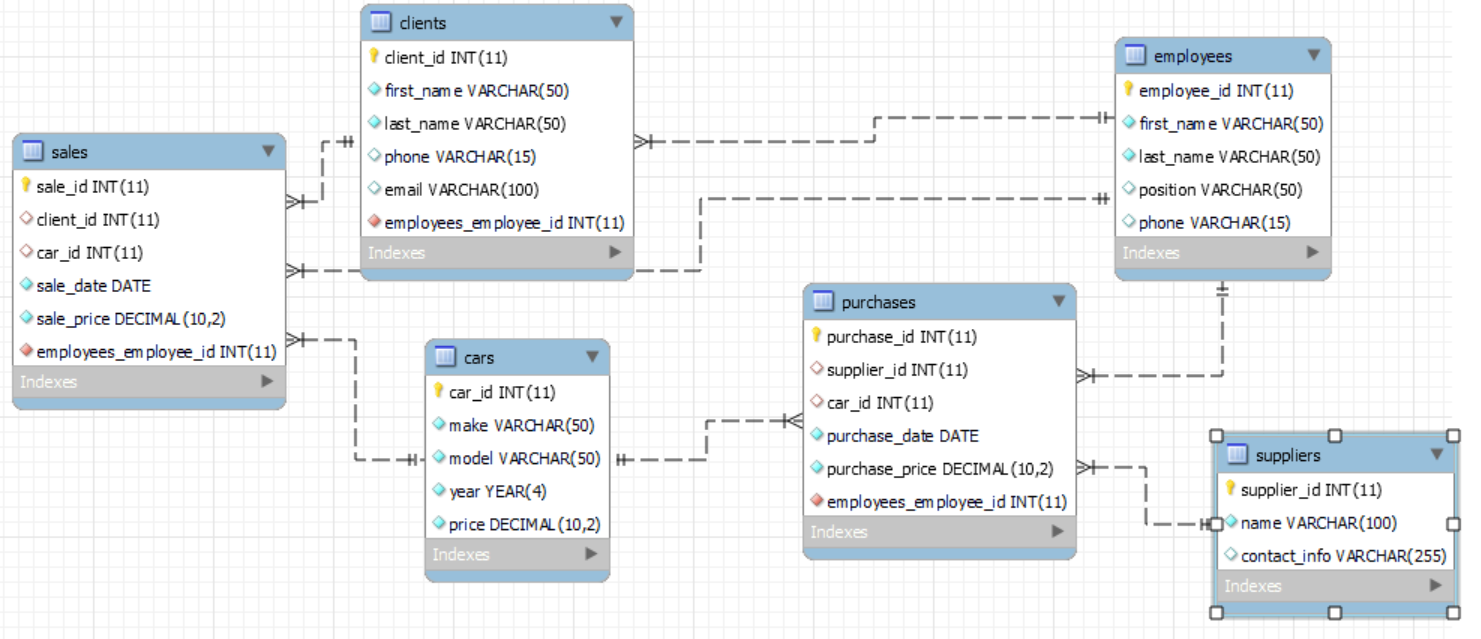


Рис. 1 – Даталогическая модель

В данной модели следующие сущности:

Клиенты:

* client\_id (PK)
* first\_name
* last\_name
* phone
* email

Автомобили:

* car\_id (PK)
* make
* model
* year
* price

Продажи:

* sale\_id (PK)
* client\_id (FK)
* car\_id (FK)
* sale\_date
* sale\_price

Сотрудники:

* employee\_id (PK)
* first\_name
* last\_name
* position
* phone

Поставщики:

* supplier\_id (PK)
* name
* contact\_info

Закупки:

* purchase\_id (PK)
* supplier\_id (FK)
* car\_id (FK)
* purchase\_date
* purchase\_price

Связи между сущностями:

1. Один клиент может совершить много продаж, но каждая продажа принадлежит только одному клиенту.
2. Один автомобиль может быть продан много раз, но каждая продажа относится к одному автомобилю.
3. Один сотрудник может работать с многими продажами, но каждая продажа фиксируется за одним сотрудником.
4. Один поставщик может поставлять много автомобилей, но каждый автомобиль может быть поставлен только одним поставщиком.

Процесс создания БД через SQL-запрос (Рис. 2).

