Колледж Автономной некоммерческой образовательной организации высшего образования

«Научно-технологический университет «Сириус»

**Учебная практика №1**

***«Создание базы данных по предметной области»***

Работу подготовил:

Студент группы К0109-23

Куликов К. А

Проверил:

Картавых Е. В

Сириус, 2024

Содержание

[**Раздел «Введение»** 3](#_Toc187361996)

[**Раздел «Описание предметной области»** 4](#_Toc187361997)

[**Раздел «Проектирование БД»** 6](#_Toc187361998)

[**Раздел «Таблицы базы данных»** 11](#_Toc187361999)

[**Раздел «Связи»** 19](#_Toc187362000)

[**Раздел «Пример данных»** 23](#_Toc187362001)

[**Раздел «Заключение»** 28](#_Toc187362002)

## **Раздел «Введение»**

Название предметной области: плантации бананов в Колумбии.

Описание предметной области: в данной работе рассматривается создание базы данных, посвященной банановым плантациям в Колумбии. Основной целью базы является хранение, обработка и анализ данных о банановых плантациях, что включает в себя информацию о их расположении, производительности, сезонах сбора урожая и сотрудниках, работающих на плантациях. База данных должна помочь исследовать экономическую эффективность, отслеживать урожай и улучшать устойчивость производителей.

## **Раздел «Описание предметной области»**

В предметной области "Банановые плантации в Колумбии" выделяются следующие сущности:

1. Плантация (Plantation)
   * Атрибуты:
     + ID\_плантации (INT) - уникальный идентификатор плантации
     + Название (VARCHAR) - название плантации
     + Местоположение (VARCHAR) - географическое положение
     + Площадь (FLOAT) - площадь плантации в гектарах
     + Тип\_почвы (VARCHAR) - тип почвы
2. Урожай (Harvest)
   * Атрибуты:
     + ID\_урожая (INT) - уникальный идентификатор урожая
     + ID\_плантации (INT) - идентификатор плантации (связь с сущностью Плантация)
     + Год (YEAR) - год сбора урожая
     + Количество\_бананов (FLOAT) - количество собранных бананов в тоннах
     + Качество (VARCHAR) - категория качества урожая
3. Сотрудник (Employee)
   * Атрибуты:
     + ID\_сотрудника (INT) - уникальный идентификатор сотрудника
     + Имя (VARCHAR) - имя сотрудника
     + Должность (VARCHAR) - должность сотрудника
     + ID\_плантации (INT) - идентификатор плантации (связь с сущностью Плантация)
     + Дата\_начала\_работы (DATE) - дата начала работы в компании
4. Экспорт (Export)
   * Атрибуты:
     + ID\_экспорта (INT) - уникальный идентификатор экспорта
     + ID\_урожая (INT) - идентификатор урожая (связь с сущностью Урожай)
     + Дата\_экспорта (DATE) - дата экспорта бананов
     + Страна\_назначения (VARCHAR) - страна, куда отправляются бананы

Связи между сущностями

Плантация ↔ Урожай: Один к многим (одна плантация может иметь много урожаев).

Плантация ↔ Сотрудник: Один к многим (одна плантация может иметь много сотрудников).

Урожай ↔ Экспорт: Один к одному (каждый урожай может быть экспортирован один раз).

Эта структура позволяет эффективно управлять данными о банановых плантациях, их урожаях, сотрудниках и процессах экспорта, что способствует оптимизации бизнес-представлений.

## **Раздел «Проектирование БД»**

**3.1 Первичный список отношений**

Согласно списку сущностей, представленных в разделе «Описание предметной области», получаем первичный список отношений базы данных:

