Департамент образования и науки города Москвы

Государственное автономное образовательное учреждение высшего образования города Москвы

«Московский городской педагогический университет»

Институт цифрового образования

Департамент информатики, управления и технологий

ДИСЦИПЛИНА:

**Проектный практикум по разработке ETL-решений**

**Самостоятельная работа 1**

**Интеграция данных из разных источников (баз данных)**

Выполнила: Шведова С.С., группа: АДЭУ-211

Преподаватель: Босенко Т.М.

Москва

2025

**Цель:** разработка ETL-процесса для интеграции данных между PostgreSQL и MySQL с использованием Pentaho Data Integration.

**Задачи:**

− Создать исходные таблицы в PostgreSQL с различными наборами данных.

− Настроить целевые таблицы в MySQL для приема данных.

− Разработать процессы трансформации данных в Pentaho.

− Реализовать механизмы обработки ошибок и валидации данных.

− Создать представления для связанных данных.

**Вариант St\_101.** Создать таблицу equipment (id, type, purchase\_date, cost, location), Создать таблицу equipment\_tracking с полем depreciation\_value, Фильтр по типу оборудования, Расчет амортизации

Для начала надо остановить монго (рисунок 1).



Рисунок 1. Остановка монго

Далее надо остановить и снова запустить Postres (рисунок 2).

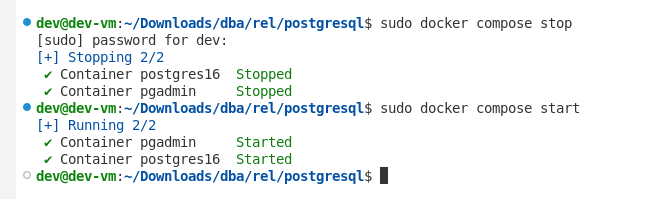


Рисунок 2. Остановка и запуск Postres

Теперь нужно запустить Pentaho (рисунок 3).



Рисунок 3. Запуск Pentaho

После этого нужно создать базу данных (рисунок 4).

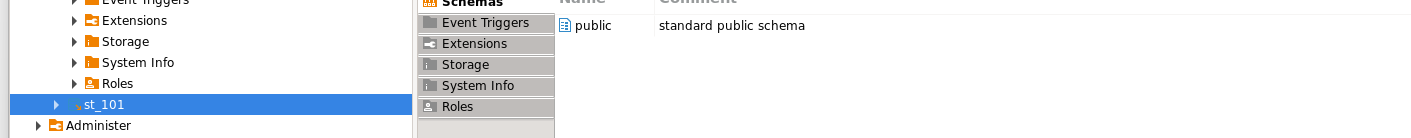


Рисунок 4. Создание базы данных

После этого нужно создать две таблицы и заполнить их данными (рисунок 5-6).

