**Analiza kont**

Napisz kwerendę, która uszereguje konta według następujących kryteriów:

* liczba udzielonych pożyczek (malejąco),
* kwota udzielonych pożyczek (malejąco),
* średnia kwota pożyczki.

Pod uwagę bierzemy tylko **spłacone** pożyczki.

**Rozwiązanie**

Na podstawie warunków zadania wystarczy, że użyjemy tabeli **loan**, jednak mamy tutaj dwie składowe:

1. grupowanie wierszy według **account\_id**,
2. rankowanie wyników poprzedniego zapytania.

Czyli zadanie trzeba wykonać w dwóch krokach. Przykładowo użyjemy podzapytania, a następnie dodamy funkcję okna, aby otrzymać wynik.

Podzapytanie

Rozwiązanie zadania rozpoczniemy od stworzenia kwerendy do podzapytania. Jako pierwszy krok zaczniemy od odpowiedniej filtracji wierszy w tabeli **loan**:

1

**SELECT** \*

2

**FROM** financial.loan

3

**WHERE** status **IN** ('A', 'C') -- tylko udzielone pożyczki

4

;

Gdzie z poprzedniego zadania wiemy, że statusy **('A', 'C')** definiują razem dziedzinę pożyczek spłaconych. Teraz przejdziemy do grupowania według kont, czyli **account\_id**:

1

**SELECT**

2

account\_id

3

**FROM** financial.loan

4

**WHERE** status **IN** ('A', 'C') -- tylko udzielone pożyczki

5

;

Pozostaje nam już tylko dla danego konta wyznaczyć odpowiednie statystyki:

* sumę spłaconych pożyczek, czyli **sum(amount)**,
* liczbę spłaconych pożyczek, czyli **count(amount)**,
* średnią kwotę spłaconych pożyczek, czyli **avg(amount)**.

1

**SELECT**

2

account\_id,

3

sum(amount) **as** loans\_amount,

4

count(amount) **as** loans\_count,

5

avg(amount) **as** loans\_avg

6

**FROM** financial.loan

7

**WHERE** status **IN** ('A', 'C') -- tylko udzielone pożyczki

8

**GROUP** **BY** account\_id

Zostało nam teraz tylko opakować kwerendę w podzapytanie:

1

**WITH** cte **as** (

2

**SELECT**

3

account\_id,

4

sum(amount) **as** loans\_amount,

5

count(amount) **as** loans\_count,

6

avg(amount) **as** loans\_avg

7

**FROM** financial.loan

8

**WHERE** status **IN** ('A', 'C') -- tylko udzielone pożyczki

9

**GROUP** **BY** account\_id

10

)

11

**SELECT** \*

12

**FROM** cte

Tak stworzone podzapytanie użyjemy w celu oznaczenia rankowania.

Rankowanie

Przechodzimy do rankowania kont według zadanych kryteriów. W tym celu użyjemy **ROW\_NUMBER** oraz wcześniej stworzonego podzapytania, do którego dodamy:

* **ROW\_NUMBER() OVER(ORDER BY loans\_amount DESC)** - wynik tej formuły będzie rankował **wszystkie** **account\_id** według kwoty zawartej pożyczki od najwyższej do najniższej,
* **ROW\_NUMBER() OVER(ORDER BY loans\_count DESC)** - wynik tej formuły będzie rankował **wszystkie** **account\_id** według liczby zawartych pożyczek od najwyższej do najniższej.

Ostatecznie po modyfikacjach otrzymujemy następujące zapytanie:

1

**WITH** cte **AS** (

2

-- pierwszy krok, czyli zagregowanie danych do poziomu account\_id

3

**SELECT**

4

account\_id,

5

sum(amount) **as** loans\_amount,

6

count(amount) **as** loans\_count,

7

avg(amount) **as** loans\_avg

8

**FROM** financial.loan

9

**WHERE** status **IN** ('A', 'C') -- tylko udzielone pożyczki

10

**GROUP** **BY** account\_id

11

)

12

**SELECT**

13

\*,

14

-- rankowanie

15

ROW\_NUMBER() **over** (**ORDER** **BY** loans\_amount **DESC**) **AS** rank\_loans\_amount,

16

ROW\_NUMBER() **over** (**ORDER** **BY** loans\_count **DESC**) **AS** rank\_loans\_count

17

**FROM** cte

Oczywiście, równie dobrze można użyć innej funkcji rankującej, jak **RANK** czy **DENSE\_RANK**, niemniej z technicznego punktu widzenia **ROW\_NUMBER** też rozwiązuje zadanie.