1. **Plantation** (Плантация)
   * ID\_plantation (INT) - уникальный идентификатор плантации
   * Name (VARCHAR) - название плантации
   * Location (VARCHAR) - местоположение
   * Area (FLOAT) - площадь плантации в гектарах
   * Soil\_type (VARCHAR) - тип почвы
2. **Harvest** (Урожай)
   * ID\_harvest (INT) - уникальный идентификатор урожая
   * ID\_plantation (INT) - идентификатор плантации (связь с сущностью Plantation)
   * Year (YEAR) - год сбора урожая
   * Quantity\_bananas (FLOAT) - количество собранных бананов в тоннах
   * Quality (VARCHAR) - категория качества урожая
3. **Employee** (Сотрудник)
   * ID\_employee (INT) - уникальный идентификатор сотрудника
   * First\_name (VARCHAR) - имя сотрудника
   * Last\_name (VARCHAR) - фамилия сотрудника
   * Position (VARCHAR) - должность сотрудника
   * ID\_plantation (INT) - идентификатор плантации (связь с сущностью Plantation)
   * Start\_date (DATE) - дата начала работы в компании
4. **Export** (Экспорт)
   * ID\_export (INT) - уникальный идентификатор экспорта
   * ID\_harvest (INT) - идентификатор урожая (связь с сущностью Harvest)
   * Export\_date (DATE) - дата экспорта бананов
   * Destination\_country (VARCHAR) - страна назначения
   * Quantity\_bananas (FLOAT) - количество экспортируемых бананов в тоннах

**3.2 Анализ отношения «Плантация»**

**3.2.1 Проверка на соответствие 1НФ**

1НФ требует:

* уникальной идентификации каждой записи в отношении (реализуется с помощью первичного ключа).
* атомарности значений для каждого атрибута.

В отношении **Plantation** все атрибуты уже атомарны, так как:

* **ID\_plantation** - уникальный идентификатор.
* **Name**, **Location**, **Area**, **Soil\_type** — все эти атрибуты уже атомарные, и значения не могут быть разделены на более мелкие компоненты.

Результат: отношение **Plantation** удовлетворяет требованиям 1НФ.

**3.2.2 Проверка на соответствие 2НФ**

2НФ требует:

* Отношение должно удовлетворять 1НФ.
* Каждый неключевой атрибут должен зависеть от всего первичного ключа.

В данном случае, атрибуты **Name**, **Location**, **Area** и **Soil\_type** полностью зависят от первичного ключа **ID\_plantation**, так как каждый из этих атрибутов описывает только одну плантацию.

Результат: отношение **Plantation** удовлетворяет требованиям 2НФ.

**3.2.3 Проверка на соответствие 3НФ**

3НФ требует:

* Отношение должно удовлетворять 2НФ.
* Все неключевые атрибуты должны быть независимы друг от друга (исключение транзитивных зависимостей).

В отношении **Plantation** есть транзитивные зависимости, так как не все атрибуты напрямую зависят от первичного ключа **ID\_plantation**. А именно Soil\_type, для него нужно создать отдельную таблицу.

Результат: отношение **Plantation** удовлетворяет требованиям 3НФ.

**3.3 Анализ отношения «Урожай»**

**3.3.1 Проверка на соответствие 1НФ**

В отношении **Harvest**:

* **ID\_harvest** - уникальный идентификатор урожая.
* **ID\_plantation** - ссылка на плантацию.
* **Year** - год сбора урожая.
* **Quantity\_bananas** - количество собранных бананов.
* **Quality** - категория качества урожая.

Все атрибуты являются атомарными, и нет множественных значений в одном атрибуте.

Результат: отношение **Harvest** удовлетворяет требованиям 1НФ.

**3.3.2 Проверка на соответствие 2НФ**

В отношении **Harvest**:

* Каждый атрибут зависит от всего первичного ключа, который состоит из **ID\_harvest**. Все атрибуты (кроме **ID\_plantation**) зависят от **ID\_harvest**.
* **ID\_plantation** является внешним ключом, и это не нарушает 2НФ, так как **ID\_plantation** не является частью первичного ключа.

Результат: отношение **Harvest** удовлетворяет требованиям 2НФ.

**3.3.3 Проверка на соответствие 3НФ**

В отношении **Harvest**:

* Все атрибуты, такие как **Year**, **Quantity\_bananas**, зависят от первичного ключа **ID\_harvest**. Нет транзитивных зависимостей между аттрибутами, поэтому отношение соответствует 3НФ.
* Но для **Quality** - категория качества урожая нужно создать отдельную таблицу, так как она не проходит проверку 3НФ.

Результат: отношение **Harvest** удовлетворяет требованиям 3НФ.

**3.4 Анализ отношения «Сотрудник»**

**3.4.1 Проверка на соответствие 1НФ**

Отношение **Employee** содержит атомарные значения:

* **ID\_employee** - уникальный идентификатор сотрудника.
* **First\_name**, **Last\_name** - имя и фамилия сотрудника.
* **Position** - должность.
* **ID\_plantation** - ссылка на плантацию.
* **Start\_date** - дата начала работы.

