ClickHouse Homework. UDF, Aggregate Functions and working with data types

Вариант 1:

Цель:

Цель этого домашнего задания - помочь вам понять и применить агрегатные функции, функции, работающие с типами данных, и функции, определяемые пользователем (UDF) в ClickHouse.

Задачи:

Вам предстоит выполнить следующие задания:

1. Используйте агрегатные функции для обобщения данных.
2. Применять функции, работающие с различными типами данных.
3. Создавать и использовать функции, определяемые пользователем (UDF), в ClickHouse.

Набор данных:

Для выполнения этого домашнего задания вы будете использовать пример набора данных, представляющего транзакции электронной коммерции. Предположим, что у вас есть таблица `transactions` со следующей схемой:

```
CREATE TABLE transactions (
    transaction_id UInt32,
    user_id UInt32,
    product_id UInt32,
    quantity UInt8,
    price Float32,
    transaction_date Date
) ENGINE = MergeTree()
ORDER BY (transaction_id);
```

Задание:

1. **Агрегатные функции**
    1.1. Рассчитайте общий доход от всех операций.
    1.2. Найдите средний доход с одной сделки.

1.3. Определите общее количество проданной продукции.
1.4. Подсчитайте количество уникальных пользователей, совершивших покупку.

2. Функции для работы с типами данных

2.1. Преобразуйте `transaction_date` в **строку** формата `YYYY-MM-DD`.
2.2. Извлеките год и месяц из `transaction_date`.
2.3. Округлите `price` до ближайшего целого числа.
2.4. Преобразуйте `transaction_id` в строку.

3. User-Defined Functions (UDFs)

3.1. Создайте простую UDF для расчета общей стоимости транзакции.
3.2. Используйте созданную UDF для расчета общей цены для каждой транзакции.
3.3. Создайте UDF для классификации транзакций на «высокоценные» и «малоценные» на основе порогового значения (например, 100).
3.4. Примените UDF для категоризации каждой транзакции.

Результат работы:

- Приведите SQL-запросы и их результаты для каждой задачи.
- Представьте свою работу в виде pdf файла или ссылки на гит-репозиторий.

Критерии оценки:

- Корректность и эффективность SQL-запросов.
- Правильное использование агрегатных функций, функций для работы с типами данных и UDF.

Вариант 2:

Цель:

Цель этого домашнего задания - помочь вам понять и применить исполняемые пользовательские функции (EUDF) в ClickHouse. EUDF позволяют расширить функциональность ClickHouse путем написания пользовательских функций на внешних языках программирования, таких как Python.

Задачи:

Вы выполните следующие задачи:

1. Настроить среду для использования EUDF.
2. Создайте и зарегистрируйте EUDF в ClickHouse.
3. Используйте EUDF для выполнения пользовательских преобразований данных и вычислений.

Набор данных:

Для выполнения этого домашнего задания вы будете использовать пример набора данных, представляющего транзакции электронной коммерции. Предположим, что у вас есть таблица `transactions` со следующей схемой:

```
CREATE TABLE transactions (
    transaction_id UInt32,
    user_id UInt32,
    product_id UInt32,
    quantity UInt8,
    price Float32,
    transaction_date Date
) ENGINE = MergeTree()
ORDER BY (transaction_id);
```

Задание:

1. **Настройка среды для EUDF**
    1.1. Установка необходимого программного обеспечения. Убедитесь, что у вас установлен Python и необходимые библиотеки. Используйте следующие команды для настройки среды:

```
sudo apt-get install python3 python3-pip
pip3 install clickhouse-driver
```

1.2. Настройте ClickHouse для EUDF. Убедитесь, что ClickHouse настроен на разрешение EUDF. Измените конфигурационный файл ClickHouse (обычно находится по адресу `/etc/clickhouse-server/config.xml`), чтобы включить следующие настройки:

```
<clickhouse>
    <user_defined_executable_functions_config>
        <allow_functions>true</allow_functions>
        <execution_path>/path/to/your/udf/script</execution_path>
    </user_defined_executable_functions_config>
</clickhouse>
```

1.3. Создание каталога для сценариев EUDF

```
mkdir /path/to/your/udf
```

2. Создание и применение EUDF

2.1. Создайте простой скрипт Python UDF. Напишите сценарий Python для расчета общей цены транзакции. Сохраните этот скрипт под именем `total_price.py` в вашей директории EUDF:

```
import sys
import json

def total_price(quantity, price):
    return quantity * price

if __name__ == "__main__":
    data = json.load(sys.stdin)
    quantity = data['quantity']
    price = data['price']
    print(total_price(quantity, price))
```

2.2. Применение EUDF в ClickHouse. Используйте следующую команду SQL для регистрации EUDF:

```
CREATE FUNCTION total_price AS
 '/path/to/your/udf/total_price.py'
 RETURNS Float32
 EXECUTE ON HOST;
```

3. Использование EUDF:

3.1. Рассчитайте общую цену для каждой транзакции (используя `total_price`):

```
SELECT
    transaction_id,
    total_price(quantity, price) AS total_price
FROM transactions
LIMIT 10;
```

3.2. Создайте более сложный Python UDF скрипт. Напишите Python-скрипт для классификации транзакций на 'High Value' и 'Low Value' на основе порогового значения. Сохраните этот скрипт под именем `transaction_category.py`:

```
import sys
import json

def transaction_category(total_price, threshold=100):
    if total_price > threshold:
        return 'High Value'
    else:
        return 'Low Value'

if __name__ == "__main__":
    data = json.load(sys.stdin)
    total_price = data['total_price']
    threshold = data.get('threshold', 100)
    print(transaction_category(total_price, threshold))
```

3.3. Примените новый EUDF в ClickHouse:

```
CREATE FUNCTION transaction_category AS
'/path/to/your/udf/transaction_category.py'
RETURNS String
EXECUTE ON HOST;
```

3.4. Категоризируйте каждую транзакцию, используя `transaction_category`:

```
WITH (quantity * price) AS total_price
SELECT
    transaction_id,
    total_price,
    transaction_category(total_price) AS category
FROM transactions
LIMIT 10;
```

Результат работы:

- Предоставьте SQL-запросы и их результаты для каждой задачи.
- Включите скрипты Python, используемые для EUDF.
- Пришлите работу в виде pdf файла, блокнота Jupyter или гит-репозитория, содержащей скрипты и файл README.

Критерии оценки:

- Корректность и эффективность SQL-запросов.
- Правильная настройка и конфигурация для использования EUDF.
- Функциональные и правильно реализованные скрипты EUDF.