Департамент образования города Москвы

Государственное автономное образовательное учреждение

высшего образования города Москвы

«Московский городской педагогический университет»

Институт цифрового образования

Департамент информатики, управления и технологий

ОТЧЕТ

по дисциплине «Интеграция и развертывание программного обеспечения с помощью контейнеров»

Направление подготовки 38.03.05 – бизнес-информатика

Профиль подготовки «Аналитика данных и эффективное управление»

(очная форма обучения)

Лабораторная работа 2.1 Динамические соединения с базами данных

Выполнил:

st-89

Руководитель:

Москва  
2025

**Цель работы:** получить практические навыки создания ETL-процесса для интеграции данных из различных источников с использованием динамических соединений в Pentaho Data Integration, включая обработку повторяющихся данных.

**Задача:** Вариант 2 - Фильтр по региону: только Central

**Ход работы:**

1. Для начала создадим таблицы в phpmyadmin и далее в ETL процесс добавим узел Filter rows для того, чтобы отфильтровать по региону Central и оставить только строки с этим значением. (*Рис. 1* и *Рис. 2*)

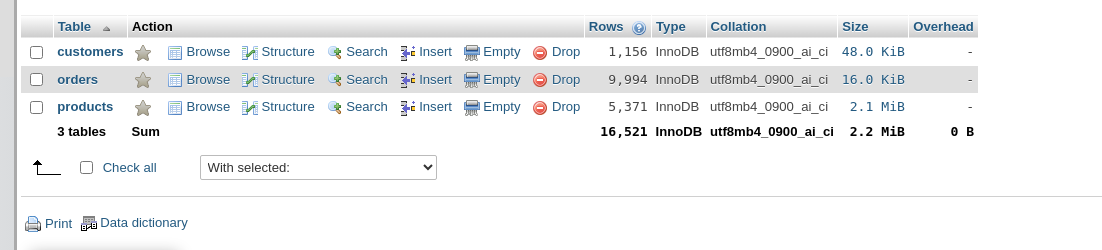


Рис. 1 – Создание таблица в phpmyadmin

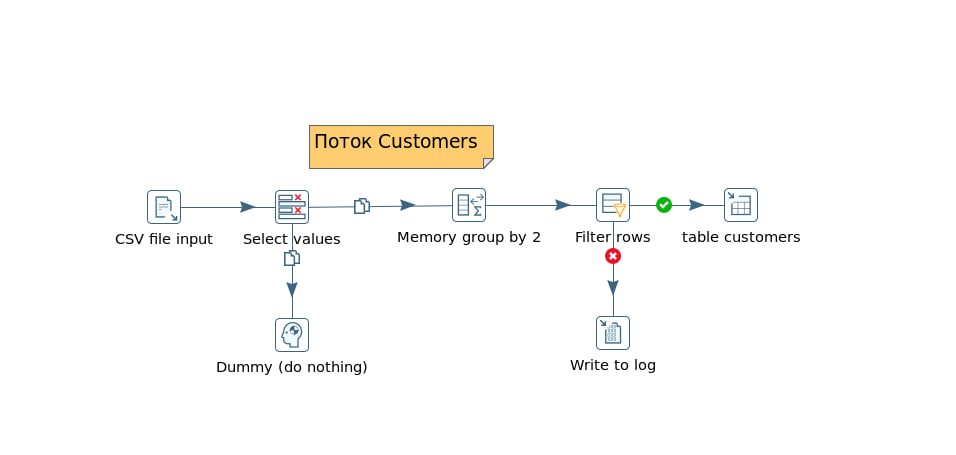


Рис. 2 – Добавление фильтра для загрузки данных в таблицу Customer только по региону Central

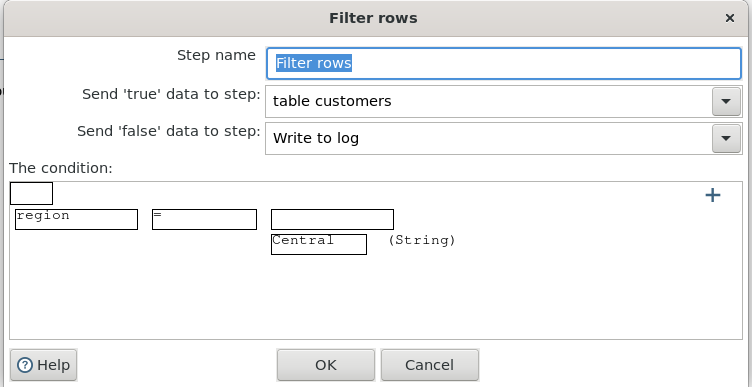


Рис. 3 – Настройки узла Filter rows

1. После чего запустим все ETL процессы и проверим загрузку данных в phpmyadmin.