SQL-код для создания базы данных:

CREATE DATABASE CarDealership;

USE CarDealership;

CREATE TABLE Clients (

client\_id INT AUTO\_INCREMENT PRIMARY KEY,

first\_name VARCHAR(50) NOT NULL,

last\_name VARCHAR(50) NOT NULL,

phone VARCHAR(15),

email VARCHAR(100) UNIQUE

);

CREATE TABLE Cars (

car\_id INT AUTO\_INCREMENT PRIMARY KEY,

make VARCHAR(50) NOT NULL,

model VARCHAR(50) NOT NULL,

year YEAR NOT NULL,

price DECIMAL(10, 2) NOT NULL

);

CREATE TABLE Sales (

sale\_id INT AUTO\_INCREMENT PRIMARY KEY,

client\_id INT,

car\_id INT,

employee\_id INT,

sale\_date DATE NOT NULL,

sale\_price DECIMAL(10, 2) NOT NULL,

FOREIGN KEY (client\_id) REFERENCES Clients(client\_id),

FOREIGN KEY (car\_id) REFERENCES Cars(car\_id),

FOREIGN KEY (employee\_id) REFERENCES Employees(employee\_id)

);

CREATE TABLE Employees (

employee\_id INT AUTO\_INCREMENT PRIMARY KEY,

first\_name VARCHAR(50) NOT NULL,

last\_name VARCHAR(50) NOT NULL,

position VARCHAR(50),

phone VARCHAR(15)

);

CREATE TABLE Suppliers (

supplier\_id INT AUTO\_INCREMENT PRIMARY KEY,

name VARCHAR(100) NOT NULL,

contact\_info VARCHAR(255)

);

CREATE TABLE Purchases (

purchase\_id INT AUTO\_INCREMENT PRIMARY KEY,

supplier\_id INT,

car\_id INT,

employee\_id INT,

purchase\_date DATE NOT NULL,

purchase\_price DECIMAL(10, 2) NOT NULL,

FOREIGN KEY (supplier\_id) REFERENCES Suppliers(supplier\_id),

FOREIGN KEY (car\_id) REFERENCES Cars(car\_id),

FOREIGN KEY (employee\_id) REFERENCES Employees(employee\_id)

);

