**СУБД SQL Server**

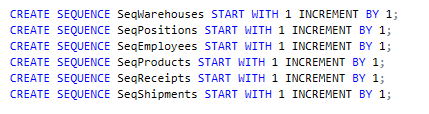
**“Склад”**

**1. Последовательности**

Для автоматической генерации уникальных идентификаторов (ключей) для каждой

таблицы созданы следующие последовательности:

* **SeqWarehouses:** Используется для автоматической генерации ID в таблице Warehouses.
* **SeqPositions:** Используется для автоматической генерации ID в таблице Positions.
* **SeqEmployees:** Используется для автоматической генерации ID в таблице Employees.
* **SeqProducts:** Используется для автоматической генерации ID в таблице Products.
* **SeqReceipts:** Используется для автоматической генерации ID в таблице Receipts.
* **SeqShipments:** Используется для автоматической генерации ID в таблице Shipments.

****

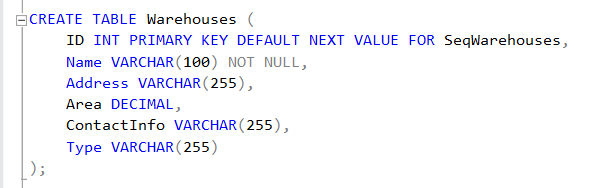
**2. Таблицы**

Для хранения данных созданы следующие таблицы:

* **Warehouses:** Хранит информацию о складах (ID, название, адрес, площадь, контактная информация, тип).
* **Positions:** Хранит информацию о должностях (ID, название, описание, уровень доступа, ID склада, признак активности).
* **Employees:** Хранит информацию о сотрудниках (ID, ФИО, ID должности, ID склада, контактная информация, статус).
* **Products:** Хранит информацию о продуктах (ID, название, артикул, описание, единица измерения, цена).
* **Receipts:** Хранит информацию о поступлениях (ID, дата, поставщик, номер накладной, ID склада, общая сумма).
* **Shipments:** Хранит информацию об отгрузках (ID, дата, клиент, номер накладной, ID склада, общая сумма).

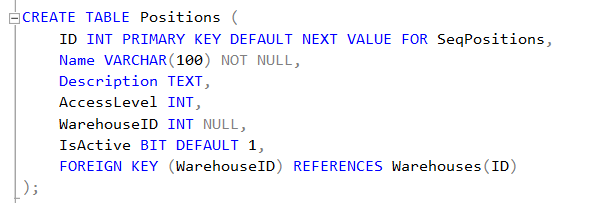
**Таблица Warehouses**

| **Поле** | **Тип данных** | **Описание** | **Ограничения** |
| --- | --- | --- | --- |
| ID | INT | Уникальный идентификатор склада | PRIMARY KEY, DEFAULT NEXT VALUE FOR SeqWarehouses |
| Name | VARCHAR(100) | Название склада | NOT NULL |
| Address | VARCHAR(255) | Адрес склада |  |
| Area | DECIMAL | Площадь склада |  |
| ContactInfo | VARCHAR(255) | Контактная информация склада |  |
| Type | VARCHAR(255) | Тип склада |  |

****

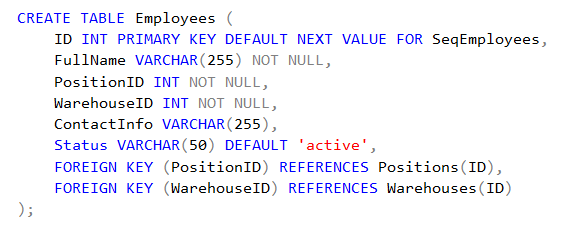
**Таблица Positions**

| **Поле** | **Тип данных** | **Описание** | **Ограничения** |
| --- | --- | --- | --- |
| ID | INT | Уникальный идентификатор должности. | PRIMARY KEY, DEFAULT NEXT VALUE FOR SeqPositions |
| Name | VARCHAR(100) | Название должности. | NOT NULL |
| Description | TEXT | Описание должности. |  |
| AccessLevel | INT | Уровень доступа должности. |  |
| WarehouseID | INT | ID склада, к которому относится должность | FOREIGN KEY (WarehouseID) REFERENCES Warehouses(ID) |
| IsActive | BIT | Признак активности должности | DEFAULT 1 |

****

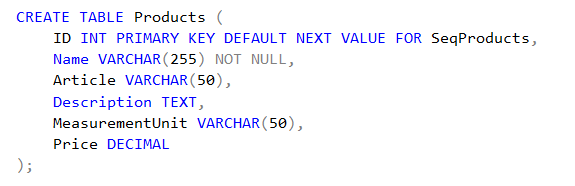
**Таблица Employees**

| **Поле** | **Тип данных** | **Описание** | **Ограничения** |
| --- | --- | --- | --- |
| ID | INT | Уникальный идентификатор сотрудника. | PRIMARY KEY, DEFAULT NEXT VALUE FOR SeqEmployees |
| FullName | VARCHAR(255) | ФИО сотрудника. | NOT NULL |
| PositionID | INT | ID должности сотрудника. | FOREIGN KEY (PositionID) REFERENCES Positions(ID), NOT NULL |
| WarehouseID | INT | ID склада, к которому относится сотрудник. | FOREIGN KEY (WarehouseID) REFERENCES Warehouses(ID), NOT NULL |
| ContactInfo | VARCHAR(255) | Контактная информация сотрудника. |  |
| Status | VARCHAR(50) | Статус сотрудника (например, 'active', 'inactive'). | DEFAULT 'active' |

****

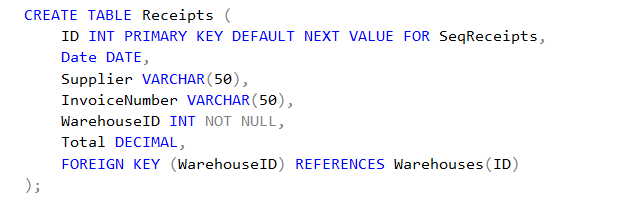
**Таблица Products**

| **Поле** | **Тип данных** | **Описание** | **Ограничения** |
| --- | --- | --- | --- |
| ID | INT | Уникальный идентификатор продукта. | PRIMARY KEY, DEFAULT NEXT VALUE FOR SeqProducts |
| Name | VARCHAR(255) | Название продукта. | NOT NULL |
| Article | VARCHAR(50) | Артикул продукта. |  |
| Description | TEXT | Описание продукта. |  |
| MeasurementUnit | VARCHAR(50) | Единица измерения продукта. |  |
| Price | DECIMAL | Цена продукта. |  |

****

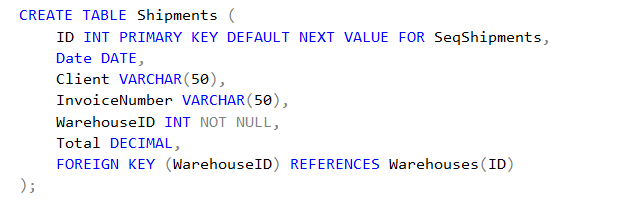
**Таблица Receipts**

| **Поле** | **Тип данных** | **Описание** | **Ограничения** |
| --- | --- | --- | --- |
| ID | INT | Уникальный идентификатор поступления. | PRIMARY KEY, DEFAULT NEXT VALUE FOR SeqReceipts |
| Date | DATE | Дата поступления. |  |
| Supplier | VARCHAR(50) | Поставщик. |  |
| InvoiceNumber | VARCHAR(50) | Номер накладной. |  |
| WarehouseID | INT | ID склада, на который поступил товар. | FOREIGN KEY (WarehouseID) REFERENCES Warehouses(ID), NOT NULL |
| Total | DECIMAL | Общая сумма поступления. |  |

