

Lekcja

Temat: Funkcje związane z czasem, datą, operatorami łańcuchowymi

Funkcje daty i czasu

Link do dokumentacji MySQL:

<https://dev.mysql.com/doc/refman/8.4/en/date-and-time-functions.html>

Metoda	Wyjaśnienie	Przykład SQL	Wynik
ADDDATE()	Dodaje interwał do daty	SELECT ADDDATE('2024-01-01', INTERVAL 5 DAY);	2024-01-06
ADDTIME()	Dodaje czas	SELECT ADDTIME('10:00:00','02:30:00');	12:30:00
CONVERT_TZ()	Konwersja strefy czasowej	SELECT CONVERT_TZ('2024-01-01 12:00','UTC','Europe/Warsaw');	2024-01-01 13:00
CURDATE()	Bieżąca data	SELECT CURDATE();	2025-11-10
CURTIME()	Bieżący czas	SELECT CURTIME();	np. 14:22:01
DATE()	Zwraca część datową	SELECT DATE('2024-01-01 10:00:00');	2024-01-01
DATE_ADD()	Dodaje interwał do daty	SELECT DATE_ADD('2024-01-01', INTERVAL 1 MONTH);	2024-02-01
DATE_FORMAT()	Formatuje datę	SELECT DATE_FORMAT('2024-01-15','%d-%m-%Y');	15-01-2024
DATE_SUB()	Odejmuje interwał	SELECT DATE_SUB('2024-01-10', INTERVAL 3 DAY);	2024-01-07
DATEDIFF()	Różnica między datami	SELECT DATEDIFF('2024-02-01','2024-01-01');	31

DAY()	Dzień miesiąca	<code>SELECT DAY('2024-01-15');</code>	15
DAYNAME()	Nazwa dnia	<code>SELECT DAYNAME('2024-01-15');</code>	Tuesday
DAYOFMONTH()	Dzień miesiąca	<code>SELECT DAYOFMONTH('2024-01-15');</code>	15
DAYOFWEEK()	Numer dnia tyg. (1=nd)	<code>SELECT DAYOFWEEK('2024-01-15');</code>	3
DAYOFYEAR()	Dzień roku	<code>SELECT DAYOFYEAR('2024-01-15');</code>	15
EXTRACT()	Wyodrębnia część daty	<code>SELECT EXTRACT(YEAR FROM '2024-01-15');</code>	2024
FROM_DAYS()	Dni → data	<code>SELECT FROM_DAYS(750000);</code>	2044-01-22
FROM_UNIXTIME()	UNIX → data	<code>SELECT FROM_UNIXTIME(1700000000);</code>	2023-11-14 22:13:20
HOUR()	Pobiera godzinę	<code>SELECT HOUR('12:45:00');</code>	12
LAST_DAY()	Ostatni dzień miesiąca	<code>SELECT LAST_DAY('2024-02-10');</code>	2024-02-29
MAKEDATE()	Tworzy datę z dnia roku	<code>SELECT MAKEDATE(2024,32);</code>	2024-02-01
MAKETIME()	Tworzy czas	<code>SELECT MAKETIME(10,20,30);</code>	10:20:30
MICROSECOND()	Mikrosekundy	<code>SELECT MICROSECOND('10:00:00.123456');</code>	123456

MINUTE()	Minuta	<code>SELECT MINUTE('12:45:30');</code>	45
MONTH()	Numer miesiąca	<code>SELECT MONTH('2024-05-10');</code>	5
MONTHNAME()	Nazwa miesiąca	<code>SELECT MONTHNAME('2024-05-10');</code>	May
NOW()	Aktualny datetime	<code>SELECT NOW();</code>	2025-11-10 14:20:xx
PERIOD_ADD()	Dodaje miesiące do YYYYMM	<code>SELECT PERIOD_ADD(202401,2);</code>	202403
PERIOD_DIFF()	Ilość miesięcy między okresami	<code>SELECT PERIOD_DIFF(202402,202401);</code>	1
QUARTER()	Kwartał	<code>SELECT QUARTER('2024-05-10');</code>	2
SEC_TO_TIME()	Sekundy → czas	<code>SELECT SEC_TO_TIME(3661);</code>	01:01:01
SECOND()	Sekundy	<code>SELECT SECOND('12:45:59');</code>	59
STR_TO_DATE()	Tekst → data	<code>SELECT STR_TO_DATE('31-01-2024','%d-%m-%Y');</code>	2024-01-31
SUBTIME()	Odejmuje czas	<code>SELECT SUBTIME('10:00:00','01:30:00');</code>	08:30:00
SYSDATE()	Czas wykonania	<code>SELECT SYSDATE();</code>	2025-11-10...