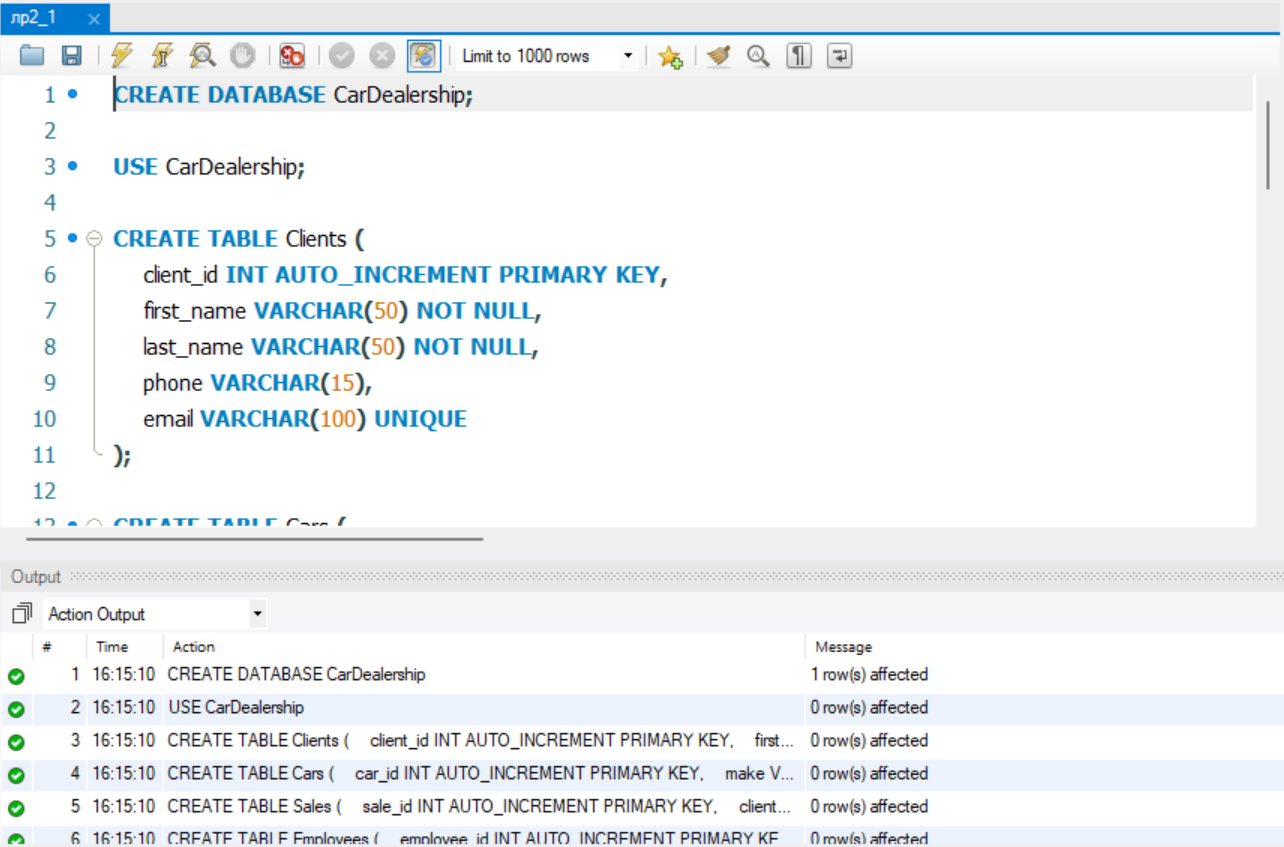


Рис. 2 – Создание БД через SQL-запрос

# Реализуйте структуру базы данных в соответствии с моделью Data Vault

Для создания структуры базы данных автосалона в соответствии с моделью Data Vault, будет использовано три основных компонента: Hubs, Links и Satellites. Эти компоненты помогут организовать данные таким образом, чтобы обеспечить гибкость и возможность интеграции из различных источников.

Даталогическая модель (Рис. 3).