Результат: отношение **Employee** удовлетворяет требованиям 1НФ.

**3.4.2 Проверка на соответствие 2НФ**

Отношение **Employee**:

* Каждый атрибут зависит от первичного ключа **ID\_employee**.
* **ID\_plantation** является внешним ключом, но не влияет на зависимость атрибутов от **ID\_employee**.

Результат: отношение **Employee** удовлетворяет требованиям 2НФ.

**3.4.3 Проверка на соответствие 3НФ**

В отношении **Employee** есть транзитивная зависимость, так как атрибуты **First\_name**, **Last\_name**, **ID\_plantation**, и **Start\_date** напрямую зависят от **ID\_employee**, но кроме Position. Для этого нужно создать отдельную таблицу.

Результат: отношение **Employee** удовлетворяет требованиям 3НФ.

**3.5 Анализ отношения «Экспорт»**

**3.5.1 Проверка на соответствие 1НФ**

Отношение **Export** содержит атомарные значения:

* **ID\_export** - уникальный идентификатор экспорта.
* **ID\_harvest** - ссылка на урожай.
* **Export\_date** - дата экспорта.
* **Destination\_country** - страна назначения.

Результат: отношение **Export** удовлетворяет требованиям 1НФ.

**3.5.2 Проверка на соответствие 2НФ**

Отношение **Export**:

* Все атрибуты зависят от первичного ключа **ID\_export**.
* **ID\_harvest** является внешним ключом и не нарушает зависимость.

Результат: отношение **Export** удовлетворяет требованиям 2НФ.

**3.5.3 Проверка на соответствие 3НФ**

В отношении **Export** нет транзитивных зависимостей. Все атрибуты зависят от **ID\_export** и не зависят друг от друга, кроме Destination\_country - страна назначения. Создадим для этого отдельную таблицу.

Результат: отношение **Export** удовлетворяет требованиям 3НФ.

# **Раздел «Таблицы базы данных»**

**Таблица для сущности "Плантация"**

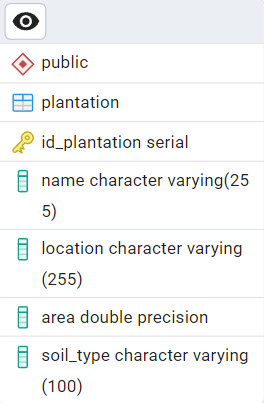
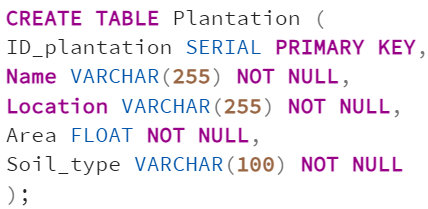
****

Таблица для сущности "Плантация"

****

Скрипт для создания таблицы

**Таблица для сущности "Урожай"**

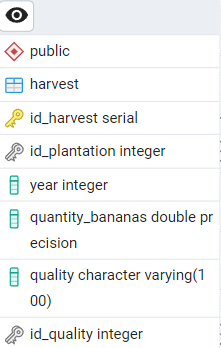
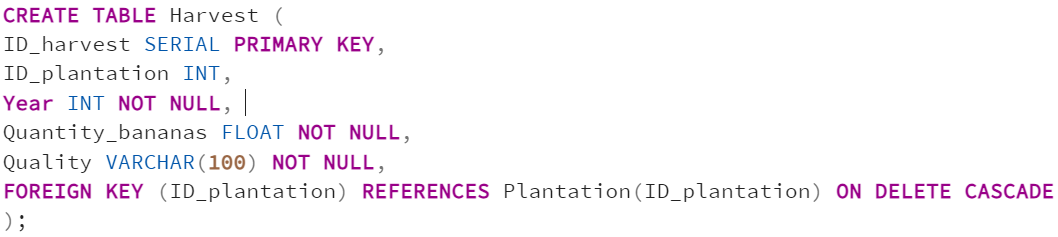
****

Таблица для сущности "Урожай"



