# **Оновлене Технічне завдання (ТЗ)**

## **1. Вхідні дані**

Таблиця credit\_contracts із такими важливими полями:

* document\_number
* balance
* overdue\_balance
* overdue\_interest
* overdue\_commission
* overdue\_days\_body
* overdue\_days\_interest
* status
* start\_date
* contract\_amount
* tranche\_type

## **2. Відбір записів**

* Виключити кредити зі статусами 'Закрыт' та 'Зарезервирован'.
* Вибрати кредити з start\_date >= '2022-06-01' або старіші без прострочок і tranche\_type відмінним від 'род'.

## **3. Додаткові колонки в CTE base**

* balance\_final = balance + COALESCE(overdue\_balance, 0).
* Визначення типу кредиту credit\_type (ОВЕР, ВКЛ, КРБ).
* Визначення, чи кредит є траншем чи основним (vkl\_subtype).
* Формування main\_vkl\_id для групування траншів.
* Визначення типу прострочки overdue\_type.

## **4. Агрегація траншів ВКЛ (CTE vkl\_group)**

* Обчислити суму balance\_final траншів по кожному основному ВКЛ (main\_vkl\_id).

## **5. Логіка розрахунку utilization**

* Для КРБ: завжди 1.0.
* Для ОВЕР:  
  + Якщо balance\_final = 0 — утилізація = 0.
  + Якщо contract\_amount NULL або 0 (немає актуального ліміту), але є баланс — утилізація = 1 (максимальна).
  + Інакше утилізація = мінімум із balance\_final / contract\_amount і 1.
* Для ВКЛ:  
  + Якщо основний договір (vkl\_subtype = 'основний'):  
    - Якщо сумарний баланс траншів = 0 — утилізація = 0.
    - Якщо contract\_amount NULL або 0, але є транші — утилізація = 1.
    - Інакше утилізація = total\_tranche\_balance / contract\_amount (округлено до 6 знаків).
  + Для траншів утилізація = NULL.
* Для інших випадків утилізація = NULL.

## **6. Підключення даних**

* base з’єднується з vkl\_group по main\_vkl\_id для отримання суми траншів.

## **7. Результат**

* VIEW overdue\_dataset з усіма колонками з credit\_contracts і новими:  
  + balance\_final
  + credit\_type
  + vkl\_subtype
  + main\_vkl\_id
  + overdue\_type
  + utilization з оновленою логікою.

CREATE OR REPLACE VIEW overdue\_dataset AS

WITH base AS (

SELECT

\*,

balance + COALESCE(overdue\_balance, 0) AS balance\_final,

-- Визначаємо тип кредиту

CASE

WHEN document\_number LIKE '%-О%' THEN 'ОВЕР'

WHEN document\_number LIKE '%VKL%' THEN 'ВКЛ'

ELSE 'КРБ'

END AS credit\_type,

-- Визначаємо, чи це транш

CASE

WHEN document\_number LIKE '%/%/%' THEN 'транш'

ELSE 'основний'

END AS vkl\_subtype,

-- Визначаємо ID головного ВКЛ для групування траншів

CASE

WHEN document\_number LIKE '%/%/%' THEN split\_part(document\_number, '/', 1) || '/' || split\_part(document\_number, '/', 2)

ELSE document\_number

END AS main\_vkl\_id,

-- Тип прострочки

CASE

WHEN COALESCE(overdue\_balance, 0) = 0

AND COALESCE(overdue\_interest, 0) = 0

AND COALESCE(overdue\_commission, 0) = 0 THEN 'без прострочки'

WHEN COALESCE(overdue\_days\_body, 0) >= 5

OR COALESCE(overdue\_days\_interest, 0) >= 5 THEN 'більше 5'

ELSE 'менше 5'

END AS overdue\_type

FROM credit\_contracts

WHERE

status NOT IN ('Закрыт', 'Зарезервирован')

AND (

start\_date >= '2022-06-01'

OR (

start\_date < '2022-06-01'

AND COALESCE(overdue\_balance, 0) = 0

AND COALESCE(overdue\_interest, 0) = 0

AND COALESCE(overdue\_commission, 0) = 0

AND tranche\_type IS DISTINCT FROM 'род'

)

)

),

-- Сумуємо balance\_final по траншах для кожного основного ВКЛ (main\_vkl\_id)

vkl\_group AS (

SELECT

main\_vkl\_id,

SUM(balance\_final) AS total\_tranche\_balance

FROM base

WHERE credit\_type = 'ВКЛ' AND vkl\_subtype = 'транш'

GROUP BY main\_vkl\_id

)

SELECT

b.\*,

CASE

WHEN credit\_type = 'КРБ' THEN 1.0

WHEN credit\_type = 'ОВЕР' THEN

CASE

WHEN balance\_final = 0 THEN 0

WHEN contract\_amount IS NULL OR contract\_amount = 0 THEN 1

ELSE LEAST(balance\_final / contract\_amount, 1.0)

END

WHEN credit\_type = 'ВКЛ' AND vkl\_subtype = 'основний' THEN

CASE

WHEN total\_tranche\_balance = 0 THEN 0

WHEN contract\_amount IS NULL OR contract\_amount = 0 THEN 1

ELSE ROUND(total\_tranche\_balance / contract\_amount, 6)

END

WHEN credit\_type = 'ВКЛ' AND vkl\_subtype = 'транш' THEN NULL

ELSE NULL

END AS utilization

FROM base b

LEFT JOIN vkl\_group v ON b.main\_vkl\_id = v.main\_vkl\_id;