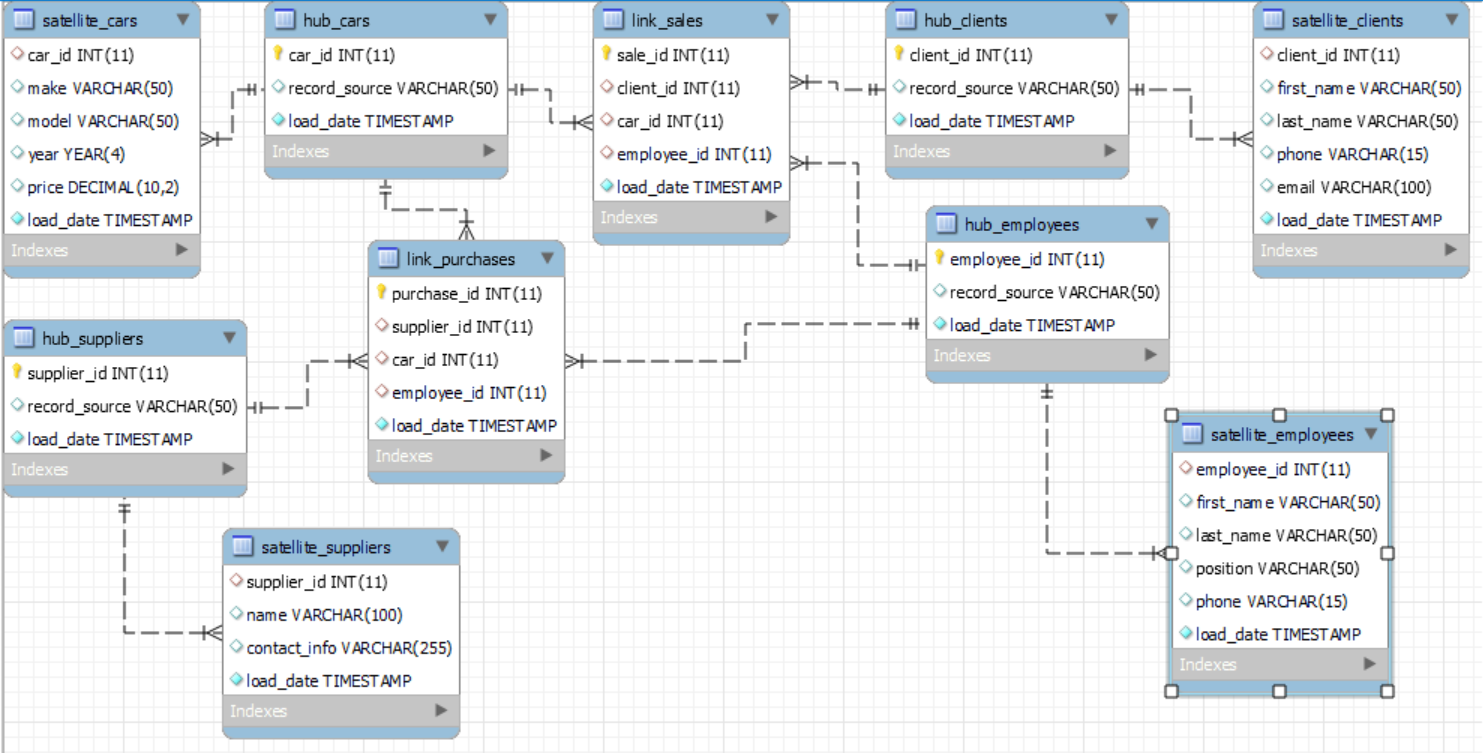


Рис. 3 – Даталогическая модель

Hubs – основные бизнес-ключи, которые идентифицируют уникальные сущности.

Hub\_Clients:

* client\_id (PK)
* record\_source
* load\_date

Hub\_Cars

* car\_id (PK)
* record\_source
* load\_date

Hub\_Employees

* employee\_id (PK)
* record\_source
* load\_date

Hub\_Suppliers

* supplier\_id (PK)
* record\_source
* load\_date

Links – связи между бизнес-ключами.

Link\_Sales

* sale\_id (PK)
* client\_id (FK к Hub\_Clients)
* car\_id (FK к Hub\_Cars)
* employee\_id (FK к Hub\_Employees)
* load\_date

Link\_Purchases

* purchase\_id (PK)
* supplier\_id (FK к Hub\_Suppliers)
* car\_id (FK к Hub\_Cars)
* employee\_id (FK к Hub\_Employees)
* load\_date

Satellites – дополнительные атрибуты, которые описывают сущности.

Satellite\_Clients

* client\_id (FK к Hub\_Clients)
* first\_name
* last\_name
* phone
* email
* load\_date

Satellite\_Cars

* car\_id (FK к Hub\_Cars)
* make
* model
* year
* price
* load\_date

Satellite\_Employees

* employee\_id (FK к Hub\_Employees)
* first\_name
* last\_name
* position
* phone
* load\_date

Satellite\_Suppliers

* supplier\_id (FK к Hub\_Suppliers)
* name
* contact\_info
* load\_date

Процесс создания БД через SQL-запрос (Рис. 3).

SQL-код для создания базы данных:

CREATE DATABASE CarDealership2;

USE CarDealership2;

-- Hubs

CREATE TABLE Hub\_Clients (

client\_id INT AUTO\_INCREMENT PRIMARY KEY,

record\_source VARCHAR(50),

load\_date DATETIME DEFAULT CURRENT\_TIMESTAMP

);

CREATE TABLE Hub\_Cars (

car\_id INT AUTO\_INCREMENT PRIMARY KEY,

record\_source VARCHAR(50),

load\_date DATETIME DEFAULT CURRENT\_TIMESTAMP

);

CREATE TABLE Hub\_Employees (

employee\_id INT AUTO\_INCREMENT PRIMARY KEY,

record\_source VARCHAR(50),

load\_date DATETIME DEFAULT CURRENT\_TIMESTAMP

);

CREATE TABLE Hub\_Suppliers (

supplier\_id INT AUTO\_INCREMENT PRIMARY KEY,

record\_source VARCHAR(50),

load\_date DATETIME DEFAULT CURRENT\_TIMESTAMP

);

-- Links

CREATE TABLE Link\_Sales (

sale\_id INT AUTO\_INCREMENT PRIMARY KEY,

client\_id INT,

car\_id INT,

employee\_id INT,

load\_date DATETIME DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (client\_id) REFERENCES Hub\_Clients(client\_id),

FOREIGN KEY (car\_id) REFERENCES Hub\_Cars(car\_id),

FOREIGN KEY (employee\_id) REFERENCES Hub\_Employees(employee\_id)

);