Скрипт для создания таблицы

**Таблица для сущности "Сотрудник"**

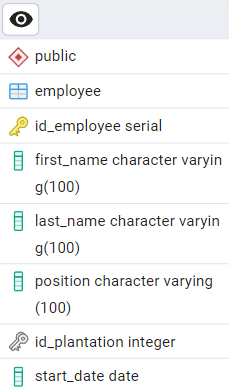
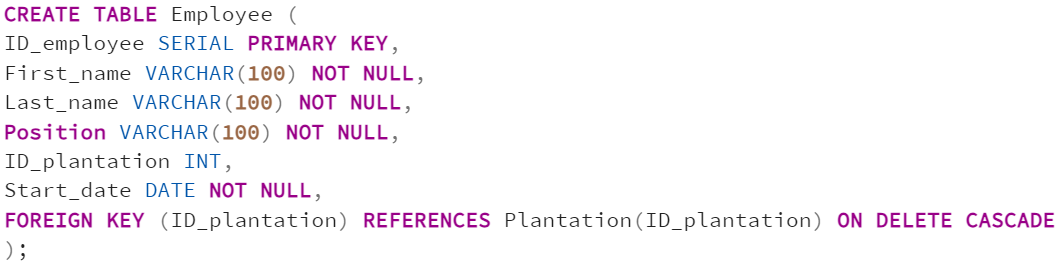
****

Таблица для сущности "Сотрудник"



Скрипт для создания таблицы

**Таблица для сущности "Экспорт"**

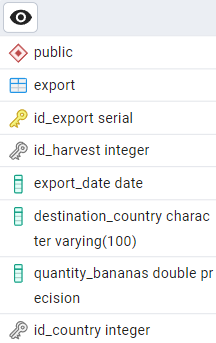
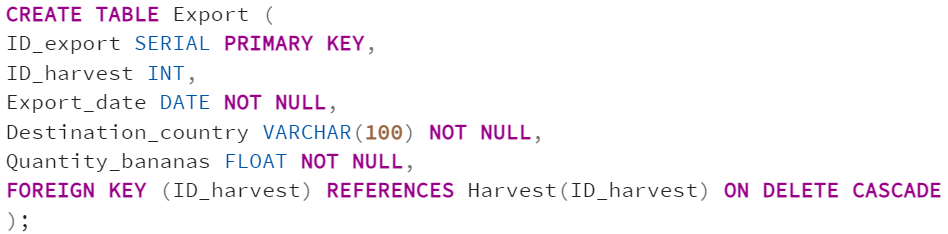
****

Таблица для сущности "Экспорт"



Скрипт для создания таблицы

**Создание отдельной таблицы для "Качества урожая" (для нормализации)**

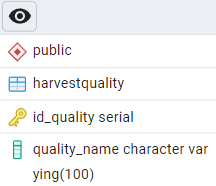
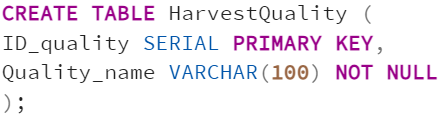
****

Таблица для сущности "Качества урожая"



Скрипт для создания таблицы

Теперь нужно обновить таблицу Harvest, чтобы использовать ссылку на таблицу качества.



Скрипт для обновления таблицы

**Таблица для сущности "Страны назначения" (для нормализации)**

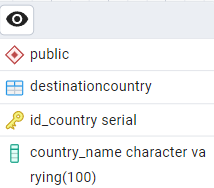
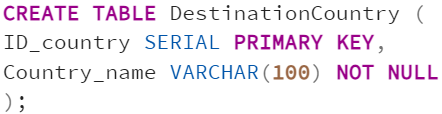
****

Таблица для сущности "Страны назначения" (для нормализации)



Скрипт для создания таблицы

Теперь обновим таблицу Export, чтобы использовать ссылку на таблицу стран.

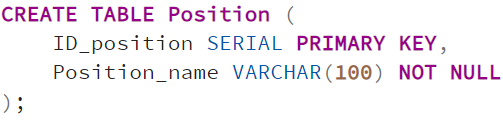


Скрипт для обновления таблицы

**Таблица для сущности "Должность" (для нормализации)**

****

Таблица для сущности "Должность" (для нормализации)