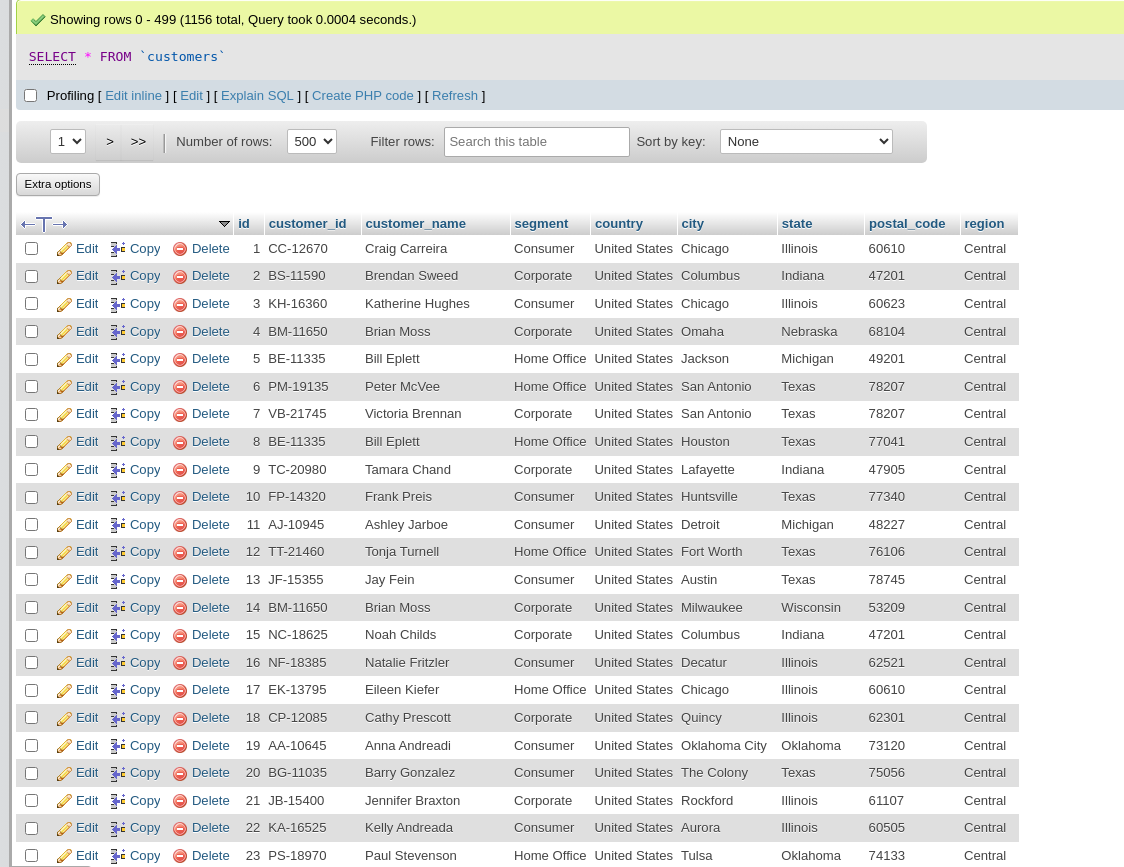


Рис. 4 – Загрузка данных в customers

На Рис. 4 видим, что загрузка по региону Central прошла успешно.

