ДЗ_Функции_для_работы_с_типами_данных_агрегатные_функции_и_UDF

Первый способ

CREATE TABLE transactions (

   transaction_id **UInt32** COMMENT 'идентификатор транзакции_id',

   user_id **UInt32** COMMENT 'идентификатор пользователя',

   product_id **UInt32** COMMENT 'идентификатор продукта_id',

   quantity **UInt8** COMMENT 'количество',

   price **Float32** COMMENT 'цена',

   transaction_date **Date** COMMENT 'дата транзакции_date'

) **ENGINE** = MergeTree()

ORDER BY (transaction_id);

--Агрегатные функции

Select

sum("quantity" * "price") **as** *"общий доход от всех операций"*

,**AVG**("quantity" * "price") **as** *"средний доход с одной сделки"*

,**sum**("quantity") **as** *"общее количество проданной продукции"*

,**uniqExact**("user_id") **as** *"количество уникальных пользователей, совершивших покупку"*

from "default"."transactions"

where transaction_date **is not null**

--Функции для работы с типами данных

Select

toYYYYMM("transaction_date")

,**toYear**("transaction_date")

,**toMonth**("transaction_date")

,**round**(price, 1)

,**toString**("transaction_id")

from "default"."transactions"

--User-Defined Functions (UDFs)

CREATE FUNCTION total_price **AS** (q, p) -> q * p;

SELECT

        transaction_id,

        total_price(quantity, price) **AS** *total*

FROM "default"."transactions"

CREATE FUNCTION tx_category **AS** (v, thr) -> **if**(v > thr, 'High Value', 'Low Value');

WITH *total_price*(*quantity*, *price*) **AS** v

SELECT

        transaction_id,

        v **AS** *total*,

          tx_category(v, 100) **AS** *category*

FROM "default"."transactions"

SELECT

toTypeName(**total_price**(quantity, price)) **AS** *total_type*,

toTypeName(**tx_category**(123, 100)) **AS** *cat_type*

FROM "default"."transactions"

Второй способ

PS C:\Users\doner> wsl

chuda@DESKTOP-5MDI75J:/mnt/c/Users/doner$ sudo mkdir -p /var/lib/clickhouse/user_scripts && sudo chown -R clickhouse:clickhouse /var/lib/clickhouse/user_scripts

[sudo] password for chuda:

chuda@DESKTOP-5MDI75J:/mnt/c/Users/doner$ sudo tee /var/lib/clickhouse/user_scripts/total_price.py >/dev/null <<'PY'\n#!/usr/bin/env python3\nimport sys, json\nfor line in sys.stdin:\n o = json.loads(line)\n q = o.get('quantity') or 0\n p = o.get('price') or 0.0\n print(q * p)\nPY

> #!/usr/bin/env python3

import sys, json

for line in sys.stdin:

o = json.loads(line)

q = o.get('quantity') or 0

p = o.get('price') or 0.0

print(q * p)

PY

> PY

> PY

> ^C

chuda@DESKTOP-5MDI75J:/mnt/c/Users/doner$ sudo tee /var/lib/clickhouse/user_scripts/total_price.py >/dev/null <<'PY'

```
#!/usr/bin/env python3

import sys, json

for line in sys.stdin:

o = json.loads(line)

q = o.get('quantity') or 0

p = o.get('price') or 0.0

print(q * p)

PY.
```

> printf '%s\n' '#!/usr/bin/env python3' 'import sys, json' 'for line in sys.stdin:' ' o = json.loads(line)' " q = o.get('quantity') or 0" " p = o.get('price') or 0.0" ' print(q * p)' | sudo tee /var/lib/clickhouse/user_scripts/total_price.py >/dev/null

> ^C

chuda@DESKTOP-5MDI75J:/mnt/c/Users/doner$ printf '%s\n' '#!/usr/bin/env python3' 'import sys, json' 'for line in sys.stdin:' ' o = json.loads(line)' " q = o.get('quantity') or 0" " p = o.get('price') or 0.0" ' print(q * p)' | sudo tee /var/lib/clickhouse/user_scripts/total_price.py >/dev/null

[sudo] password for chuda:

Sorry, try again.