Скрипт для создания таблицы

**Таблица для сущности "Тип почвы" (для нормализации)**

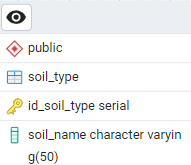
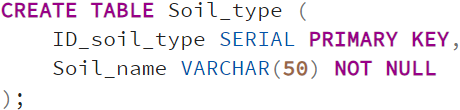
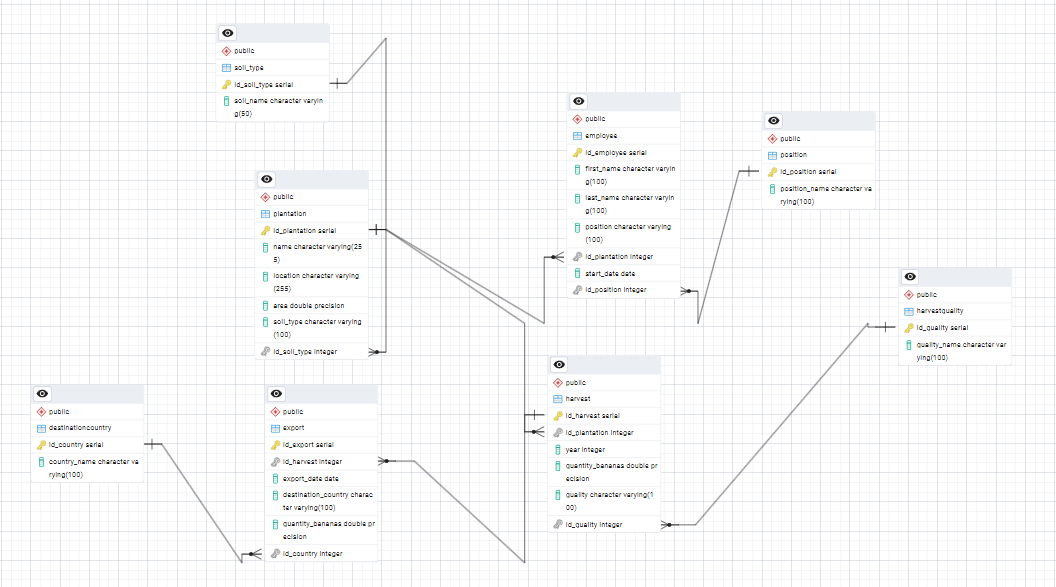
****

Таблица для сущности "Тип почвы" (для нормализации)



Скрипт для создания таблицы

# **Раздел «Связи»**



Всего 8 таблиц. Все отношения созданы в формате Foreign Key.

**1. Таблица Harvest**

Таблица **Harvest** имеет поле ID\_plantation, которое является внешним ключом. Оно ссылается на таблицу **Plantation**.

FOREIGN KEY (ID\_plantation) REFERENCES Plantation(ID\_plantation)

**Описание:**  
Поле ID\_plantation в таблице Harvest ссылается на таблицу Plantation, которая содержит информацию о плантациях. Каждое урожайное событие связано с конкретной плантацией, и это отношение создается с помощью внешнего ключа.

**2. Таблица Employee**

Таблица **Employee** имеет поле ID\_plantation, которое является внешним ключом. Оно ссылается на таблицу **Plantation**.

FOREIGN KEY (ID\_plantation) REFERENCES Plantation(ID\_plantation) ON DELETE CASCADE

**Описание:**  
Поле ID\_plantation в таблице Employee ссылается на таблицу Plantation, чтобы указать, на какой плантации работает сотрудник. При удалении записи в таблице Plantation, все связанные сотрудники будут удалены, благодаря ON DELETE CASCADE.

**3. Таблица Export**

Таблица **Export** имеет поле ID\_harvest, которое является внешним ключом. Оно ссылается на таблицу **Harvest**.

FOREIGN KEY (ID\_harvest) REFERENCES Harvest(ID\_harvest) ON DELETE CASCADE

**Описание:**  
Поле ID\_harvest в таблице Export ссылается на таблицу Harvest, что указывает, какой урожай был экспортирован. Также используется ограничение ON DELETE CASCADE, которое автоматически удаляет связанные записи из таблицы Export, если удаляется запись из таблицы Harvest.

**4. Таблица HarvestQuality**

Таблица **HarvestQuality** содержит поле Quality\_name, которое представляет собой описание категории качества урожая. Хотя связь между таблицей HarvestQuality и другими таблицами не указана напрямую в вашей схеме, можно предположить, что таблица Harvest могла бы иметь поле, ссылающееся на HarvestQuality (например, добавив в таблицу Harvest внешний ключ, ссылающийся на ID\_quality).