****

**Таблица Shipments**

| **Поле** | **Тип данных** | **Описание** | **Ограничения** |
| --- | --- | --- | --- |
| ID | INT | Уникальный идентификатор отгрузки. | PRIMARY KEY, DEFAULT NEXT VALUE FOR SeqShipments |
| Date | DATE | Дата отгрузки. |  |
| Client | VARCHAR(50) | Клиент. |  |
| InvoiceNumber | VARCHAR(50) | Номер накладной. |  |
| WarehouseID | INT | ID склада, с которого был отгружен товар. | FOREIGN KEY (WarehouseID) REFERENCES Warehouses(ID), NOT NULL |
| Total | DECIMAL | Общая сумма отгрузки. |  |



**3. Индексы**

В рамках оптимизации запросов были созданы следующие индексы:

**1. IX\_Receipts\_WarehouseID (Receipts.WarehouseID):**

* Цель: Быстрый поиск поступлений по складу.
* CREATE INDEX IX\_Receipts\_WarehouseID ON Receipts (WarehouseID);

**2. IX\_Shipments\_WarehouseID (Shipments.WarehouseID):**

* Цель: Быстрый поиск отгрузок по складу.
* CREATE INDEX IX\_Shipments\_WarehouseID ON Shipments (WarehouseID);

**3. IX\_Employees\_PositionID (Employees.PositionID):**

* Цель: Быстрый поиск сотрудников по должности.
* CREATE INDEX IX\_Employees\_PositionID ON Employees (PositionID);

**4. IX\_Employees\_WarehouseID (Employees.WarehouseID):**

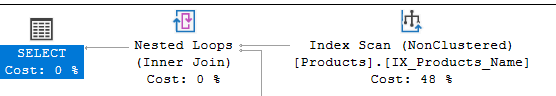
* Цель: Быстрый поиск сотрудников по складу.
* CREATE INDEX IX\_Employees\_WarehouseID ON Employees (WarehouseID);

**5. IX\_Positions\_WarehouseID (Positions.WarehouseID):**

* Цель: Быстрый поиск должностей по складу.
* CREATE INDEX IX\_Positions\_WarehouseID ON Positions (WarehouseID);

**6. IX\_Products\_Name (Products.Name):**

* Цель: Быстрый поиск продуктов по названию.
* CREATE INDEX IX\_Products\_Name ON Products (Name);

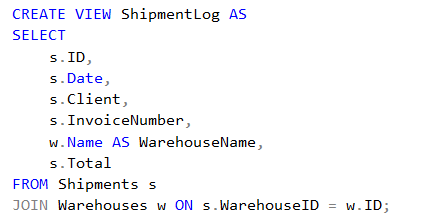


**4. Представления**

Представления – виртуальные таблицы, основанные на SQL-запросах. Они упрощают доступ к данным, скрывают сложность базовых таблиц и повышают безопасность.

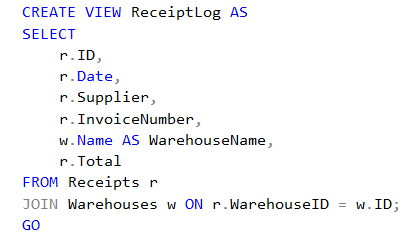
**ShipmentLog (Журнал отгрузок):** Представление объединяет информацию из таблиц Shipments и Warehouses, предоставляя упрощенный вид журнала отгрузок с указанием названия склада.

* + **Столбцы:**
    - ID: ID отгрузки (из таблицы Shipments).
    - Date: Дата отгрузки (из таблицы Shipments).
    - Client: Клиент (из таблицы Shipments).
    - InvoiceNumber: Номер накладной (из таблицы Shipments).
    - WarehouseName: Название склада (из таблицы Warehouses).
    - Total: Общая сумма отгрузки (из таблицы Shipments).
  + **Преимущества:**
    - Упрощает запросы для получения информации об отгрузках, так как не нужно явно указывать JOIN между таблицами Shipments и Warehouses.
    - Обеспечивает единообразный формат данных для отчетов и аналитики.
    - Позволяет скрыть структуру базовых таблиц от пользователей, которым не требуется полный доступ к данным.



**ReceiptLog (Журнал поступлений):** Представление объединяет информацию из таблиц Receipts и Warehouses, предоставляя упрощенный вид журнала поступлений с указанием названия склада.

* + **Столбцы:**
    - ID: ID поступления (из таблицы Receipts).
    - Date: Дата поступления (из таблицы Receipts).
    - Supplier: Поставщик (из таблицы Receipts).
    - InvoiceNumber: Номер накладной (из таблицы Receipts).
    - WarehouseName: Название склада (из таблицы Warehouses).
    - Total: Общая сумма поступления (из таблицы Receipts).
  + **Преимущества:**
    - Аналогично ShipmentLog, упрощает запросы для получения информации о поступлениях.
    - Обеспечивает консистентность данных для отчетов о поступлениях.
    - Повышает безопасность, ограничивая доступ к базовым таблицам.

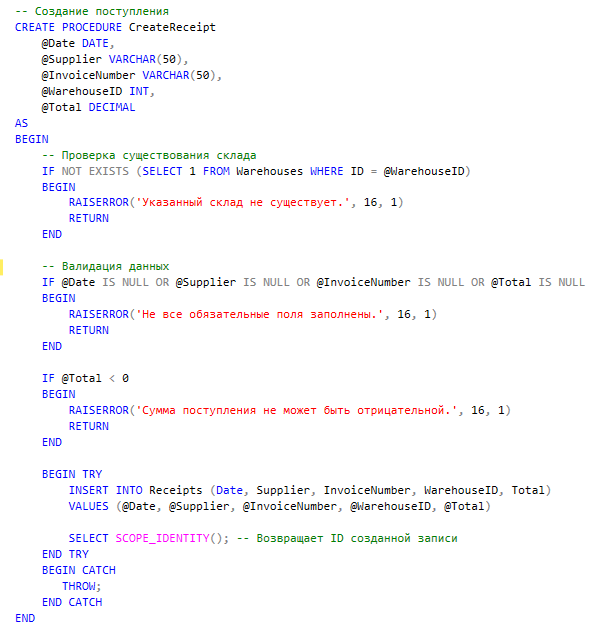


**5. Процедуры**

**1. Создание поступления**

**Процедура: CreateReceipt**

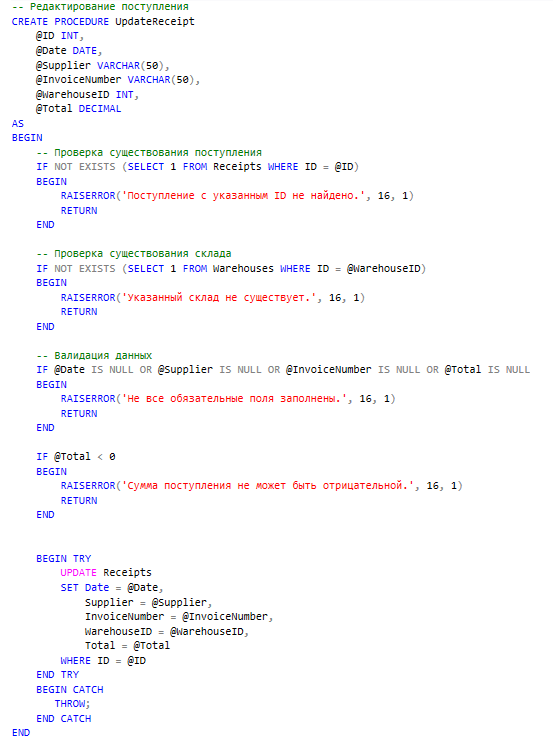
* **Параметры:**
  + @Date: дата поступления
  + @Supplier: поставщик
  + @InvoiceNumber: номер накладной
  + @WarehouseID: ID склада
  + @Total: общая сумма
* **Проверки:**
  + Проверяет существование склада.
  + Проверяет заполненность обязательных полей.
  + Проверяет, что сумма не отрицательная.
* **Действия:** Вставляет запись в таблицу Receipts.