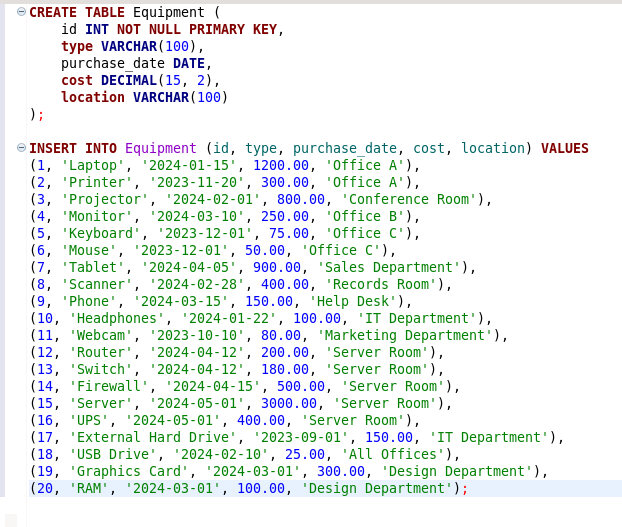


Рисунок 5. Создание таблицы Equipment

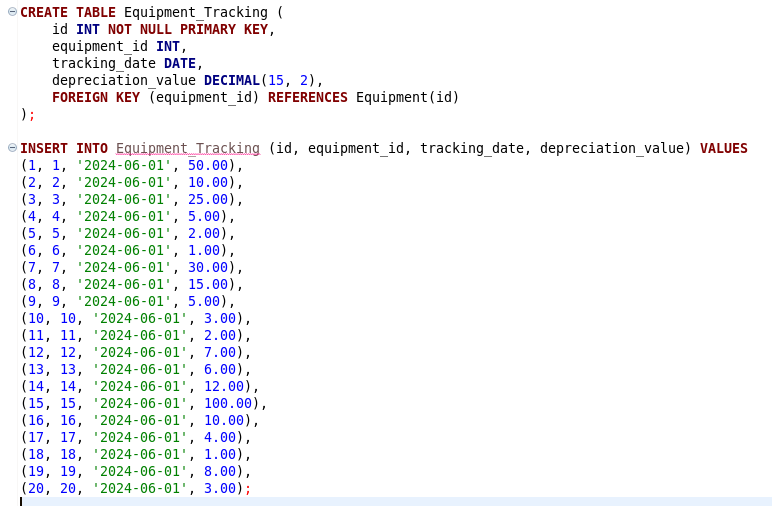


Рисунок 6. Создание таблицы Equipment

Схема базы данных продемонстрирована на рисунке 7.

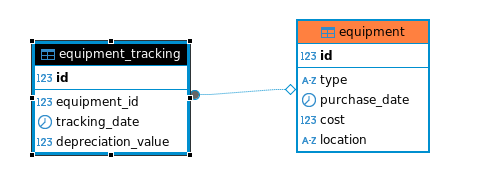


Рисунок 7. Схема базы данных

Далее нужно создать эти таблицы в PhpMyAdmin (рисунок 8).

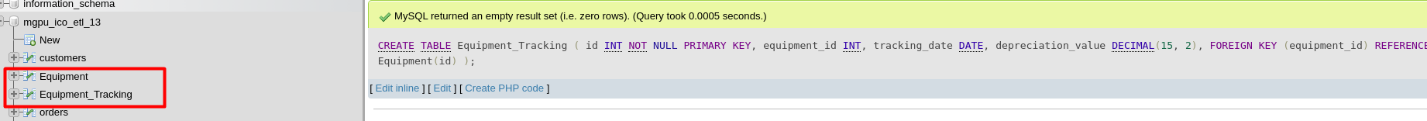


Рисунок 8. Создание таблиц в PhpMyAdmin

Теперь надо подключиться к PostreSQL (рисунок 9).