Пример добавления внешнего ключа:

ALTER TABLE Harvest

ADD COLUMN ID\_quality INT,

ADD FOREIGN KEY (ID\_quality) REFERENCES HarvestQuality(ID\_quality);

**Описание:**  
Каждый урожай может иметь определённую категорию качества, которая хранится в таблице HarvestQuality. Поле ID\_quality в таблице Harvest будет ссылаться на таблицу HarvestQuality, что позволяет указать качество каждого урожая.

**5. Таблица DestinationCountry**

Таблица **Export** также может содержать поле, которое будет ссылаться на таблицу **DestinationCountry**, если мы хотим хранить информацию о странах назначения для экспортируемых бананов.

Пример добавления внешнего ключа в таблицу Export:

ALTER TABLE Export

ADD COLUMN ID\_country INT,

ADD FOREIGN KEY (ID\_country) REFERENCES DestinationCountry(ID\_country);

**Описание:**  
Поле ID\_country в таблице Export будет ссылается на таблицу DestinationCountry, что позволяет указать страну назначения для экспортируемых бананов.

**6. Таблица Soil\_type**

Таблица **Soil\_type** имеет поле **ID\_soil\_type**, которое является первичным ключом и используется в таблице **Plantation**.

FOREIGN KEY (ID\_soil\_type) REFERENCES Soil\_type(ID\_soil\_type) ON DELETE SET NULL

**Описание**:  
Поле **ID\_soil\_type** в таблице **Plantation** ссылается на таблицу **Soil\_type**, чтобы указать тип почвы для каждой плантации. При удалении записи в таблице **Soil\_type**, ссылки на нее в таблице **Plantation** будут установлены в NULL, благодаря ON DELETE SET NULL.

**7. Таблица Position**

Таблица **Position** имеет поле **ID\_position**, которое является первичным ключом и используется в таблице **Employee**.

FOREIGN KEY (ID\_position) REFERENCES Position(ID\_position) ON DELETE SET NULL

**Описание**:  
Поле **ID\_position** в таблице **Employee** ссылается на таблицу **Position**, чтобы указать должность сотрудника. При удалении записи в таблице **Position**, должность сотрудника будет установлена в NULL, благодаря ON DELETE SET NULL.

**8. Таблица Plantation**

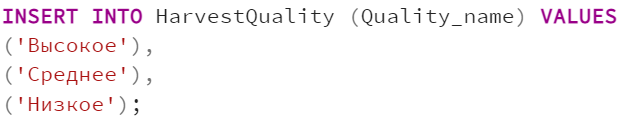
Таблица **Plantation** имеет поле **ID\_plantation**, которое является первичным ключом и используется в таблицах **Employee** и **Harvest**.

FOREIGN KEY (ID\_plantation) REFERENCES Plantation(ID\_plantation) ON DELETE CASCADE

**Описание**:  
Поле **ID\_plantation** в таблицах **Employee** и **Harvest** ссылается на таблицу **Plantation**, чтобы указать, на какой плантации работает сотрудник или был собран урожай. При удалении записи в таблице **Plantation**, все связанные сотрудники и урожаи будут удалены, благодаря ON DELETE CASCADE.