**2. Редактирование поступления**

**Процедура: UpdateReceipt**

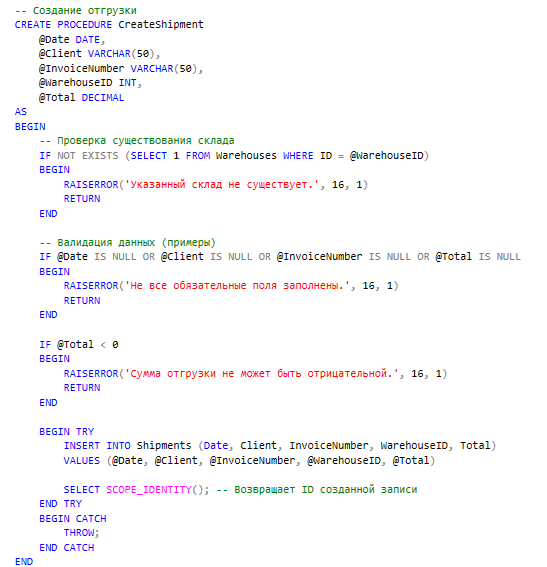
* **Параметры:**
  + @ID: ID поступления
  + @Date, @Supplier, @InvoiceNumber, @WarehouseID, @Total: обновляемые значения
* **Проверки:** Аналогичны процедуре создания.
* **Действия:** Обновляет запись в таблице Receipts.



**3. Создание отгрузки**

**Процедура: CreateShipment**

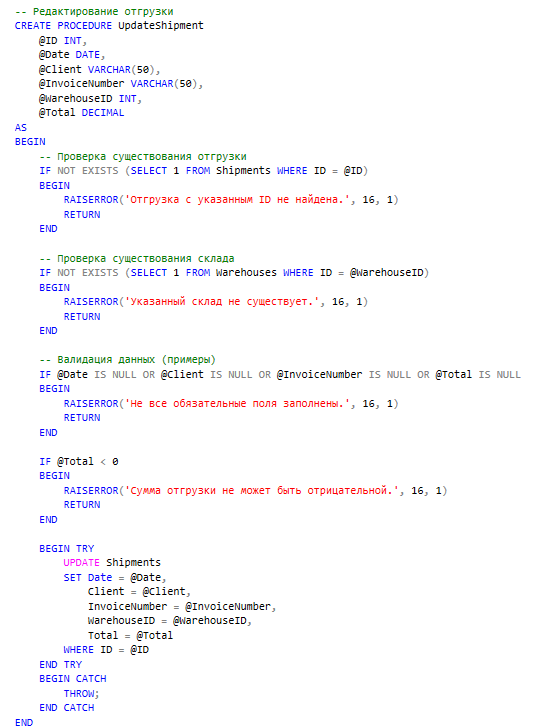
* **Параметры:**
  + @Date, @Client, @InvoiceNumber, @WarehouseID, @Total
* **Проверки:** Аналогичны процедуре создания поступления.
* **Действия:** Вставляет запись в таблицу Shipments.



**4. Редактирование отгрузки**

**Процедура: UpdateShipment**

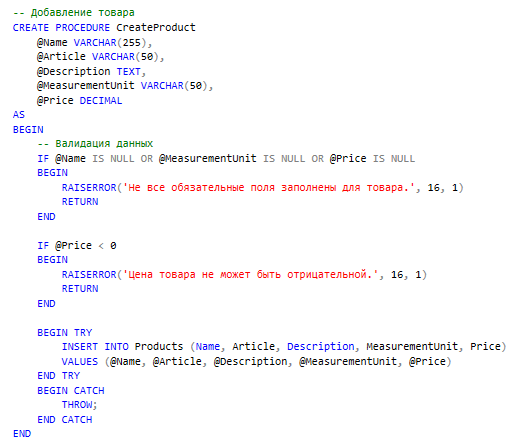
* **Параметры:**
  + @ID, @Date, @Client, @InvoiceNumber, @WarehouseID, @Total
* **Проверки:** Аналогичны процедуре редактирования поступления.
* **Действия:** Обновляет запись в таблице Shipments.



**5. Управление товарами**

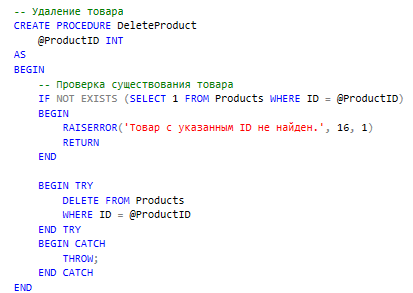
**Создание товара: CreateProduct**

* **Параметры:**@Name, @Article, @Description, @MeasurementUnit, @Price
* **Проверки:** Проверяет заполненность обязательных полей и неотрицательность цены.
* **Действия:** Вставляет запись в таблицу Products.



**Удаление товара: DeleteProduct**

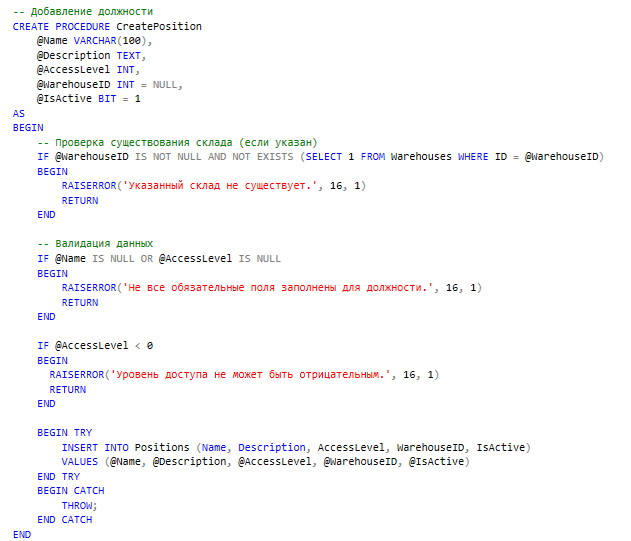
* **Параметры:**
  + @ProductID
* **Проверки:** Проверяет существование товара.
* **Действия:** Удаляет запись из таблицы Products.



**6. Управление должностями**

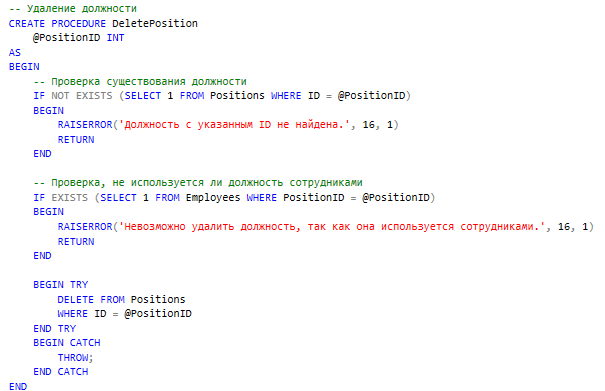
**Создание должности: CreatePosition**

* **Параметры:**
  + @Name, @Description, @AccessLevel, @WarehouseID, @IsActive
* **Проверки:** Проверяет существование склада (если указан) и заполненность обязательных полей.
* **Действия:** Вставляет запись в таблицу Positions.



**Удаление должности: DeletePosition**

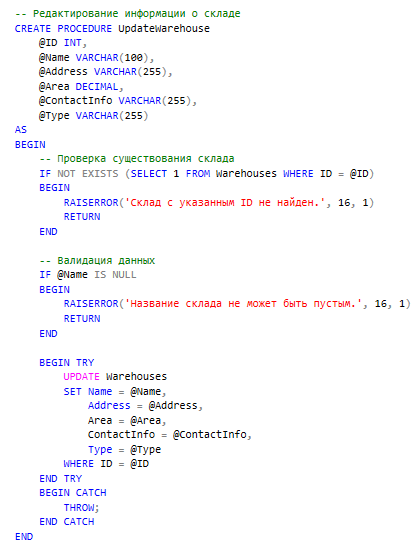
* **Параметры:**
  + @PositionID
* **Проверки:** Проверяет существование должности и её использование сотрудниками.
* **Действия:** Удаляет запись из таблицы Positions.



