Twoja firma korzysta z aplikacji z bazą danych do zarządzania asortymentem. Na początku wystarczało im przypisywanie statusów asortymentu dla każdego produktu i każdego sklepu osobno. Z czasem jednak zdecydowali się wprowadzić **klastry sklepów** (każdy klaster może zawierać wiele sklepów) i ustawiać statusy głównie na poziomie **klaster + produkt** (choć w niektórych sytuacjach nadal będą śledzić statusy na poziomie **produkt + sklep**). Zmieniono już strukturę tabel w bazie i poproszono Cię o napisanie skryptu migracyjnego w SQL, który wypełni nową tabelę statusów na podstawie istniejących danych.

Baza danych ma tabele:

* **products**
  + id (integer): Klucz główny
  + name (text): Nazwa produktu
* **stores**
  + id (integer): Klucz główny
  + name (text): Nazwa sklepu
  + cluster\_id (integer): ID klastra (może być NULL)
* **clusters**
  + id (integer): Klucz główny
  + name (text): Nazwa klastra
* **statuses**
  + id (integer): Klucz główny
  + name (text): Nazwa statusu
  + priority (integer): Priorytet statusu
* **product\_store\_statuses**
  + id (integer): Klucz główny
  + product\_id (integer): ID produktu
  + store\_id (integer): ID sklepu
  + status\_id (integer): ID statusu
* **product\_cluster\_statuses**
  + id (integer): Klucz główny
  + product\_id (integer): ID produktu
  + cluster\_id (integer): ID klastra
  + status\_id (integer): ID statusu

**Twoje zadanie** to uzupełnić tabelę **product\_cluster\_statuses** na podstawie danych z tabeli **product\_store\_statuses**, a następnie usunąć dane nadmiarowe z tabeli **product\_store\_statuses**. Są więc dwa kroki:

**1. Uzupełnienie statusów dla klastrów**

* Dla każdego produktu sprawdź, w których sklepach ma ustawione statusy i do jakich klastrów te sklepy należą.
* Dla tych klastrów ustaw statusy dla tego samego produktu w tabeli **product\_cluster\_statuses**.
* Dla pary *(produkt, klaster)* status powinien być tym, który **najczęściej** występuje w sklepach tego klastra.
* W przypadku remisu wybierz status o **niższej wartości w polu priority** (zakładamy, że nie ma dwóch statusów z tym samym priorytetem).
* Sklepy, które nie mają przypisanego klastra (cluster\_id IS NULL), nie są brane pod uwagę.
* Liczą się tylko klucze obce, więc nie musisz przejmować się polem id ani kolejnością wstawiania rekordów.

**2. Usunięcie nadmiarowych statusów ze sklepów**

* Po kroku 1 zdarzy się, że niektóre sklepy będą miały status identyczny z tym, który został wyznaczony dla ich klastra.
* Takie wpisy należy **usunąć** z tabeli **product\_store\_statuses**, bo są redundantne.
* Jeżeli jednak sklep ma status inny niż status klastra → ten wpis **zostaje**.
* Sklepy bez klastra nie są w ogóle przetwarzane.