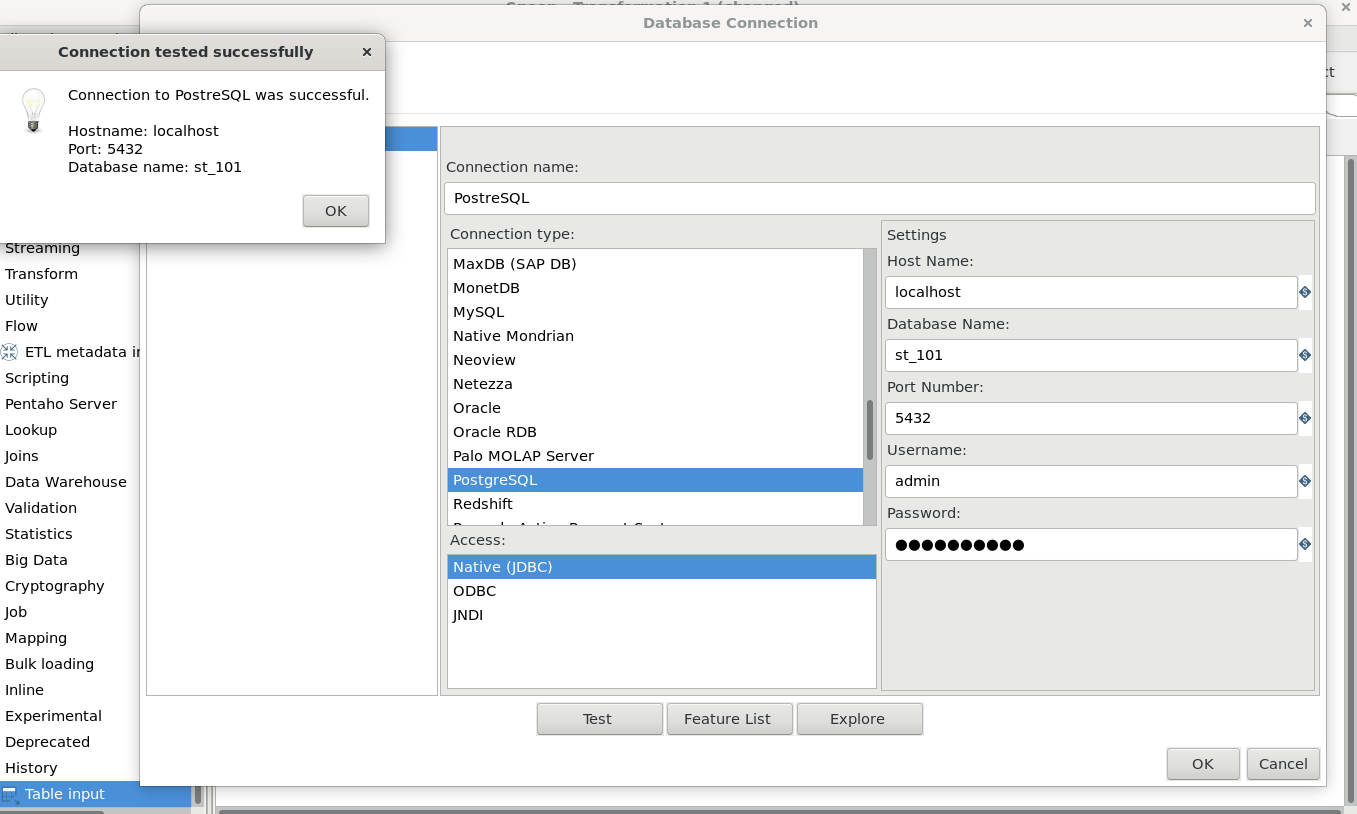


Рисунок 9. Подключение к базе данных PostreSQL

Потом нужно написать запрос, чтобы эта таблица выводилась (рисунок 10).