**7. Управление складами**

**Редактирование склада: UpdateWarehouse**

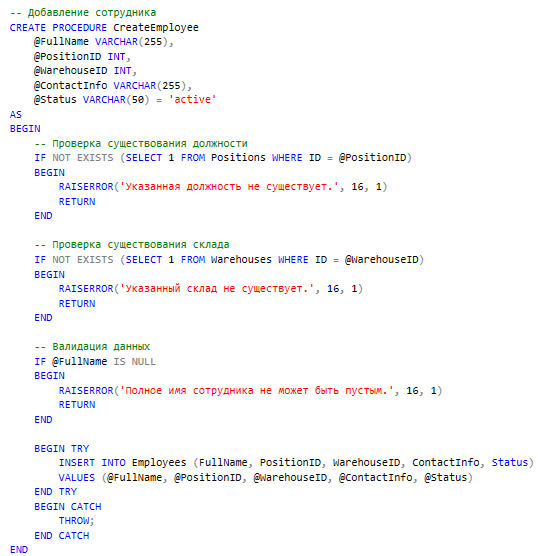
* **Параметры:**
  + @ID, @Name, @Address, @Area, @ContactInfo, @Type
* **Проверки:** Проверяет существование склада и заполненность названия.
* **Действия:** Обновляет запись в таблице Warehouses.



**8. Управление сотрудниками**

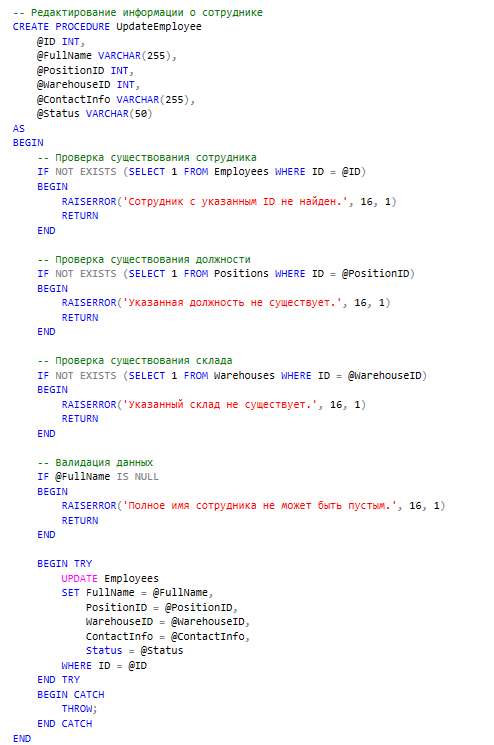
**Создание сотрудника: CreateEmployee**

* **Параметры:**
  + @FullName, @PositionID, @WarehouseID, @ContactInfo, @Status
* **Проверки:** Проверяет существование должности и склада, а также заполненность полного имени.
* **Действия:** Вставляет запись в таблицу Employees.



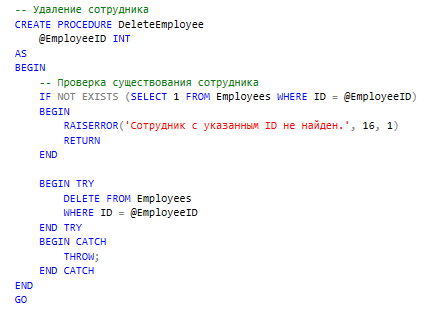
**Редактирование сотрудника: UpdateEmployee**

* **Параметры:**
  + @ID, @FullName, @PositionID, @WarehouseID, @ContactInfo, @Status
* **Проверки:** Аналогичны процедуре создания сотрудника.
* **Действия:** Обновляет запись в таблице Employees.



**Удаление сотрудника: DeleteEmployee**

* **Параметры:**
  + @EmployeeID
* **Проверки:** Проверяет существование сотрудника.
* **Действия:** Удаляет запись из таблицы Employees.



**6. Функции**

### Функция: GetProductInfo

### 

* **Параметры:**
  + @ProductID: ID товара, информацию о котором необходимо получить.
* **Возвращаемое значение:**
  + Возвращает таблицу, содержащую всю информацию о товаре с указанным ProductID.
* **Описание:**  
  Эта функция позволяет пользователю получить полные данные о конкретном товаре, что особенно полезно для просмотра деталей, необходимых для анализа или отчетности.

### Функция: GetShipmentLog

### 

* **Параметры:** Нет.
* **Возвращаемое значение:**
  + Возвращает таблицу с информацией обо всех записях в журнале отгрузок.
* **Описание:**  
  Данная функция предоставляет полный доступ к данным о всех отгрузках, что позволяет анализировать историю отгрузок, выявлять тенденции и контролировать выполнение заказов.

### Функция: GetReceiptLog

### 

* **Параметры:** Нет.
* **Возвращаемое значение:**
  + Возвращает таблицу с информацией обо всех записях в журнале поступлений.
* **Описание:**  
  Эта функция позволяет получить доступ ко всем данным о поступлениях, что важно для учета товаров на складе и анализа поставок.

**7. Триггеры**

### Триггер: TR\_Warehouses\_Area\_Check

### 

* **Событие:** Срабатывает после операций INSERT и UPDATE на таблице Warehouses.
* **Описание:**  
  Триггер предназначен для проверки значения поля Area (площадь) в таблице Warehouses. Если значение площади, которое пытается установить пользователь, меньше нуля, триггер автоматически изменяет его на 0. Это позволяет избежать некорректных значений и поддерживает целостность данных.

**Oracle**

**“Склад”**

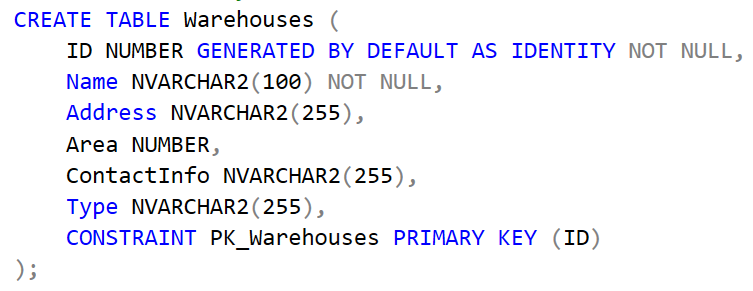
**1. Таблицы**

Для хранения данных созданы следующие таблицы:

* **Warehouses:** Хранит информацию о складах (ID, название, адрес, площадь, контактная информация, тип).
* **Positions:** Хранит информацию о должностях (ID, название, описание, уровень доступа, ID склада, признак активности).
* **Employees:** Хранит информацию о сотрудниках (ID, ФИО, ID должности, ID склада, контактная информация, статус).
* **Products:** Хранит информацию о продуктах (ID, название, артикул, описание, единица измерения, цена).
* **Receipts:** Хранит информацию о поступлениях (ID, дата, поставщик, номер накладной, ID склада, общая сумма).
* **Shipments:** Хранит информацию об отгрузках (ID, дата, клиент, номер накладной, ID склада, общая сумма).

