

Concat(str1, str2) - zwraca ciąg znaków będący połączeniem ciągu str1 z ciągiem str2

```
SELECT FIRST_NAME || ' ' || LAST_NAME FROM EMPLOYEES; // WYŚWIETLA WYNIKI ODDZIELONE SPACJĄ
```

```
SELECT CONCAT(FIRST_NAME, LAST_NAME) FROM EMPLOYEES; // WYŚWIETLA WYNIKI BEZ SPACJI
```

Initcap(str) - zwraca sformatowany ciąg wejściowy, tak, że pierwsza litera jest duża, zaś pozostałe małe.

```
SELECT INITCAP(EMAIL) FROM EMPLOYEES; // TYLKO PIERWSZA LITERA DUŻA W EMAIL
```

INSTR(ciąg_znaków_1, ciąg_znaków_2 [m, n]) – szuka n-tego wystąpienia ciągu2 w ciągu1 rozpoczynając od pozycji m.

```
SELECT DEPARTMENT_NAME, INSTR(DEPARTMENT_NAME, 'a', 1, 1) FROM DEPARTMENTS; //  
Wyszukajmy pozycję pierwszej litery 'a' w DEPARTMENT_NAME – szukając od pierwszego znaku
```

Length(str) - zwraca długość str.

```
SELECT COUNTRY_ID, COUNTRY_NAME, LENGTH(COUNTRY_NAME) FROM countries; // ZLICZA  
DŁUGOŚĆ NAZW W COUNTRY_NAME
```

Lower(str) - zwraca łańcuch str przekształcony na małe litery.

```
SELECT LOWER(EMAIL) FROM EMPLOYEES; // PRZEKSZTAŁCA Z DUŻYCH LITER NA MAŁE EMAIL
```

LPAD(ciąg_znaków_1, n1 [, ciąg_znaków_2]) – zwraca ciąg znaków o długości n1 utworzony przez dodanie z lewej strony znaków z ciąg_znaków_2. W przypadku braku ciąg_znaków_2 dodawane są spacje.

```
SELECT DEPARTMENT_NAME, LPAD(DEPARTMENT_NAME, 5, '+') FROM DEPARTMENTS; // WYŚWIETLA  
5 LITER ZACZYNAJĄC OD LEWEJ STRONY; WYNIK: Marketing BĘDZIE Marke, IT BĘDZIE +++IT (JEŚLI ZBYT  
MAŁO DODAJE '+')
```

LTRIM(ciąg_znaków_1 [, ciąg_znaków_2]) – usuwa z ciąg_znaków_1 z lewej strony znaki zawarte w ciąg_znaków_2. W przypadku braku ciąg_znaków_2 usuwane są spacje.

```
SELECT LTRIM('ADMINISTRATION', 'A') FROM DUAL; // USUWA A ZACZYNAJĄC OD LEWEJ STRONY
```

REPLACE(ciąg_znaków1, ciąg_znaków2 [, ciąg_znaków3]) – zamienia wszystkie wystąpienia ciąg_2 w ciąg_1 na ciąg_3. W przypadku braku ciąg_3 zamienia na znak pusty

```
SELECT FIRST_NAME, REPLACE(FIRST_NAME, 'a', 'la') FROM EMPLOYEES; // ZAMIENIA KAŻDĄ LITERĘ 'a'  
NA 'la'; WYNIK: Neena → Neenla, Valli → Vlalli
```

RPAD(ciąg_znaków_1, n1 [, ciąg_znaków_2]) – działa analogicznie jak LPAD dodając znaki z prawej strony.

```
select FIRST_NAME, RPAD(FIRST_NAME, 5, '|') from EMPLOYEES; // WYŚWIETLA 5 LITER JEŚLI WYRAZ  
JEST KRÓTKI DODAJE ZNAK '|'
```

RTRIM(ciąg_znaków_1 [,ciąg_znaków_2]) – działa analogicznie jak LTRIM usuwając znaki z prawej strony.

select RTRIM('DONALD','D') from DUAL; //WYNIK: DONALD→DONAL

SUBSTR(ciąg_znaków_1, m [, n]) – wybiera z ciąg_znaków n znaków od m pozycji. W przypadku braku n wybiera znaki do końca.

select FIRST_NAME, LAST_NAME, SUBSTR(FIRST_NAME,1,1) || SUBSTR(LAST_NAME,1,1) from EMPLOYEES; //WYŚWIETLA INICJAŁY PRACOWNIKÓW, WYNIK: Donald, OConnell→DO

TRIM(ciąg_znaków_1) – usuwa z ciąg_znaków_1 spacje – zarówno z lewej jak i z prawej strony

select TRIM(' BAZA ') from DUAL; //USUWA SPACJE

Upper(str) - przekształca wszystkie litery w str na duże.