TIME()	Czas z datetime	<code>SELECT TIME('2024-01-01 12:30:45');</code>	12:30:45
TIME_FORMAT()	Formatuje czas	<code>SELECT TIME_FORMAT('12:30:45','%H:%i');</code>	12:30
TIME_TO_SEC()	Czas → sekundy	<code>SELECT TIME_TO_SEC('01:00:00');</code>	3600
TIMEDIFF()	Różnica czasu	<code>SELECT TIMEDIFF('12:00:00','10:00:00');</code>	02:00:00
TIMESTAMP()	Tworzy datetime	<code>SELECT TIMESTAMP('2024-01-01');</code>	2024-01-01 00:00:00
TIMESTAMPADD()	Dodaje interwał	<code>SELECT TIMESTAMPADD(HOUR,2,'2024-01-01 10:00');</code>	2024-01-01 12:00
TIMESTAMPDIFF()	Różnica datetime	<code>SELECT TIMESTAMPDIFF(DAY,'2024-01-01','2024-01-10');</code>	9
TO_DAYS()	Data → dni od roku 0	<code>SELECT TO_DAYS('2024-01-01');</code>	739252
TO_SECONDS()	Data → sekundy od roku 0	<code>SELECT TO_SECONDS('2024-01-01');</code>	64092288000
UNIX_TIMESTAMP()	Aktualny UNIX time	<code>SELECT UNIX_TIMESTAMP();</code>	np. 1768060000
UTC_DATE()	Data UTC	<code>SELECT UTC_DATE();</code>	2025-11-10

UTC_TIME()	Czas UTC	<code>SELECT UTC_TIME();</code>	13:14:xx
UTC_TIMESTAMP()	Datetime UTC	<code>SELECT UTC_TIMESTAMP();</code>	2025-11-10 13:14:xx
WEEK()	Numer tygodnia	<code>SELECT WEEK('2024-01-10');</code>	1
WEEKDAY()	Dzień tyg. (0=pon)	<code>SELECT WEEKDAY('2024-01-10');</code>	3
WEEKOFYEAR()	Tydzien ISO	<code>SELECT WEEKOFYEAR('2024-01-10');</code>	2
YEAR()	Rok	<code>SELECT YEAR('2024-01-10');</code>	2024
YEARWEEK()	Rok + tydzień	<code>SELECT YEARWEEK('2024-01-10');</code>	202402

UTC (Uniwersalny Czas Koordynowany) to światowy standard czasu atomowego, który służy jako podstawa do ustalania lokalnego czasu w różnych strefach czasowych. Polska znajduje się w strefie czasowej UTC+1 (czas środkowoeuropejski, CET) zimą i UTC+2 (czas środkowoeuropejski letni, CEST) latem, a lokalny czas w Polsce jest o 1 lub 2 godziny późniejszy od czasu UTC.

- **Co to jest UTC:**
 - UTC to międzynarodowy standard czasu, który jest niezależny od ruchu obrotowego Ziemi i oparty na bardzo precyzyjnym czasie atomowym.

- Jest to punkt odniesienia, taki sam na całym świecie, do którego dodaje się lub od którego odejmuje się czas, aby uzyskać lokalny czas dla danej strefy czasowej.
- **UTC w Polsce:**
 - Polska leży w strefie czasowej UTC+1 (czas zimowy) lub UTC+2 (czas letni).
 - Czas zimowy (CET): Obowiązuje od ostatniej niedzieli października do ostatniej niedzieli marca. Czas lokalny w Polsce jest o 1 godzinę późniejszy niż UTC. (np. jeśli UTC to 12:00, w Polsce jest 13:00).
 - Czas letni (CEST): Obowiązuje od ostatniej niedzieli marca do ostatniej niedzieli października. Czas lokalny w Polsce jest o 2 godziny późniejszy niż UTC. (np. jeśli UTC to 12:00, w Polsce jest 14:00).