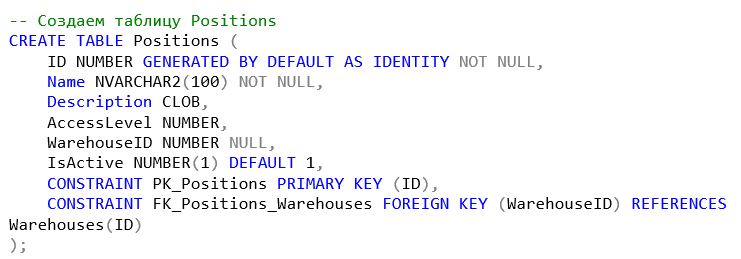
**Таблица Warehouses**

| **Поле** | **Тип данных** | **Описание** | **Ограничения** |
| --- | --- | --- | --- |
| ID | NUMBER | Уникальный идентификатор склада (PRIMARY KEY). | PRIMARY KEY, GENERATED BY DEFAULT AS IDENTITY |
| Name | NVARCHAR(100) | Название склада. | NOT NULL |
| Address | NVARCHAR(255) | Адрес склада. |  |
| Area | NUMBER | Площадь склада. |  |
| ContactInfo | NVARCHAR(255) | Контактная информация. |  |
| Type | NVARCHAR(255) | Тип склада. |  |

****

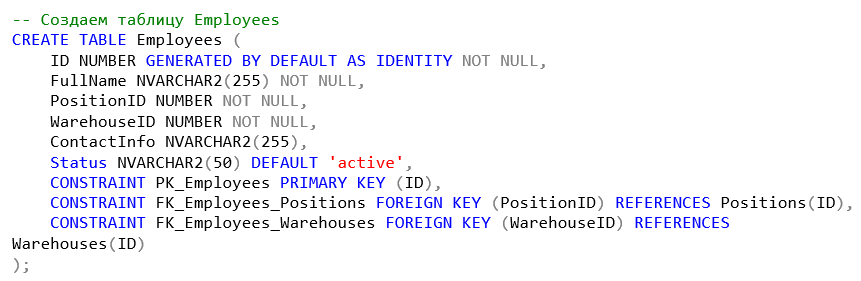
**Таблица Positions**

| **Поле** | **Тип данных** | **Описание** | **Ограничения** |
| --- | --- | --- | --- |
| ID | NUMBER | Уникальный идентификатор должности. | PRIMARY KEY, GENERATED BY DEFAULT AS IDENTITY |
| Name | NVARCHAR(100) | Название должности. | NOT NULL |
| Description | CLOB | Описание должности. |  |
| AccessLevel | NUMBER | Уровень доступа должности. |  |
| WarehouseID | NUMBER | ID склада, к которому относится должность | FOREIGN KEY (WarehouseID) REFERENCES Warehouses(ID) |
| IsActive | NUMBER | Признак активности должности | DEFAULT 1 |

****

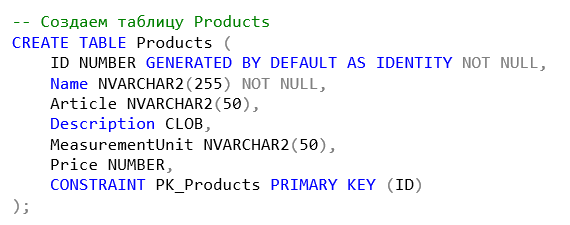
**Таблица Employees**

| **Поле** | **Тип данных** | **Описание** | **Ограничения** |
| --- | --- | --- | --- |
| ID | NUMBER | Уникальный идентификатор сотрудника. | PRIMARY KEY, GENERATED BY DEFAULT AS IDENTITY |
| FullName | NVARCHAR(255) | ФИО сотрудника. | NOT NULL |
| PositionID | NUMBER | ID должности сотрудника. | FOREIGN KEY (PositionID) REFERENCES Positions(ID), NOT NULL |
| WarehouseID | NUMBER | ID склада, к которому относится сотрудник. | FOREIGN KEY (WarehouseID) REFERENCES Warehouses(ID), NOT NULL |
| ContactInfo | NVARCHAR(255) | Контактная информация сотрудника. |  |
| Status | NVARCHAR(50) | Статус сотрудника (например, 'active', 'inactive'). | DEFAULT 'active' |

****

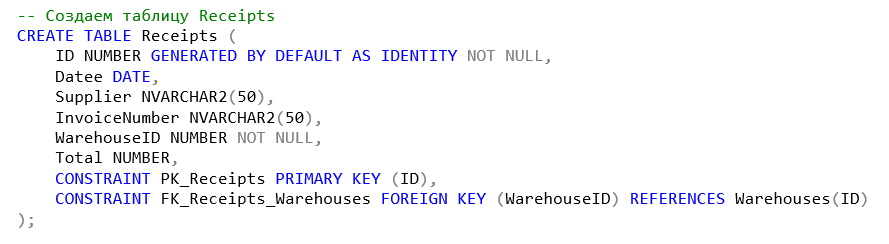
**Таблица Products**

| **Поле** | **Тип данных** | **Описание** | **Ограничения** |
| --- | --- | --- | --- |
| ID | NUMBER | Уникальный идентификатор продукта. | PRIMARY KEY, GENERATED BY DEFAULT AS IDENTITY |
| Name | NVARCHAR(255) | Название продукта. | NOT NULL |
| Article | NVARCHAR(50) | Артикул продукта. |  |
| Description | CLOB | Описание продукта. |  |
| MeasurementUnit | NVARCHAR(50) | Единица измерения продукта. |  |
| Price | NUMBER | Цена продукта. |  |

****

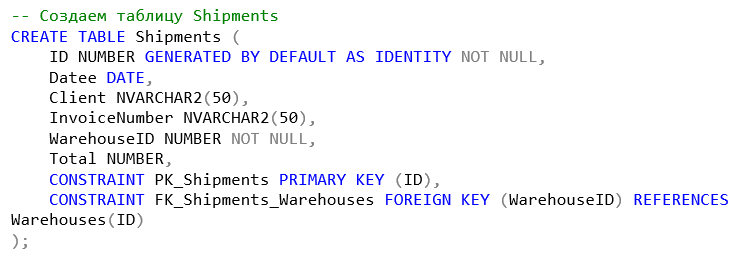
**Таблица Receipts**

| **Поле** | **Тип данных** | **Описание** | **Ограничения** |
| --- | --- | --- | --- |
| ID | NUMBER | Уникальный идентификатор поступления. | PRIMARY KEY, GENERATED BY DEFAULT AS IDENTITY |
| Date | DATE | Дата поступления. |  |
| Supplier | NVARCHAR(50) | Поставщик. |  |
| InvoiceNumber | NVARCHAR(50) | Номер накладной. |  |
| WarehouseID | NUMBER | ID склада, на который поступил товар. | FOREIGN KEY (WarehouseID) REFERENCES Warehouses(ID), NOT NULL |
| Total | NUMBER | Общая сумма поступления. |  |

****

**Таблица Shipments**

| **Поле** | **Тип данных** | **Описание** | **Ограничения** |
| --- | --- | --- | --- |
| ID | NUMBER | Уникальный идентификатор отгрузки. | PRIMARY KEY, GENERATED BY DEFAULT AS IDENTITY |
| Date | DATE | Дата отгрузки. |  |
| Client | NVARCHAR(50) | Клиент. |  |
| InvoiceNumber | NVARCHAR(50) | Номер накладной. |  |
| WarehouseID | NUMBER | ID склада, с которого был отгружен товар. | FOREIGN KEY (WarehouseID) REFERENCES Warehouses(ID), NOT NULL |
| Total | NUMBER | Общая сумма отгрузки. |  |



**2. Индексы**

В рамках оптимизации запросов были созданы следующие индексы:

**1. IX\_Receipts\_WarehouseID (Receipts.WarehouseID):**

* Цель: Быстрый поиск поступлений по складу.
* CREATE INDEX IX\_Receipts\_WarehouseID ON Receipts (WarehouseID);

**2. IX\_Shipments\_WarehouseID (Shipments.WarehouseID):**

* Цель: Быстрый поиск отгрузок по складу.
* CREATE INDEX IX\_Shipments\_WarehouseID ON Shipments (WarehouseID);

**3. IX\_Employees\_PositionID (Employees.PositionID):**

* Цель: Быстрый поиск сотрудников по должности.
* CREATE INDEX IX\_Employees\_PositionID ON Employees (PositionID);

**4. IX\_Employees\_WarehouseID (Employees.WarehouseID):**

* Цель: Быстрый поиск сотрудников по складу.
* CREATE INDEX IX\_Employees\_WarehouseID ON Employees (WarehouseID);

**5. IX\_Positions\_WarehouseID (Positions.WarehouseID):**

* Цель: Быстрый поиск должностей по складу.
* CREATE INDEX IX\_Positions\_WarehouseID ON Positions (WarehouseID);

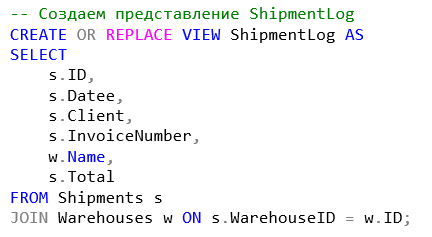
**6. IX\_Products\_Name (Products.Name):**

* Цель: Быстрый поиск продуктов по названию.
* CREATE INDEX IX\_Products\_Name ON Products (Name);

**3. Представления**

Представления – виртуальные таблицы, основанные на SQL-запросах. Они упрощают доступ к данным, скрывают сложность базовых таблиц и повышают безопасность.

