

BD4U - Filtrowanie zaawansowane

BETWEEN, IN, LIKE, ILIKE, IS NULL - PostgreSQL / pgAdmin 4

1. BETWEEN - zakres wartości

BETWEEN sprawdza, czy wartość mieści się w podanym zakresie. Obie wartości graniczne są włączone.

```
SELECT imie, nazwisko, wiek -- Wybieramy podstawowe dane ucznia.  
FROM uczniowie            -- Dane pobieramy z tabeli uczniowie.  
WHERE wiek BETWEEN 17 AND 19; -- Zostawiamy wiek od 17 do 19 włącznie.
```

2. IN - jedna z wielu wartości

```
SELECT imie, nazwisko, klasa -- Wybieramy dane ucznia i klasę.  
FROM uczniowie              -- Dane pobieramy z tabeli uczniowie.  
WHERE klasa IN ('4A','4B','5A'); -- Dopuszczamy każdą z trzech wymienionych klas.
```

NOT IN

```
SELECT * -- Pobieramy wszystkie kolumny.  
FROM uczniowie -- Dane pobieramy z tabeli uczniowie.  
WHERE miasto NOT IN ('Poznań','Luboń'); -- Odrzucamy dwa wskazane miasta.
```

3. LIKE i ILIKE - wyszukiwanie wzorców

LIKE rozróżnia wielkość liter. PostgreSQL udostępnia także ILIKE, które nie rozróżnia wielkości liter.

```
SELECT imie, nazwisko      -- Wybieramy imię i nazwisko.
FROM uczniowie            -- Dane pobieramy z tabeli uczniowie.
WHERE nazwisko LIKE 'K%';  -- Szukamy nazwisk zaczynających się literą K.
```

Symbol %

% oznacza dowolny ciąg znaków, także pusty.

```
SELECT imie, nazwisko      -- Wybieramy imię i nazwisko.
FROM uczniowie            -- Dane pobieramy z tabeli uczniowie.
WHERE nazwisko ILIKE '%ski%'; -- Szukamy fragmentu 'ski' niezależnie od wielkości liter.
```

Symbol _

Podkreślenie _ oznacza dokładnie jeden dowolny znak.

```
SELECT imie                -- Wybieramy tylko imię.
FROM uczniowie            -- Dane pobieramy z tabeli uczniowie.
WHERE imie LIKE 'A___';    -- Szukamy czteroliterowych imion zaczynających się na A.
```

4. IS NULL i IS NOT NULL

NULL oznacza brak wartości. Do sprawdzania NULL nie używamy = ani <>.

```
SELECT imie, nazwisko, email -- Wybieramy dane kontaktowe.
FROM uczniowie              -- Dane pobieramy z tabeli uczniowie.
WHERE email IS NULL;        -- Zostawiamy osoby bez wpisanego adresu email.
```

```
SELECT imie, nazwisko, email -- Wybieramy dane kontaktowe.
FROM uczniowie              -- Dane pobieramy z tabeli uczniowie.
WHERE email IS NOT NULL;    -- Zostawiamy osoby z wpisanym adresem email.
```

5. Łączenie filtrów zaawansowanych

```
SELECT imie, nazwisko, klasa, wiek -- Wybieramy potrzebne kolumny.  
FROM uczniowie -- Dane pobieramy z tabeli uczniowie.  
WHERE klasa IN ('4A','4B') -- Dopuszczamy dwie klasy.  
AND wiek BETWEEN 17 AND 18 -- Ograniczamy wiek do 17-18 lat.  
AND nazwisko ILIKE '%a%'; -- Nazwisko ma zawierać literę a niezależnie od wielkości liter.
```

6. WHERE + ORDER BY

```
SELECT imie, nazwisko, miasto -- Wybieramy dane ucznia.  
FROM uczniowie -- Dane pobieramy z tabeli uczniowie.  
WHERE miasto IN ('Poznań','Gniezno') -- Filtrujemy do dwóch miast.  
ORDER BY miasto ASC, nazwisko ASC; -- Sortujemy po mieście, a potem po nazwisku.
```

7. Najczęstsze błędy

Najczęstsze błędy: użycie = NULL zamiast IS NULL, brak apostrofów przy tekstach w IN, pomylenie % z *, użycie LIKE w oczekiwaniu na niewrażliwość na wielkość liter oraz zapomnienie, że BETWEEN obejmuje obie granice.