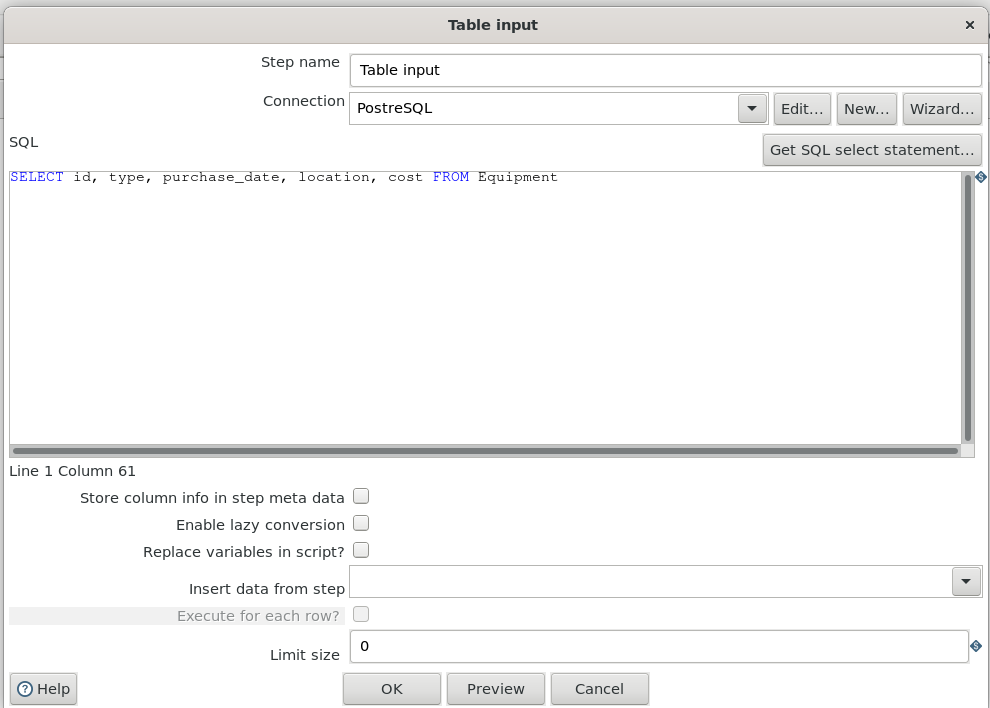


Рисунок 10. Запрос на вывод таблицы

Потом нужно сделать фильтрацию по типу оборудования (рисунок 11).

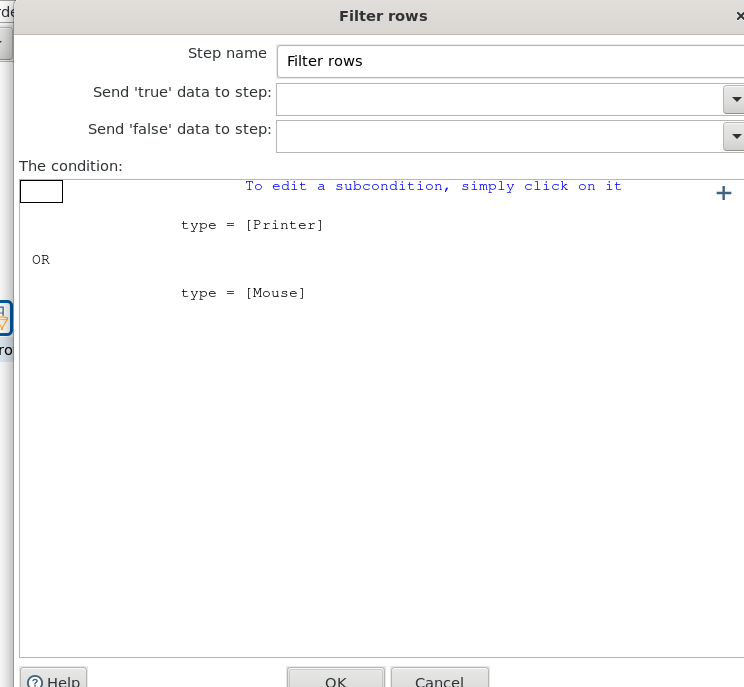


Рисунок 11. Фильтрация по типу оборудования

Общая трансформация выглядит следующим образом (рисунок 12).

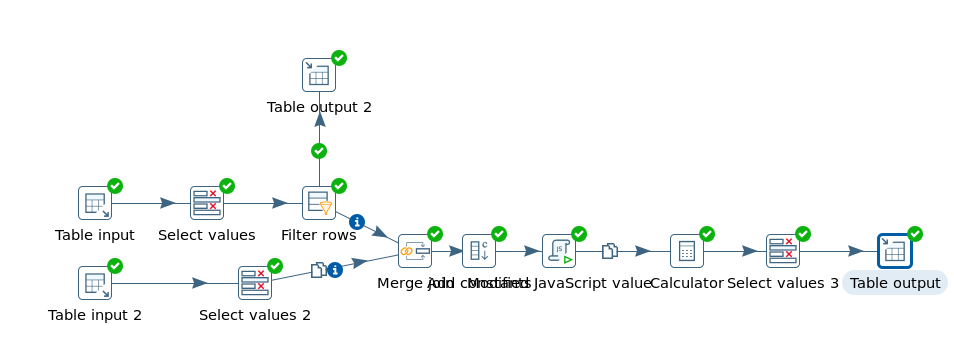


Рисунок 12. Общая трансформация

Нужно создать две констант salvage\_value и estimated\_useful\_life (рисунок 13).