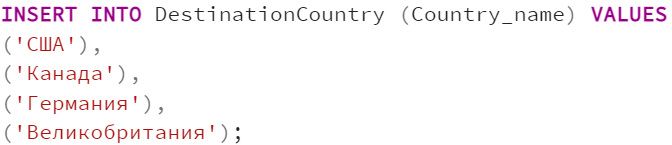
# **Раздел «Пример данных»**

Наполнение таблицы HarvestQuality



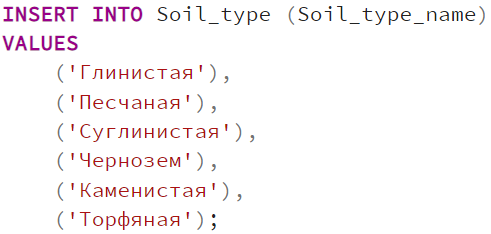
Скрипт для внесения значений в таблицу

Наполнение таблицы DestinationCountry



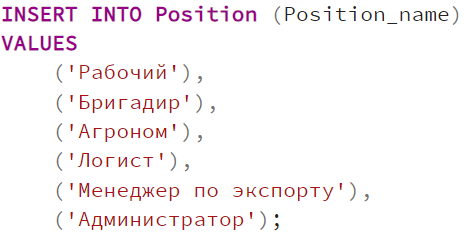
Скрипт для внесения значений в таблицу

Наполнение таблицы Soil\_type



Скрипт для внесения значений в таблицу

Наполнение таблицы Position



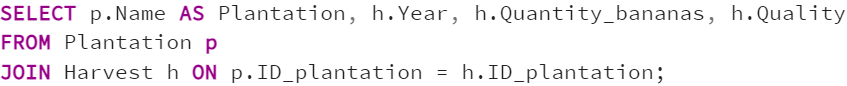
Скрипт для внесения значений в таблицу

**Пример запросов для работы с базой данных**

Получить все плантации с их урожаями



Вывод запроса

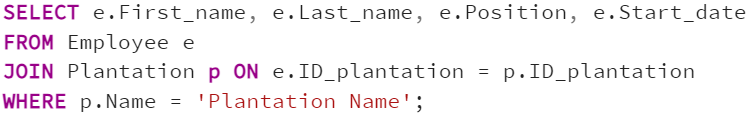


Скрипт запроса

Получить сотрудников, работающих на определённой плантации



Вывод запроса

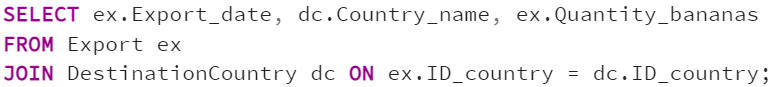


Скрипт запроса

Получить экспортированные бананы по странам назначения



Вывод запроса



Скрипт запроса

**Дополнительные улучшения**

Индексы: Можно создать дополнительные индексы для ускорения поиска, например, по столбцам Location, Year, Export\_date, и Country\_name.

Триггеры и хранимые процедуры: Можно добавить логику для автоматического расчёта количества бананов на каждой плантации или для автоматического обновления данных об урожаях.

**Сводка**

* **Таблица Harvest** имеет внешний ключ ID\_plantation, ссылающийся на таблицу **Plantation**.
* **Таблица Employee** имеет внешний ключ ID\_plantation, ссылающийся на таблицу **Plantation**.
* **Таблица Export** имеет внешний ключ ID\_harvest, ссылающийся на таблицу **Harvest** и может иметь внешний ключ ID\_country, ссылающийся на таблицу **DestinationCountry**.
* **Таблица HarvestQuality** не имеет явных внешних ключей, но может быть связана с таблицей Harvest через внешний ключ ID\_quality.

**Пример схемы связей между таблицами**

* **Harvest**: ссылается на **Plantation** через поле ID\_plantation.
* **Employee**: ссылается на **Plantation** через поле ID\_plantation.
* **Export**: ссылается на **Harvest** через поле ID\_harvest и может ссылается на **DestinationCountry** через поле ID\_country.
* **HarvestQuality**: может быть связана с **Harvest** через поле ID\_quality (через внешний ключ).

# **Раздел «Заключение»**

В данной работе была рассмотрена предметная область, связанная с банановыми плантациями в Колумбии. Основной целью являлось проектирование базы данных, способной эффективно хранить, обрабатывать и анализировать данные о плантациях, урожаях, сотрудниках и экспорте продукции.

В ходе проектирования были выделены основные сущности предметной области: "Плантация", "Урожай", "Сотрудник" и "Экспорт". Для каждой сущности были определены атрибуты и проведен процесс нормализации данных, что позволило обеспечить соответствие первой, второй и третьей нормальной форме. Это минимизировало дублирование данных и исключило транзитивные зависимости.

Дополнительно были созданы вспомогательные таблицы для нормализации отдельных атрибутов, таких как "Качество урожая", "Тип почвы", "Страна назначения" и "Должности сотрудников". Это позволило упростить структуру данных и обеспечить гибкость при их обновлении и анализе.

Представленная база данных обеспечивает четкую организацию и взаимосвязь данных, что делает её пригодной для решения следующих задач:

* мониторинг производительности плантаций и качества урожая;
* анализ эффективности работы сотрудников;
* оптимизация экспорта продукции в зависимости от стран назначения.

Проектирование базы данных и представленные SQL-запросы демонстрируют её применимость для решения задач управления данными в сфере сельского хозяйства. В дальнейшем возможны доработки, такие как добавление триггеров, индексов и хранимых процедур, для повышения производительности и автоматизации процессов.

Таким образом, работа над проектированием базы данных для банановых плантаций в Колумбии подтверждает её значимость и потенциал для оптимизации управления производственными процессами в аграрной сфере.