[sudo] password for chuda:

chuda@DESKTOP-5MDI75J:/mnt/c/Users/doner$ sudo wc -l /var/lib/clickhouse/user_scripts/total_price.py && sudo sed -n '1,20p' /var/lib/clickhouse/user_scripts/total_price.py

7 /var/lib/clickhouse/user_scripts/total_price.py

```
#!/usr/bin/env python3

import sys, json

for line in sys.stdin:

 o = json.loads(line)

 q = o.get('quantity') or 0

 p = o.get('price') or 0.0

 print(q * p)
```

chuda@DESKTOP-5MDI75J:/mnt/c/Users/doner$ sudo chmod 755 /var/lib/clickhouse/user_scripts/total_price.py && sudo chown clickhouse:clickhouse /var/lib/clickhouse/user_scripts/total_price.py

chuda@DESKTOP-5MDI75J:/mnt/c/Users/doner$ printf '%s\n' '#!/usr/bin/env python3' 'import sys, json' 'for line in sys.stdin:' ' o = json.loads(line)' " v = o.get('total_price', 0.0)" " thr = o.get('threshold', 100)" " print('High Value' if v > thr else 'Low Value')" | sudo tee /var/lib/clickhouse/user_scripts/transaction_category.py >/dev/null

chuda@DESKTOP-5MDI75J:/mnt/c/Users/doner$ sudo chmod 755 /var/lib/clickhouse/user_scripts/transaction_category.py && sudo chown clickhouse:clickhouse /var/lib/clickhouse/user_scripts/transaction_category.py

chuda@DESKTOP-5MDI75J:/mnt/c/Users/doner$

--XML--

chuda@DESKTOP-5MDI75J:/mnt/c/Users/doner$ sudo tee /etc/clickhouse-server/udf/total_price_e.xml >/dev/null <<'XML'

```
<functions>

<function>

<name>total_price_e</name>

<type>executable</type>

<return_type>Float32</return_type>

<argument><type>UInt8</type><name>quantity</name></argument>

<argument><type>Float32</type><name>price</name></argument>

<format>JSONEachRow</format>

<command>total_price.py</command>

<deterministic>true</deterministic>

</function>

</functions>

XML
```

chuda@DESKTOP-5MDI75J:/mnt/c/Users/doner$ sudo tee /etc/clickhouse-server/udf/transaction_category_e.xml >/dev/null <<'XML'

```
<functions>

<function>

<name>transaction_category_e</name>

<type>executable</type>

<return_type>String</return_type>

<argument><type>Float32</type><name>total_price</name></argument>

<argument><type>UInt16</type><name>threshold</name></argument>

<format>JSONEachRow</format>

<command>transaction_category.py</command>

<deterministic>true</deterministic>

</function>
```

</functions>

XML

--Проверка—

DESKTOP-5MDI75J.localdomain :) SELECT name FROM system.functions WHERE name IN ('total_price_e','transaction_category_e')

SELECT name

FROM system.functions

WHERE name IN ('total_price_e', 'transaction_category_e')

Query id: 0f3617a8-8346-4072-b600-5a1f8387b4cb

```
   ┌─name──────────────────┐
1. │ transaction_category_e │
2. │ total_price_e          │
   └───────────────────────┘
```

2 rows in set. Elapsed: 0.006 sec. Processed 1.67 thousand rows, 583.70 KB (268.46 thousand rows/s., 93.83 MB/s.)

Peak memory usage: 0.00 B.

DESKTOP-5MDI75J.localdomain :) SELECT transaction_id, total_price_e(quantity, price) AS total FROM default.transactions ORDER BY transaction_id

SELECT

    transaction_id,

    total_price_e(quantity, price) AS total

FROM default.transactions

ORDER BY transaction_id ASC

Query id: af00a7aa-3e8b-41e2-b460-e483260f76d7

Ok.

0 rows in set. Elapsed: 0.003 sec.

DESKTOP-5MDI75J.localdomain :) WITH total_price_e(quantity, price) AS v

SELECT transaction_id, v AS total, transaction_category_e(v, 100) AS category

FROM default.transactions ORDER BY transaction_id

WITH total_price_e(quantity, price) AS v

SELECT

   transaction_id,

   v AS total,

   transaction_category_e(v, 100) AS category

FROM default.transactions

ORDER BY transaction_id ASC

Query id: 45b17879-9839-4e2d-a79e-0ddcdc9c94e9

Ok.

0 rows in set. Elapsed: 0.003 sec.