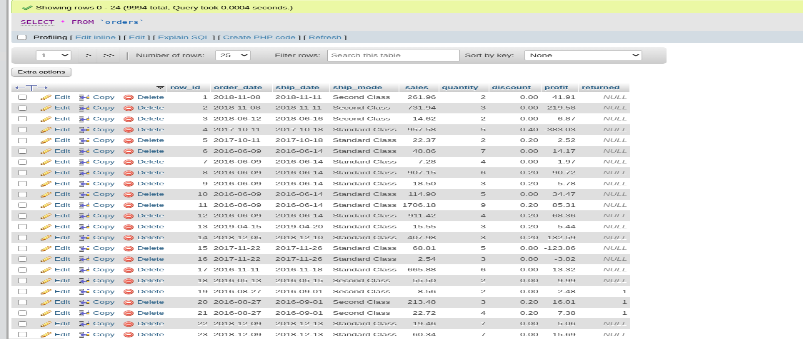


Рис. 5 – Загрузка данных в таблицу orders

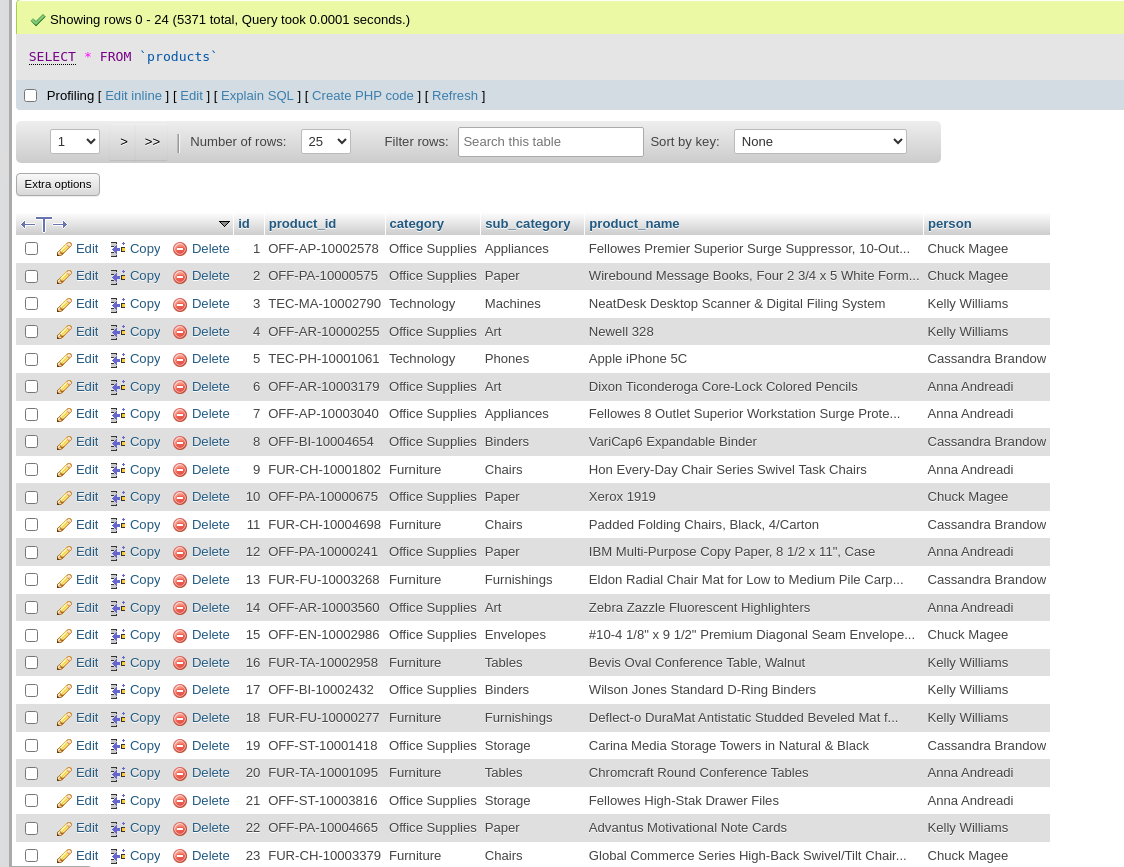


Рис. 6 – Загрузка данных в таблицу products

**Вывод:** Была проведена загрузка данных, а также фильтрация по региону Central по средством ETL процесса в Pentaho

**ПРИЛОЖЕНИЕ**

SQL код для создания таблиц

-- Таблица заказов (основная информация о продажах)

CREATE TABLE orders (

row\_id INT PRIMARY KEY,

order\_date DATE,

ship\_date DATE,

ship\_mode VARCHAR(50),

sales DECIMAL(10,2),

quantity INT,

discount DECIMAL(4,2),

profit DECIMAL(10,2),

returned TINYINT(1) DEFAULT 0 -- 1 = Yes, 0 = No

);

-- Таблица клиентов

-- Пересоздаем таблицу customers

DROP TABLE IF EXISTS customers;

CREATE TABLE customers (

id INT AUTO\_INCREMENT PRIMARY KEY,

customer\_id VARCHAR(20) NOT NULL,

customer\_name VARCHAR(100),

segment VARCHAR(50),

country VARCHAR(100),

city VARCHAR(100),

state VARCHAR(100),

postal\_code VARCHAR(20),

region VARCHAR(50),

INDEX idx\_customer\_id (customer\_id),

INDEX idx\_region (region)

);

-- Пересоздаем таблицу products

DROP TABLE IF EXISTS products;

CREATE TABLE products (

id INT AUTO\_INCREMENT PRIMARY KEY,

product\_id VARCHAR(20) NOT NULL,

category VARCHAR(50),

sub\_category VARCHAR(50),

product\_name VARCHAR(255),

person VARCHAR(100),

INDEX idx\_product\_id (product\_id), -- Обычный индекс вместо UNIQUE

INDEX idx\_category (category),

INDEX idx\_subcategory (sub\_category)

);