Zastosowania:

- Programowanie - przechowywanie dat i czasu w bazach danych
- Lotnictwo - koordynacja lotów międzynarodowych
- Internet - synchronizacja serwerów
- Telekomunikacja - koordynacja transmisji
- Nauka - precyzyjne pomiary czasu

W praktyce: Gdy widzisz znacznik czasu typu [2025-11-11T14:30:00Z](#), litera "Z" na końcu oznacza właśnie UTC (od "Zulu time" - wojskowego określenia UTC).

Przykłady:

- Polska: UTC+1 (zimą) lub UTC+2 (latem)
- Nowy Jork: UTC-5 (zimą) lub UTC-4 (latem)
- Tokio: UTC+9
- Londyn: UTC+0 (zimą) lub UTC+1 (latem)

Funkcje i operatory łańcuchowe

Link do dokumentacji MySQL:

<https://dev.mysql.com/doc/refman/8.4/en/string-functions.html>

Metoda	Wyjaśnienie	Przykład	Wynik
ASCII()	Zwraca kod ASCII pierwszego znaku	<code>SELECT ASCII('A');</code>	65
BIN()	Zwraca liczbę w postaci binarnej	<code>SELECT BIN(10);</code>	1010
BIT_LENGTH()	Zwraca długość napisu w bitach	<code>SELECT BIT_LENGTH('ABC');</code>	24
CHAR()	Zwraca znak odpowiadający podanemu kodowi ASCII	<code>SELECT CHAR(65);</code>	'A'
CHAR_LENGTH()	Liczba znaków (nie bajtów)	<code>SELECT CHAR_LENGTH('Łódź');</code>	4
CHARACTER_LENGTH()	To samo co CHAR_LENGTH()	<code>SELECT CHARACTER_LENGTH('Test');</code>	4
CONCAT()	Łączy napisy	<code>SELECT CONCAT('A', 'B', 'C');</code>	'ABC'
CONCAT_WS()	Łączy napisy z separatorem	<code>SELECT CONCAT_WS('-', 'A','B','C');</code>	'A-B-C'

ELT()	Zwraca element listy na indeksie (1-based)	SELECT ELT(2,'jeden','dwa','trzy');	'dwa'
EXPORT_SET()	Zamienia liczby bitowe na tekst ON/OFF	SELECT EXPORT_SET(5, 'ON', 'OFF', '', 4);	ON,OFF,ON,OFF
FIELD()	Zwraca pozycję pierwszego argumentu w liście	SELECT FIELD('kot','pies','kot','mysz');	2
FIND_IN_SET()	Pozycja elementu w liście CSV	SELECT FIND_IN_SET('B', 'A,B,C');	2
FORMAT()	Formatuje liczbę z przecinkami	SELECT FORMAT(12345.678, 2);	'12,345.68'
FROM_BASE64()	Dekoduje Base64	SELECT FROM_BASE64('SGVsbgG8=');	'Hello'
HEX()	Zamienia liczbę lub tekst na hex	SELECT HEX('ABC');	414243
INSERT()	Wstawia podciąg w podaną pozycję, zastępując określoną liczbę znaków	SELECT INSERT('abcdef', 3, 2, 'XYZ');	'abXYZef'
INSTR()	Pozycja pierwszego wystąpienia podciągu	SELECT INSTR('abcabc','ca');	3
LCASE()	To samo co LOWER() – zamienia na małe litery	SELECT LCASE('Test');	'test'

LEFT()	Zwraca określoną liczbę znaków od lewej	SELECT LEFT('abcdef', 3);	'abc'
LENGTH()	Długość napisu w bajtach	SELECT LENGTH('ABC');	3
LIKE	Sprawdza dopasowanie wzorca	SELECT 'Ala' LIKE 'A%';	1
LOAD_FILE()	Wczytuje zawartość pliku (jeśli SQL ma dostęp)	SELECT LOAD_FILE('/path/file.txt');	<i>treść pliku</i>
LOCATE()	Pozycja podcięgu (jak INSTR, ale kolejność argumentów odwrotna)	SELECT LOCATE('b','abc');	2
LOWER()	Zamienia na małe litery	SELECT LOWER('TEST');	'test'
LPAD()	Uzupełnia z lewej do zadanej długości	SELECT LPAD('7', 3, '0');	'007'
LTRIM()	Usuwa spacje z lewej	SELECT LTRIM(' test');	'test'
MAKE_SET()	Zwraca listę elementów pasujących do bitów liczby	SELECT MAKE_SET(5,'A','B','C');	'A,C'
MATCH() AGAINST()	Pełnotekstowe wyszukiwanie	SELECT MATCH(text) AGAINST('kot');	<i>ocena dopasowania</i>