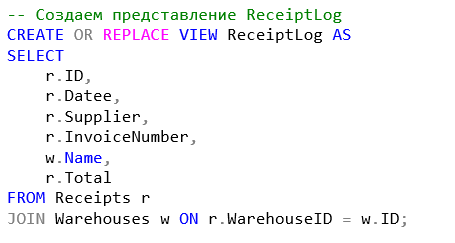
**ShipmentLog (Журнал отгрузок):** Представление объединяет информацию из таблиц Shipments и Warehouses, предоставляя упрощенный вид журнала отгрузок с указанием названия склада.



* + **Столбцы:**
    - ID: ID отгрузки (из таблицы Shipments).
    - Datee: Дата отгрузки (из таблицы Shipments).
    - Client: Клиент (из таблицы Shipments).
    - InvoiceNumber: Номер накладной (из таблицы Shipments).
    - Name: Название склада (из таблицы Warehouses).
    - Total: Общая сумма отгрузки (из таблицы Shipments).
  + **Преимущества:**
    - Упрощает запросы для получения информации об отгрузках, так как не нужно явно указывать JOIN между таблицами Shipments и Warehouses.
    - Обеспечивает единообразный формат данных для отчетов и аналитики.
    - Позволяет скрыть структуру базовых таблиц от пользователей, которым не требуется полный доступ к данным.

**ReceiptLog (Журнал поступлений):** Представление объединяет информацию из таблиц Receipts и Warehouses, предоставляя упрощенный вид журнала поступлений с указанием названия склада.

* + **Столбцы:**
    - ID: ID поступления (из таблицы Receipts).
    - Datee: Дата поступления (из таблицы Receipts).
    - Supplier: Поставщик (из таблицы Receipts).
    - InvoiceNumber: Номер накладной (из таблицы Receipts).
    - Name: Название склада (из таблицы Warehouses).
    - Total: Общая сумма поступления (из таблицы Receipts).
  + **Преимущества:**
    - Аналогично ShipmentLog, упрощает запросы для получения информации о поступлениях.
    - Обеспечивает консистентность данных для отчетов о поступлениях.
    - Повышает безопасность, ограничивая доступ к базовым таблицам.

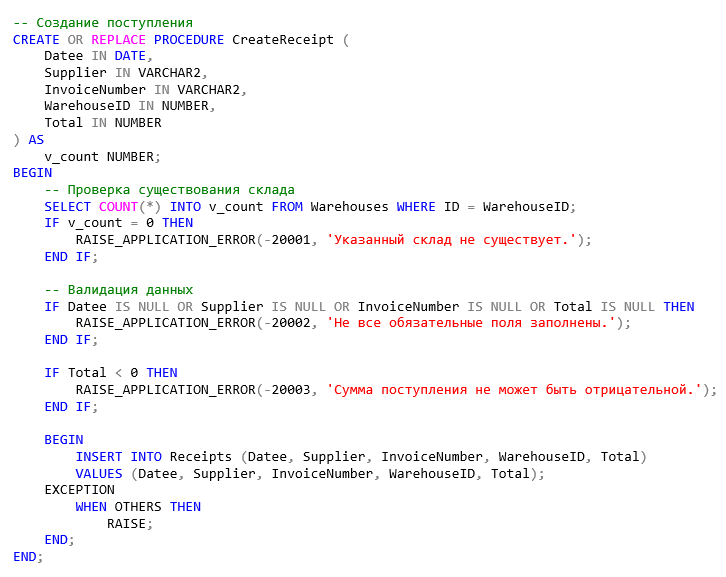


**4. Процедуры**

**1. Создание поступления**

**Процедура: CreateReceipt**

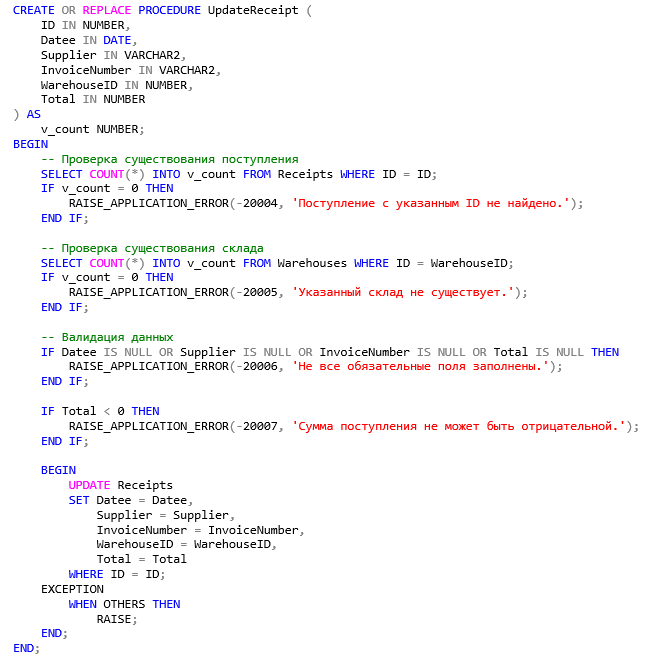
* **Параметры:**
  + Date: дата поступления
  + Supplier: поставщик
  + InvoiceNumber: номер накладной
  + WarehouseID: ID склада
  + Total: общая сумма
* **Проверки:**
  + Проверяет существование склада.
  + Проверяет заполненность обязательных полей.
  + Проверяет, что сумма не отрицательная.
* **Действия:** Вставляет запись в таблицу Receipts.



**2. Редактирование поступления**

**Процедура: UpdateReceipt**

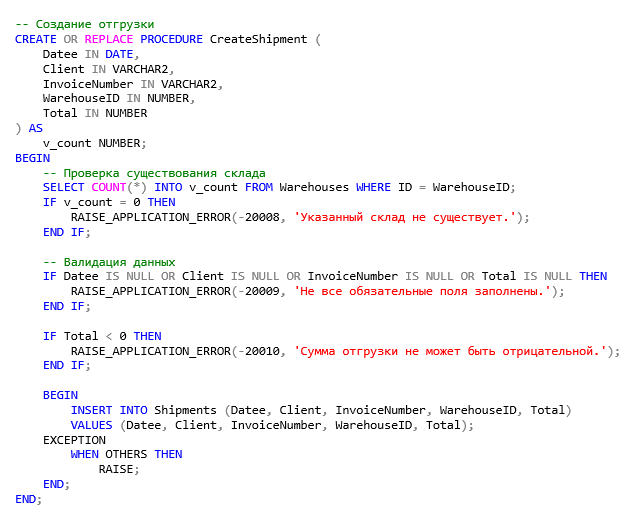
* **Параметры:**
  + ID: ID поступления
  + Date, Supplier, InvoiceNumber, WarehouseID, Total: обновляемые значения
* **Проверки:** Аналогичны процедуре создания.
* **Действия:** Обновляет запись в таблице Receipts.



