

SYSDATE:

SYSDATE zwraca obecną datę i czas z systemu operacyjnego na którym silnik bazodanowy działa
Przykład:

```
SELECT TO_CHAR(SYSDATE, 'DD-MM-YYYY HH24:MI:SS') "TERAZ"  
FROM DUAL;
```

Wynik:

```
TERAZ
```

```
-----
```

```
17-04-2018 12:45:51
```

MONTHS_BETWEEN:

MONTHS_BETWEEN(date1, date2) – zwraca liczbę miesięcy pomiędzy date1 oraz date2. Jeżeli date1 następuje po date2 wynik jest dodatni. Jeżeli date1 występuje przed date2 wynik jest ujemny. Gdy dni miesiąca zawarte w datach są takie same lub jeśli są to ostatnie dni miesiąca wynik jest liczą całkowitą. W przeciwnym wypadku jest obliczany ułamek bazując na 31 dniowym miesiącu.

Przykład:

```
SELECT MONTHS_BETWEEN  
      (TO_DATE('02-02-1995', 'MM-DD-YYYY'),  
       TO_DATE('01-01-1995', 'MM-DD-YYYY')) "MIESIACE"  
FROM DUAL;
```

Wynik:

```
MIESIACE
```

```
-----
```

```
1.03225806
```

DISTINCT:

Klauzula DISTINCT jest używana w celu usunięcia duplikujących się rekordów z wyniku.

Składnia: `SELECT DISTINCT wyrażenia FROM tabele [WHERE warunki];`

HAVING:

Klauzula HAVING użyta wraz z klauzulą GROUP BY organicza grupy zwracanych wierszy do takich, które spełniają warunek.

Składnia:

```
SELECT wyrażenie1, wyrażenie_n,  
       Funkcja_agregująca(wyrażenie_agregowane)  
FROM tabele  
[WHERE warunki]  
GROUP BY wyrażenie1, wyrażenie_n  
HAVING warunek_HAVING;
```

Przykład:

```
SELECT department, SUM(sales) AS "Sprzedaz calkowita"  
FROM order_details  
GROUP BY department  
HAVING SUM(sales) > 25000;
```

NEXT_DAY:

NEXT_DAY(date,day) zwraca datę pierwszego dnia o nazwie (day), następującego po date.

Przykład:

```
SELECT NEXT_DAY('02-FEB-2001','TUESDAY') "NEXT DAY"  
       FROM DUAL;
```

```
NEXT DAY  
-----  
06-FEB-2001
```

Pierwszy wtorek po 2 lutego 2001.

CASE:

Wyrażenie CASE wybiera spośród postawionych warunków spełniony i wykonuje następujący po nim ciąg czynności.

```
CASE { WHEN wyrażenie logiczne THEN {wyrażenie;} ... }...  
[ ELSE {wyrażenie;}... ]  
END CASE [ nazwa_etykiety ];
```

Przykład:

```
SELECT table_name,  
CASE  
  WHEN owner='SYS' THEN 'The owner is SYS'  
  WHEN owner='SYSTEM' THEN 'The owner is SYSTEM'  
  ELSE 'The owner is another value'  
END  
FROM all_tables;
```

ZADANIA DO WYKONANIA:

1. Wyświetl nazwę miasta/miast, z którego pochodzi najmniejsza liczba studentów na kierunku matematyka.
2. Wyświetl nazwiska prowadzących, którzy są doktorami i uczą Analizy Danych.
3. Wyświetl dane prowadzącego oraz tytuł naukowy, dla tych osób, które mają najwięcej wykładów.
4. Wyświetl dane o studentach, którzy dostali stypendium na najkrótszy okres czasu.
5. Wyświetl ile średnio ocen z przedmiotu otrzymuje grupa 'GRUPA 21'.
6. Wyświetl dane (imię i nazwisko) prowadzących, którzy nie mają zajęć w poniedziałek i czwartek w salach zaczynających się na literę K.
7. Wyświetl jakie zajęcia (charakter) odbywają się w budynku Wydziału Inżynierii i Technologii Chemicznej.
8. Kto prowadzi seminaria (dane osobowe+ tytuł naukowy+ kierunek). Dane wyświetl wg. wzoru:
*****Adam Mickiewicz*****prof. dr hab. *****Matematyka*****
9. Wyświetl dane o studentach, których stypendia są mniejsze niż najmniejsze stypendium na Informatyce.
10. Wyświetl nazwiska, imiona i numery albumów studentów, którzy mają najwięcej ocen z ćwiczeń.
11. Wyświetl, w które dni tygodnia jest najmniej zajęć.
12. Wyświetl imię i nazwisko, nr albumu najlepszego studenta (najwyższa średnia) oraz jego pełny adres.