**Spłacone pożyczki**

Sprawdź, ile pożyczek zostało spłaconych w podziale na płeć klienta.

Dodatkowo w wybrany przez siebie sposób sprawdź, czy kwerenda jest poprawna.

**Rozwiązanie**

Zadanie wygląda niepozornie, jednak aby uzyskać odpowiedź na to pytanie, należy dokonać kilku **JOIN**ów, wychodząc od **loan**, a aby otrzymać informację o płci, należy przejść przez **account** -> **disp** -> **client**.

Selekcja dziedziny kwerendy

Ponieważ w warunkach zadania mamy informację o tym, że interesują nas tylko pożyczki spłacone, na samym początku zaczniemy od wybrania tylko tych wierszy, które nas interesują.

Z rozwiązania zadania o statusach pożyczek, wiemy, że pożyczki spłacone to te, które zostały oznaczone statusami **('A', 'C')**:

1

**SELECT** \*

2

**FROM** financial.loan **as** l

3

**WHERE** l.status **IN** ('A', 'C')

Przechodzenie przez schemat bazy

W pierwszej części zajmiemy się wyłącznie napisaniem odpowiednich warunków złączeń.

Złączenie **loan** oraz **account** - tutaj nie ma większego problemu, można to zrobić bezpośrednio używając kolumny **account\_id**:

1

**SELECT** \*

2

**FROM**

3

financial.loan **as** l

4

**INNER** **JOIN**

5

financial.account **as** a **USING** (account\_id)

6

**WHERE** l.status **IN** ('A', 'C')

Następnie, do dołączenia do aktualnego stanu, mamy tabelę **disp**. To ona zawiera kolumnę **client\_id**, której użyjemy później do połączenia się z tabelą **client\_id**.

Złączenie z aktualnym wynikiem odbywa się poprzez **account\_id**, z tego powodu ponownie użyjemy klauzuli **USING**:

1

**SELECT** \*

2

**FROM**

3

financial.loan **as** l

4

**INNER** **JOIN**

5

financial.account **as** a **USING** (account\_id)

6

**INNER** **JOIN**

7

financial.disp **as** d **USING** (account\_id)

8

**WHERE** l.status **IN** ('A', 'C')

Aby otrzymać informację o płci klienta, potrzebna jest nam tabela **client**. Tutaj złączenie odbywa się poprzez kolumnę **client\_id**, która pochodzi z poprzednio dołączonej tabeli **disp**:

1

**SELECT**

2

\*

3

**FROM**

4

financial.loan **as** l

5

**INNER** **JOIN**

6

financial.account a **using** (account\_id)

7

**INNER** **JOIN**

8

financial.disp **as** d **using** (account\_id)

9

**INNER** **JOIN**

10

financial.client **as** c **using** (client\_id)

11

**WHERE** l.status **IN** ('A', 'C')

Grupowanie

W tym momencie mamy do dyspozycji wszystko, co trzeba, wystarczy odpowiednio pogrupować według kolumny **gender** oraz zsumować kolumnę **amount**:

1

**SELECT**

2

c.gender,

3

**sum**(l.amount) **as** amount

4

**FROM**

5

financial.loan **as** l

6

**INNER** **JOIN**

7

financial.account a **using** (account\_id)

8

**INNER** **JOIN**

9

financial.disp **as** d **using** (account\_id)

10

**INNER** **JOIN**

11

financial.client **as** c **using** (client\_id)

12

**WHERE** l.status **IN** ('A', 'C')

13

**GROUP** **BY** c.gender

Sprawdzenie

Zapiszmy wyniki do tabeli tymczasowej, aby zwalidować poprawność rozwiązania:

1

**DROP** **TABLE** **IF** **EXISTS** tmp\_results;

2

**CREATE** **TEMPORARY** **TABLE** tmp\_results **AS**

3

**SELECT**

4

c.gender,

5

sum(l.amount) **as** amount

6

**FROM**

7

financial.loan **as** l

8

**INNER** **JOIN**

9

financial.account a **using** (account\_id)

10

**INNER** **JOIN**

11

financial.disp **as** d **using** (account\_id)

12

**INNER** **JOIN**

13

financial.client **as** c **using** (client\_id)

14

**WHERE** l.status **IN** ('A', 'C')

15

**GROUP** **BY** c.gender

Możemy teraz sprawdzić czy sumy pomiędzy tabelami się zgadzają:

1

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

2

**SELECT** sum(amount) **as** amount

3

**FROM** financial.loan **as** l

4

**WHERE** l.status **IN** ('A', 'C')

5

)

6

**SELECT** (**SELECT** SUM(amount) **FROM** tmp\_results) - (**SELECT** amount **FROM** cte)

Wykonując powyższą kwerendę zauważymy, że różnica jest różna od 0, a co za tym idzie, kwota udzielonych pożyczek ulegała *powieleniu*, zatem jest błąd.

Pułapka tkwi w łączeniu pomiędzy **account** oraz **disp**, gdzie dla jednego **account\_id** może być wielu klientów (właściciel i dysponent). Aby to poprawić, należy zmodyfikować kwerendę na przykład następująco:

1

**DROP** **TABLE** **IF** **EXISTS** tmp\_results;

2

**CREATE** **TEMPORARY** **TABLE** tmp\_results **AS**

3

**SELECT**

4

c.gender,

5

sum(l.amount) **as** amount

6

**FROM**

7

financial.loan **as** l

8

**INNER** **JOIN**

9

financial.account a **using** (account\_id)

10

**INNER** **JOIN**

11

financial.disp **as** d **using** (account\_id)

12

**INNER** **JOIN**

13

financial.client **as** c **using** (client\_id)

14

**WHERE** **True**

15

**AND** l.status **IN** ('A', 'C')

16

**AND** d.**type** = 'OWNER'

17

**GROUP** **BY** c.gender

Teraz sprawdzenie - można sprawdzić czy coś nam po drodze *nie uciekło* albo nie przybyło:

1

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

2

**SELECT** sum(amount) **as** amount

3

**FROM** financial.loan **as** l

4

**WHERE** l.status **IN** ('A', 'C')

5

)

6

**SELECT** (**SELECT** SUM(amount) **FROM** tmp\_results) - (**SELECT** amount **FROM** cte)

Czym weryfikujemy, że zadanie zostało wykonane poprawnie.