CREATE TABLE Link\_Purchases (

purchase\_id INT AUTO\_INCREMENT PRIMARY KEY,

supplier\_id INT,

car\_id INT,

employee\_id INT,

load\_date DATETIME DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (supplier\_id) REFERENCES Hub\_Suppliers(supplier\_id),

FOREIGN KEY (car\_id) REFERENCES Hub\_Cars(car\_id),

FOREIGN KEY (employee\_id) REFERENCES Hub\_Employees(employee\_id)

);

-- Satellites

CREATE TABLE Satellite\_Clients (

client\_id INT,

first\_name VARCHAR(50),

last\_name VARCHAR(50),

phone VARCHAR(15),

email VARCHAR(100),

load\_date DATETIME DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (client\_id) REFERENCES Hub\_Clients(client\_id)

);

CREATE TABLE Satellite\_Cars (

car\_id INT,

make VARCHAR(50),

model VARCHAR(50),

year YEAR,

price DECIMAL(10, 2),

load\_date DATETIME DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (car\_id) REFERENCES Hub\_Cars(car\_id)

);

CREATE TABLE Satellite\_Employees (

employee\_id INT,

first\_name VARCHAR(50),

last\_name VARCHAR(50),

position VARCHAR(50),

phone VARCHAR(15),

load\_date DATETIME DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (employee\_id) REFERENCES Hub\_Employees(employee\_id)

);

CREATE TABLE Satellite\_Suppliers (

supplier\_id INT,

name VARCHAR(100),

contact\_info VARCHAR(255),

load\_date DATETIME DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (supplier\_id) REFERENCES Hub\_Suppliers(supplier\_id)

);

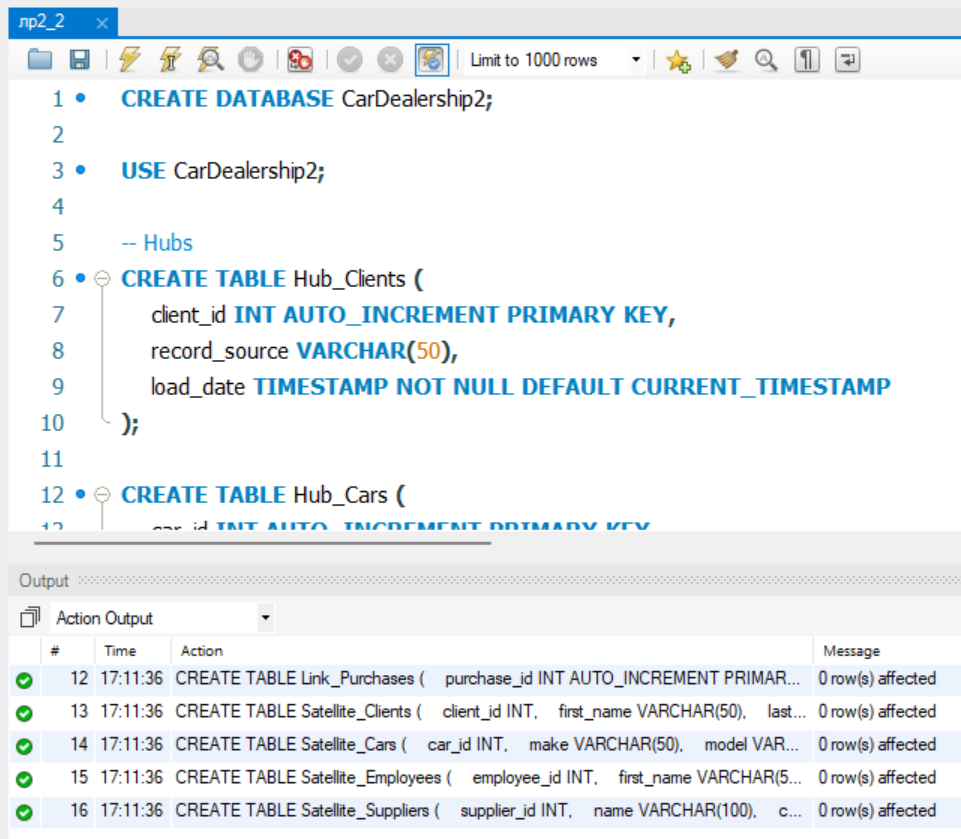


Рис. 4 – Создание БД через SQL-запрос