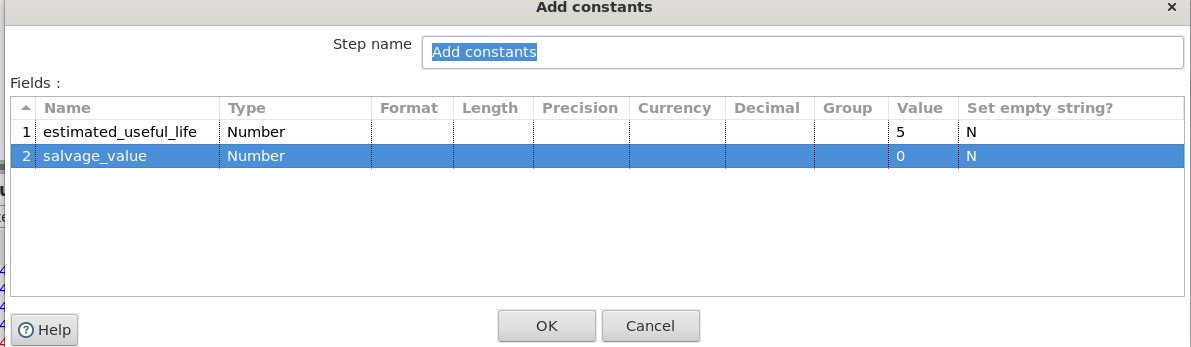


Рисунок 13. Создание двух констант

После надо для расчета разницы между датами в годах добавить компонент Modified Java Script Value и написать следующий код (рисунок 14).

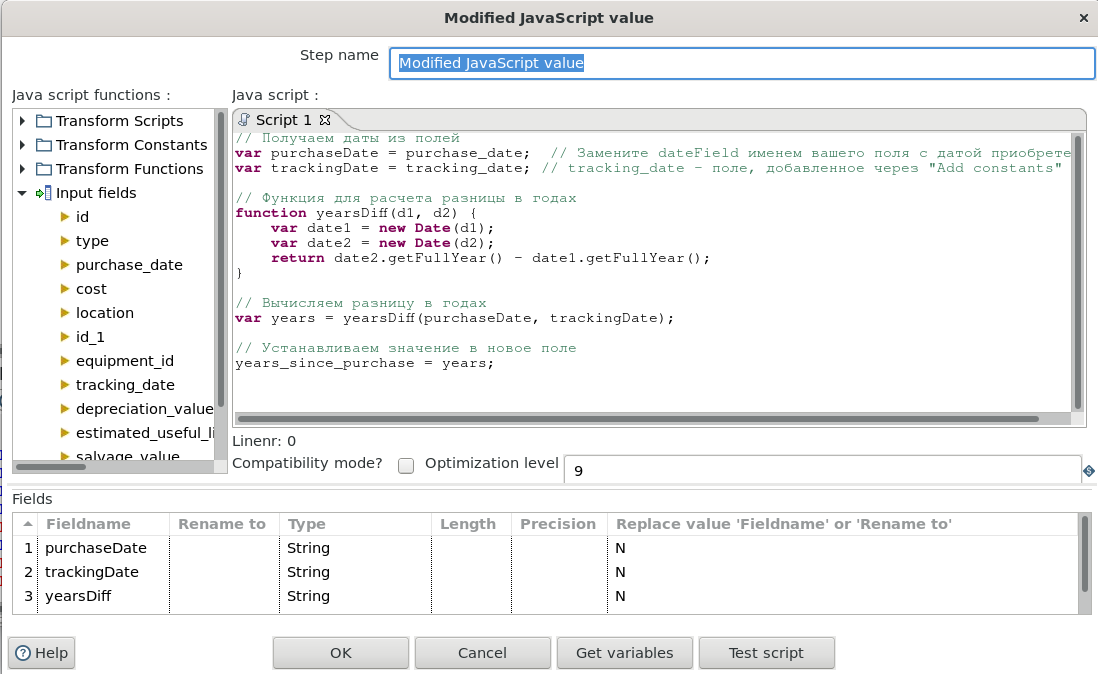


Рисунок 14. Расчет разницы между датами

После этого в компоненте калькулятор надо написать следующие формулы для расчета амортизации (рисунок 15).