**3. Создание отгрузки**

**Процедура: CreateShipment**

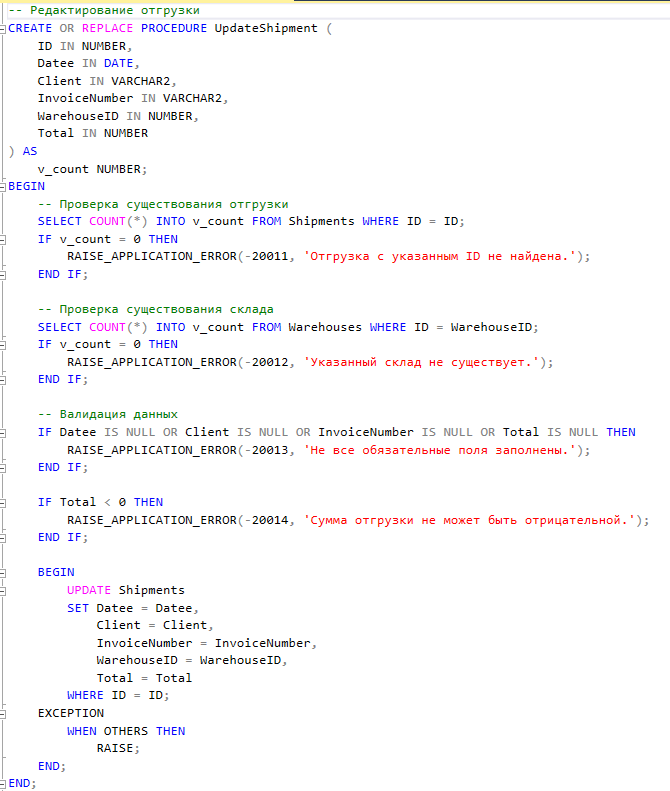
* **Параметры:**
  + Date, Client, InvoiceNumber, WarehouseID, Total
* **Проверки:** Аналогичны процедуре создания поступления.
* **Действия:** Вставляет запись в таблицу Shipments.



**4. Редактирование отгрузки**

**Процедура: UpdateShipment**

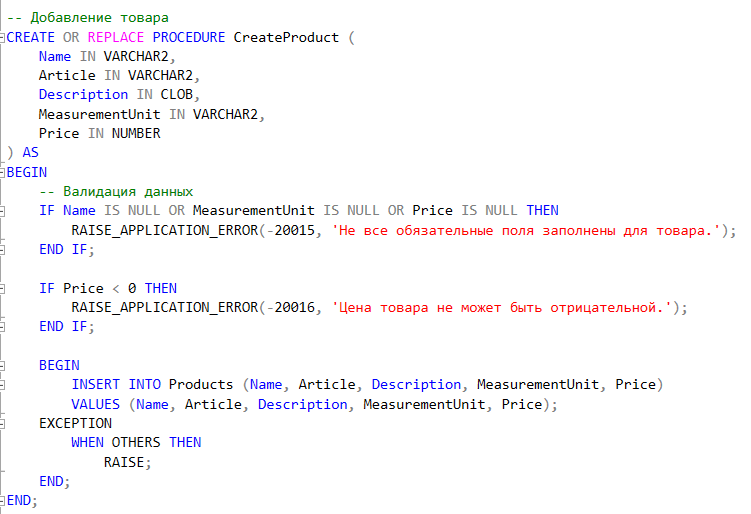
* **Параметры:**
  + ID, Date, Client, InvoiceNumber, WarehouseID, Total
* **Проверки:** Аналогичны процедуре редактирования поступления.
* **Действия:** Обновляет запись в таблице Shipments.



**5. Управление товарами**

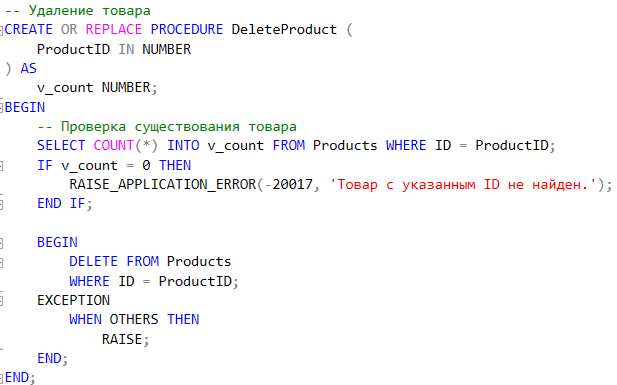
**Создание товара: CreateProduct**

* **Параметры:** Name, Article, Description, MeasurementUnit, Price
* **Проверки:** Проверяет заполненность обязательных полей и неотрицательность цены.
* **Действия:** Вставляет запись в таблицу Products.



**Удаление товара: DeleteProduct**

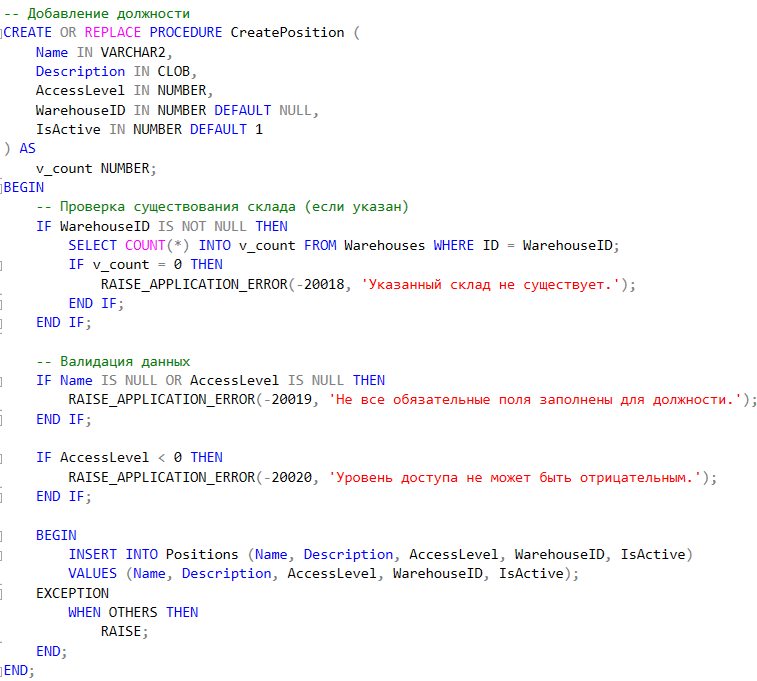
* **Параметры:**
  + ProductID
* **Проверки:** Проверяет существование товара.
* **Действия:** Удаляет запись из таблицы Products.



**6. Управление должностями**

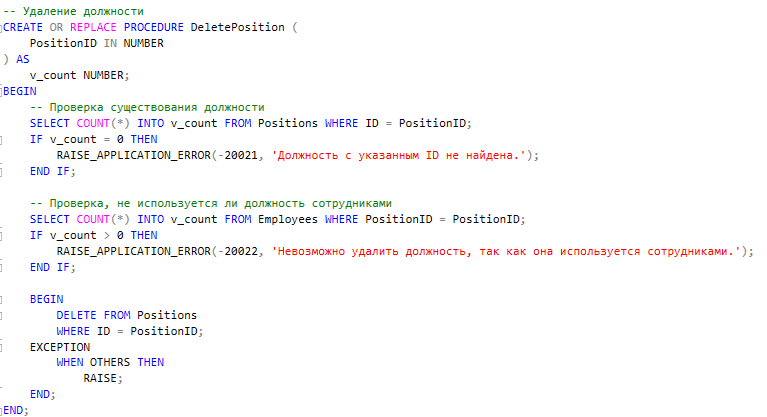
**Создание должности: CreatePosition**

* **Параметры:**
  + Name, Description, AccessLevel, WarehouseID, IsActive
* **Проверки:** Проверяет существование склада (если указан) и заполненность обязательных полей.
* **Действия:** Вставляет запись в таблицу Positions.