select REGION_NAME, UPPER(REGION_NAME) from REGIONS; //ZMIENIA NA DUŻE LITERY

ABS(n1) – zwraca wartość bezwzględną.

select ABS(343.56), ABS(-343.56), ABS(0) from DUAL; //WARTOŚĆ BEZWZGLĘDNA -343.56→343.56

CEIL(n1) – zwraca najmniejszą liczbę całkowitą większą lub równą liczbie n1.

select CEIL(343.56), CEIL(-343.56), CEIL(0) from DUAL; //WYNIK: 343.56→344, -343.56→-343

FLOOR(n1) – zwraca największą liczbę całkowitą mniejszą lub równą liczbie n1

SELECT FLOOR(565.56), FLOOR(-565.56), FLOOR(0) FROM DUAL; //WYNIK: 565.56→565, -565.56→-566

Cos(n) - zwraca cosinus kata n (wyrażonego w radianach)

select COS(30), COS(90), COS(0) from DUAL; //COS W RADIANACH

Ln(n) - zwraca logarytm naturalny z n

SELECT LN(95) "LOGARYTM NATURALNY Z 95:" FROM DUAL; //LOGARYTM

Log(n, m) - zwraca logarytm z m przy podstawie n

SELECT SALARY, LOG(10, SALARY) "LOGARYTM PRZY PODSTAWIE 10" FROM EMPLOYEES;

MOD(n1, n2) – modulo – zwraca resztę z dzielenia całkowitego n1 przez n2

SELECT MIN_SALARY, MOD(MIN_SALARY, 3) FROM JOBS; //RESZTA Z DZIELENIA

POWER(n1, n2) – zwraca n1 podniesioną do potęgi n2

SELECT POWER(6, 5) "6 DO POTĘGI 5" FROM DUAL; //POTĘGA

ROUND(n1, n2) – zaokrągla liczbę n1 do n2 miejsc dziesiętnych

SELECT ROUND(489.6348, 2), ROUND(-489.6378, 2) FROM DUAL; //ZAOKRĄGLENIE DO 2 MIEJSC

SIGN(n1) – zwraca 1 dla n1 dodatniego, -1 dla n1 ujemnego i 0 dla n1 równego 0.

SELECT SIGN(489),SIGN(-489),SIGN(0) FROM DUAL; //WYNIK:489→1; -489→-1; 0→0

Sin(n) - zwraca sinus kata n (wyrażonego w radianach)

SELECT SIN(90),SIN(0) FROM DUAL; //SINUS

Sqrt(n) - zwraca pierwiastek kwadratowy z n

SELECT SQRT((123.34*16)) PIERWIASTEK FROM DUAL; //PIERWIASTEK

Tan(n) - zwraca tangens kata n (wyrażonego w radianach)

SELECT TAN(60) TANGENS FROM DUAL; //TANGENS

TRUNC(n1,n2) – obcina liczbę n1 do n2 miejsc dziesiętnych

SELECT TRUNC(86.888,2) TRUNC FROM DUAL; //TRUNC

ZADANIA Z MOODLE

-----FUNKCJE MATEMATYCZNE-----

--01. O jaką kwotę (wartość bezwzględna) odbiegają zarobki pracowników od średniej w firmie -- wynoszącej 6461.68.

SELECT first_name,salary,ABS(salary-6461.68) WARTOSC FROM EMPLOYEES;

--02. Zaokrągł powyższy wynik do 1 miejsca po przecinku , do przecinka i do dziesiątek używając funkcji -- Round() i Trunc().

SELECT first_name,salary,ABS(salary-6461.68) WARTOSC, ROUND((salary-6461.68),1) ROUND, TRUNC((salary-6461.68),1) TRUNC FROM EMPLOYEES;

--03. Wylicz w kolejnych zapytaniach:

-- wartość wyrażenia 123.34*16,

SELECT (123.34*16) MNOZENIE FROM DUAL;

-- wylicz z niego pierwiastek,

SELECT (123.34*16) MNOZENIE,SQRT((123.34*16)) PIERWIASTEK FROM DUAL;

-- przetestuj na nim działanie funkcji Sqrt, Round,

-- Trunc, Ceil i Floor.

SELECT (123.34*16) MNOZENIE,SQRT((123.34*16)) PIERWIASTEK, ROUND(SQRT((123.34*16)),2) ROUND, TRUNC(SQRT((123.34*16)),1) TRUNC,CEIL(SQRT((123.34*16))) CEIL,FLOOR(SQRT((123.34*16))) FLOOR FROM DUAL;