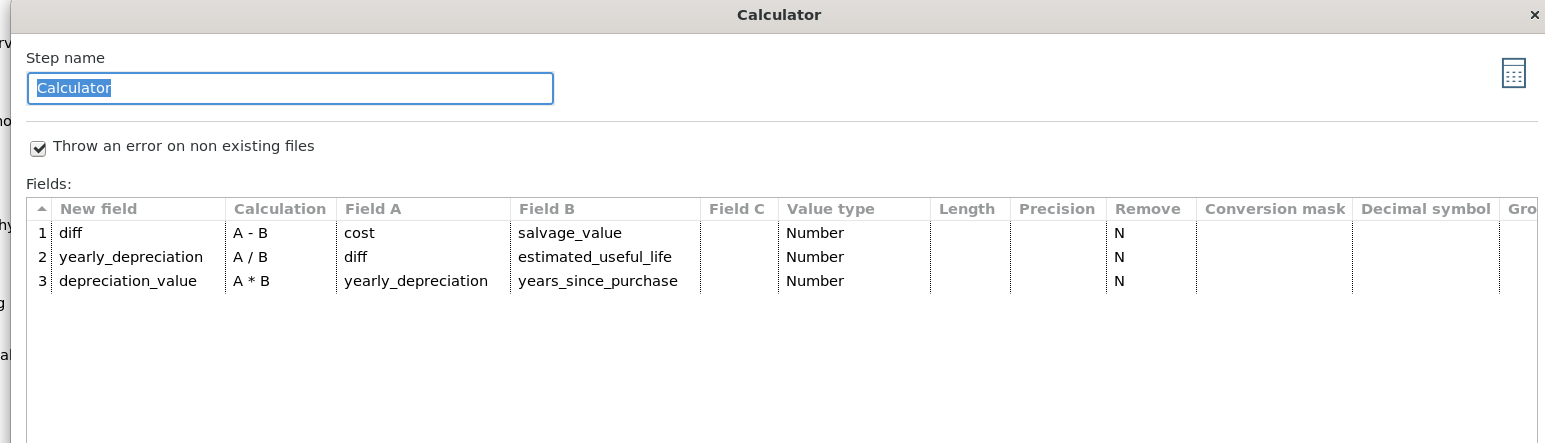


Рисунок 15. Формулы для расчета амортизации

Далее нужно выбрать следующие столбцы (рисунок 16).

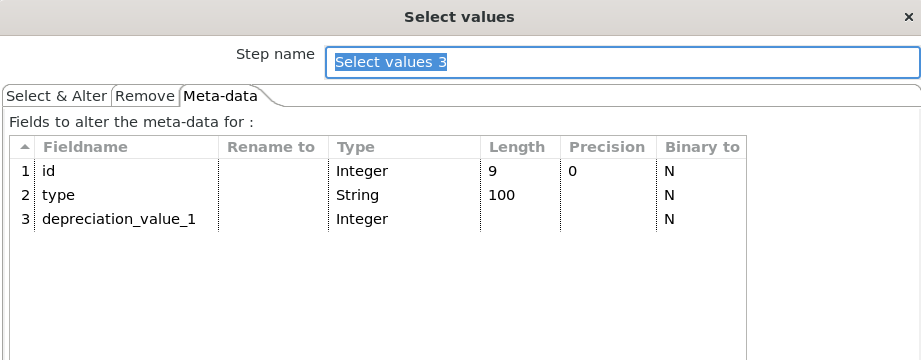


Рисунок 16. Выбор столбцов

Как можно увидеть на рисунке 17, таблицы Equipment успешно загрузилась в базу данных (рисунок 18).