MID()	Alias SUBSTRING()	<code>SELECT MID('abcdef', 2, 3);</code>	'bcd'
NOT LIKE	Odwrotność LIKE	<code>SELECT 'Ala' NOT LIKE 'K%';</code>	1
NOT REGEXP	Odwrotność REGEXP	<code>SELECT 'abc' NOT REGEXP '^[0-9]+\$';</code>	1
OCT()	Zamienia liczbę na system ósemkowy	<code>SELECT OCT(15);</code>	'17'
OCTET_LENGTH()	Alias LENGTH()	<code>SELECT OCTET_LENGTH('ABC');</code>	3
ORD()	Kod ASCII pierwszego znaku	<code>SELECT ORD('A');</code>	65
POSITION()	Alias LOCATE()	<code>SELECT POSITION('a' IN 'banan');</code>	2
QUOTE()	Zwraca tekst w bezpiecznej formie (escape)	<code>SELECT QUOTE("Ala's cat");</code>	'Ala\'s cat'
REGEXP	Dopasowanie wyrażenia regularnego	<code>SELECT 'abc123' REGEXP '[0-9]+';</code>	1
REGEXP_INSTR()	Pozycja dopasowania regexu	<code>SELECT REGEXP_INSTR('abc123','[0-9]+');</code>	4
REGEXP_LIKE()	Czy pasuje regex	<code>SELECT REGEXP_LIKE('test123','[a-z]+');</code>	1

REGEXP_REPLACE()	Zamienia dopasowane fragmenty	<code>SELECT REGEXP_REPLACE('a1b2c3','[0-9]','X');</code>	'aXbXcX'
REGEXP_SUBSTR()	Zwraca fragment pasujący do regexu	<code>SELECT REGEXP_SUBSTR('abc123','[0-9]+');</code>	'123'
REPEAT()	Powtarza tekst	<code>SELECT REPEAT('A',3);</code>	'AAA'
REPLACE()	Podmienia tekst	<code>SELECT REPLACE('ala ma kota','a','X');</code>	'XIX mX kotX'
REVERSE()	Odwraça napis	<code>SELECT REVERSE('kota');</code>	'atok'
RIGHT()	Znaki od prawej	<code>SELECT RIGHT('abcdef', 2);</code>	'ef'
RLIKE	Alias REGEXP	<code>SELECT 'abc' RLIKE '[a-z]+';</code>	1
RPAD()	Uzupełnia napis z prawej	<code>SELECT RPAD('A', 4, '.');</code>	'A...'
RTRIM()	Usuwa spacje z prawej	<code>SELECT RTRIM('test ');</code>	'test'
SOUNDEX()	Kod fonetyczny słów	<code>SELECT SOUNDEX('Robert');</code>	'R163'
SOUNDS LIKE	Porównanie brzmienia	<code>SELECT 'Robert' SOUNDS LIKE 'Rupert';</code>	1
SPACE()	Generuje spacje	<code>SELECT SPACE(5);</code>	' '
STRCMP()	Porównuje napisy	<code>SELECT STRCMP('abc','abd');</code>	-1

SUBSTR()	Podciąg (alias SUBSTRING)	<code>SELECT SUBSTR('abcdef',2,3);</code>	'bcd'
SUBSTRING()	Podciąg	<code>SELECT SUBSTRING('abcdef',3);</code>	'cdef'
SUBSTRING_INDEX()	Podciąg do N-tego separatora	<code>SELECT SUBSTRING_INDEX('a,b,c','',2);</code>	'a,b'
TO_BASE64()	Kodowanie Base64	<code>SELECT TO_BASE64('Hello');</code>	'SGVsbG8='
TRIM()	Usuwa spacje z obu stron	<code>SELECT TRIM(' test ');</code>	'test'
UCASE()	Alias UPPER()	<code>SELECT UCASE('abc');</code>	'ABC'
UNHEX()	Hex → tekst	<code>SELECT UNHEX('414243');</code>	'ABC'
UPPER()	Zamienia na wielkie litery	<code>SELECT UPPER('kot');</code>	'KOT'
WEIGHT_STRING()	Zwraca wewnętrzną wagę znaków (techniczne)	<code>SELECT WEIGHT_STRING('A');</code>	(hex bajty)

Na następnej lekcji kartkówka z powyższych tabel. Wymagam znajomości metod napisanych czerwonym kolorem

