

# Bazy danych 2022: rozwiązania zadań z listy nr 1

Bartosz Brzostowski

4 kwietnia 2022

Często istnieje kilka istotnie różnych, równie dobrych rozwiązań tego samego zadania. Poniższe rozwiązania są przykładowe. Jeżeli jednak Twoje rozwiązanie bardzo różni się od przykładowego, a zwłaszcza jeśli daje różne wyniki, zastanów się, na czym polega różnica i ew. skonsultuj się z prowadzącym.

Ustawianie aliasów kolumn (AS) nie jest obowiązkowe. Proszę przy okazji oswoić się z łamaniem wierszy i wstawianiem wcięć w długich zapytaniach.

## Zadanie 1

```
SELECT DAYOFYEAR(CURDATE());
```

## Zadanie 2

Gdyby to zadanie miał rozwiązać Michael Widenius, twórca MySQL i MariaDB, wyglądałoby to tak:

```
SELECT DAYNAME("1962-03-03");
```

Ze względu na luźne sformułowanie treści zadania równie dobrze można użyć funkcji DAYOFWEEK lub WEEKDAY.

## Zadanie 3

Rozwiązanie w całości „matematyczne”:

```
SELECT FLOOR(PI() * POW(10, 10)) % 10;
```

Rozwiązanie „napisowe”, wymagające wymuszenia odpowiedniej precyzji wyświetlania  $\pi$ :

```
SELECT SUBSTRING(PI() + 0.000000000000, 12, 1);
```

Co ciekawe, jeśli napiszemy o jedną cyfrę 0 mniej, dostaniemy wynik błędny ze względu na zaokrąglenie:  $3.14159265359 \approx 3.1415926536$ .

## Zadanie 4

Problem zaokrąglenia można uniknąć używając funkcji TRUNCATE(). Tutaj odczyt fragmentu łańcucha przy pomocy innej funkcji niż w poprzednim zadaniu:

```
SELECT RIGHT(TRUNCATE(EXP(1),8),1);
```

## Zadanie 5

```
SELECT FLOOR(RAND() * 6) + 1;
```

## Zadanie 6

```
SELECT nazwa, cena * ilosc AS wartosc FROM produkty;
```

## Zadanie 7

```
SELECT idk, nazwa, miasto,  
       SUBSTRING_INDEX(adres, ". ", -1) AS skrocony_adres,  
       telefon  
FROM klienci;
```

## Zadanie 8

```
SELECT CONCAT_WS(", ", nazwa, adres, miasto, telefon) AS wizytowka  
FROM klienci;
```

Można oczywiście użyć funkcji CONCAT i ręcznie wstawić ", " jako dodatkowe argumenty pomiędzy powyższymi, ale tak jest chyba jednak schludniej.

## Zadanie 9

```
SELECT nazwa,  
       JSON_OBJECT("zlotych", FLOOR(cena),  
                  "groszy", ROUND(100 * cena % 100)  
       ) AS obiekt  
FROM produkty;
```

Gdyby w tym zadaniu użyć FLOOR zamiast ROUND, moglibyśmy dostać w niektórych przypadkach niepoprawne wyniki wynikające z błędów zaokrągleń, ponieważ kolumna cena jest typu zmiennoprzecinkowego, a nie stałoprzecinkowego.

## Zadanie 10

```
SELECT idz, SUBDATE(ADDDATE(DATE(data), INTERVAL 1 MONTH),
    CASE WEEKDAY(ADDDATE(DATE(data), INTERVAL 1 MONTH))
        WHEN 5 THEN 1 WHEN 6 THEN 2 ELSE 0 END
    ) AS data_faktury
FROM zamow;
```

Tu można by było użyć dwóch zagnieżdżonych IFów, ale to jednak dobry pretekst do użycia instrukcji CASE. Sprawdzamy, czy dzień tygodnia (WEEKDAY) przypadający miesiąc od zamówienia (ADDDATE(DATE(data), INTERVAL 1 MONTH)) to sobota (5), niedziela (6), czy coś innego, i odejmujemy od tej daty odpowiednio 1, 2 lub 0 dni.

## Zadanie 11

```
SELECT nazwa FROM klienci WHERE miasto != "Wrocław";
```

Użycie konstrukcji miasto NOT LIKE "%Wrocław%" byłoby tu niepoprawne.

## Zadanie 12

```
SELECT nazwa
FROM klienci
WHERE LOCATE("s", nazwa) BETWEEN 1 AND LENGTH(nazwa) / 2;
```

## Zadanie 13

```
SELECT * FROM zamow WHERE WEEKDAY(data) = 0 OR WEEKDAY(data) = 4;
```

## Zadanie 14

```
SELECT * FROM zamow WHERE TIME(data) > "16:00:00";
```

## Zadanie 15

```
SELECT nazwa FROM produkty WHERE nazwa NOT LIKE "%Apple%";
```

## Zadanie 16

```
SELECT nazwa FROM klienci WHERE adres LIKE "%Marszałkowska%";
```

## Zadanie 17

```
SELECT DISTINCT miasto FROM klienci WHERE LENGTH(adres) > 15;
```

Proszę zwrócić uwagę na pojawienie się modyfikatora DISTINCT.

## Zadanie 18

```
SELECT nazwa, IF(cena > 1500, CEILING(cena * 1.1 / 12), NULL) AS rata  
FROM produkty;
```

Najwłaściwszą „wartością” wysokości raty dla produktów niedostępnych w sprzedaży ratnej jest oczywiście NULL. Jeśli ktoś chciałby się upierać, że cena musi mieć dwa miejsca po przecinku, należałoby użyć np. wyrażenia `CEILING(...) + 0.00`.

## Zadanie 19

```
SELECT * FROM produkty;  
SELECT * FROM produkty ORDER BY ilosc;  
-- najwięcej (21) sztuk jest produktu o id 4  
SELECT * FROM detal_zamow WHERE p_id = 4;  
SELECT DISTINCT * FROM detal_zamow WHERE p_id = 4;  
-- 4 rows in set (0,00 sec)  
-- odpowiedź: 4
```