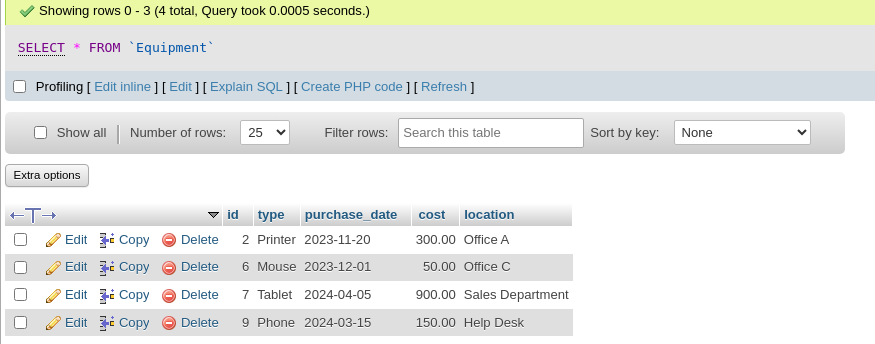


Рисунок 18. Таблица успешно загрузилась

На рисунке 19 таблица Equipment\_Tracking тоже успешно загрузилась в PhpMyAdmin.

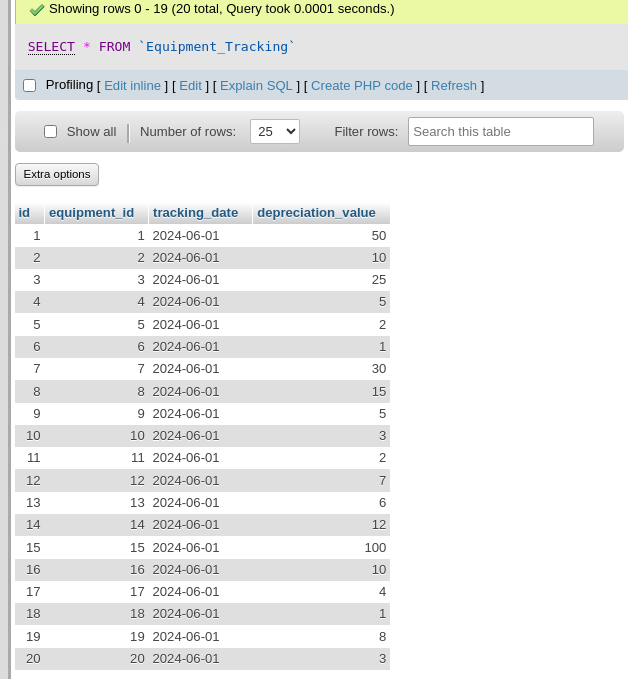


Рисунок 20. Таблица загрузилась

На рисунке 21 видно, что таблица с амортизацией загрузилась в PhpMyAdmin.

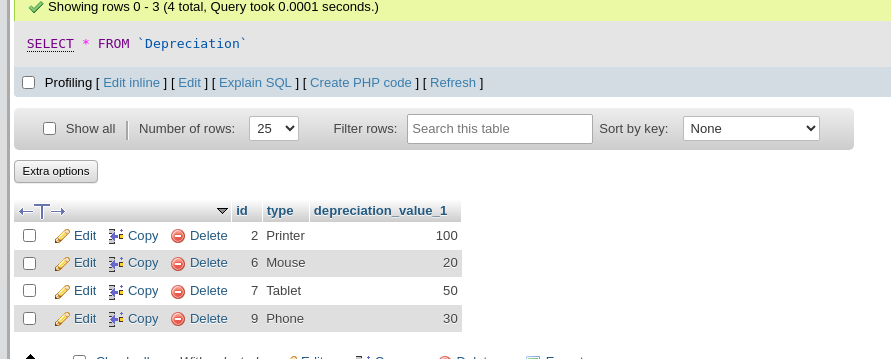


Рисунок 21. Таблица с амортизацией

**Выводы:**

− Были созданы исходные таблицы в PostgreSQL с различными наборами данных.

− Были настроены целевые таблицы в MySQL для приема данных.

− Был разработан процесс трансформации данных в Pentaho, а именно фильтрация и расчет амортизации.