**Удаление должности: DeletePosition**

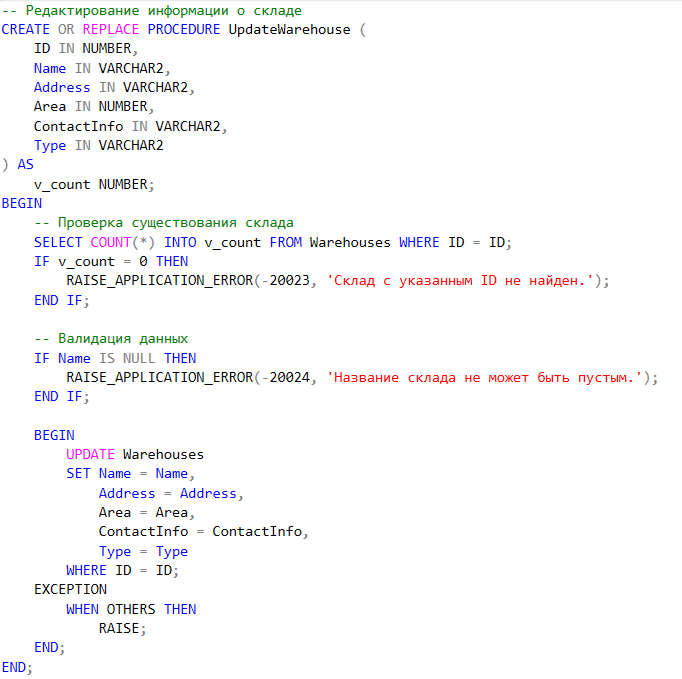
* **Параметры:**
  + PositionID
* **Проверки:** Проверяет существование должности и её использование сотрудниками.
* **Действия:** Удаляет запись из таблицы Positions.



**7. Управление складами**

**Редактирование склада: UpdateWarehouse**

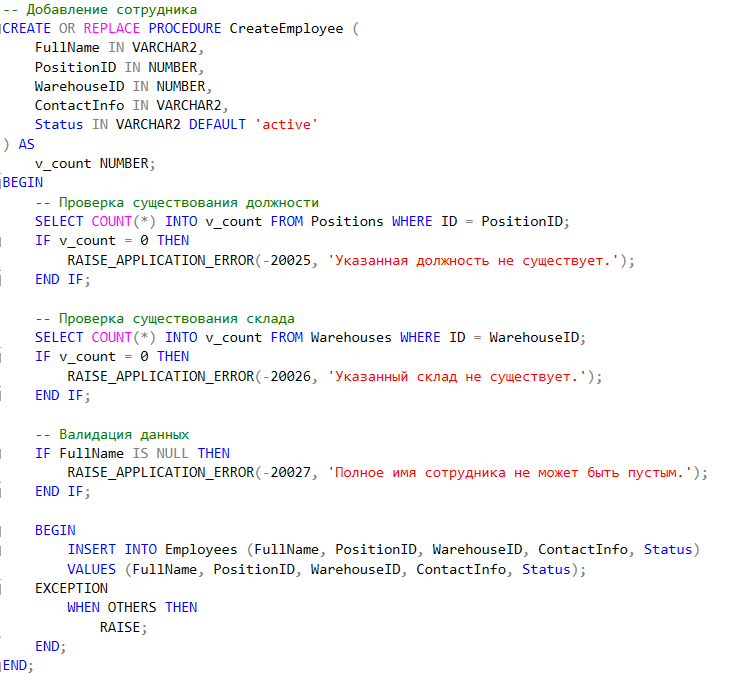
* **Параметры:**
  + ID, Name, Address, Area, ContactInfo, Type
* **Проверки:** Проверяет существование склада и заполненность названия.
* **Действия:** Обновляет запись в таблице Warehouses.



**8. Управление сотрудниками**

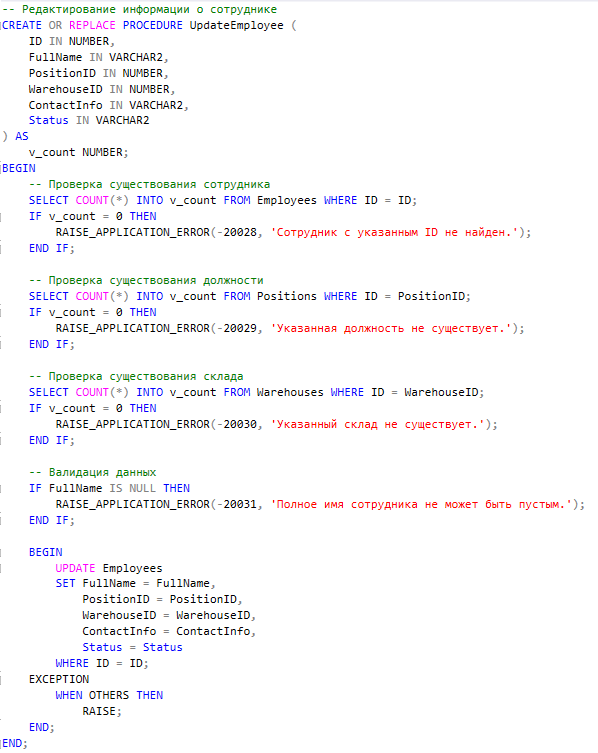
**Создание сотрудника: CreateEmployee**

* **Параметры:**
  + FullName, PositionID, WarehouseID, ContactInfo, Status
* **Проверки:** Проверяет существование должности и склада, а также заполненность полного имени.
* **Действия:** Вставляет запись в таблицу Employees.



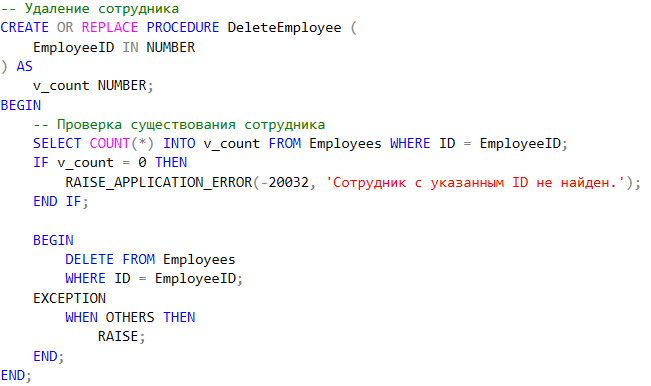
**Редактирование сотрудника: UpdateEmployee**

* **Параметры:**
  + ID, FullName, PositionID, WarehouseID, ContactInfo, Status
* **Проверки:** Аналогичны процедуре создания сотрудника.
* **Действия:** Обновляет запись в таблице Employees.



**Удаление сотрудника: DeleteEmployee**

* **Параметры:**
  + EmployeeID
* **Проверки:** Проверяет существование сотрудника.
* **Действия:** Удаляет запись из таблицы Employees.



**5. Функции**

### Функция: GetProductInfo

### 

* **Параметры:**
  + ProductID: ID товара, информацию о котором необходимо получить.
* **Возвращаемое значение:**
  + Возвращает таблицу, содержащую всю информацию о товаре с указанным ProductID.
* **Описание:**  
  Эта функция позволяет пользователю получить полные данные о конкретном товаре, что особенно полезно для просмотра деталей, необходимых для анализа или отчетности.

### Функция: GetShipmentLog

### 

* **Параметры:** Нет.
* **Возвращаемое значение:**
  + Возвращает таблицу с информацией обо всех записях в журнале отгрузок.
* **Описание:**  
  Данная функция предоставляет полный доступ к данным о всех отгрузках, что позволяет анализировать историю отгрузок, выявлять тенденции и контролировать выполнение заказов.

### Функция: GetReceiptLog

### 

* **Параметры:** Нет.
* **Возвращаемое значение:**
  + Возвращает таблицу с информацией обо всех записях в журнале поступлений.
* **Описание:**  
  Эта функция позволяет получить доступ ко всем данным о поступлениях, что важно для учета товаров на складе и анализа поставок.

**6. Триггеры**

### Триггер: TR\_Warehouses\_Area\_Check

### 

* **Событие:** Срабатывает после операций INSERT и UPDATE на таблице Warehouses.
* **Описание:**  
  Триггер предназначен для проверки значения поля Area (площадь) в таблице Warehouses. Если значение площади, которое пытается установить пользователь, меньше нуля, триггер автоматически изменяет его на 0. Это позволяет избежать некорректных значений и поддерживает целостность данных.