

Znakové (textové) funkce pro MYSQL

Name
<u>ASCII ()</u>
<u>BIN ()</u>
<u>BIT LENGTH ()</u>
<u>CHAR ()</u>
<u>CHAR LENGTH ()</u>
<u>CHARACTER LENGTH ()</u>
<u>CONCAT ()</u>
<u>CONCAT WS ()</u>
<u>ELT ()</u>
<u>EXPORT SET ()</u>
<u>FIELD ()</u>
<u>FIND IN SET ()</u>
<u>HEX ()</u>
<u>INSERT ()</u>
<u>INSTR ()</u>
<u>LCASE ()</u> <u>LOWER ()</u>
<u>LEFT ()</u>
<u>LENGTH ()</u>
<u>LIKE</u>
<u>LOCATE ()</u>
<u>LPAD ()</u>
<u>LTRIM ()</u>
<u>MATCH</u>
<u>MID ()</u>
<u>NOT LIKE</u>
<u>NOT REGEXP</u>
<u>OCT ()</u>
<u>ORD ()</u>
<u>POSITION ()</u>
<u>REGEXP</u>
<u>REPEAT ()</u>
<u>REPLACE ()</u>
<u>REVERSE ()</u>
<u>RIGHT ()</u>
<u>RLIKE</u>
<u>RPAD ()</u>
<u>RTRIM ()</u>
<u>SOUNDEX ()</u>
<u>SOUNDS LIKE</u>
<u>SPACE ()</u>
<u>STRCMP ()</u>
<u>SUBSTR ()</u> <u>SUBSTRING ()</u>
<u>SUBSTRING INDEX ()</u>
<u>TRIM ()</u>
<u>UCASE ()</u> <u>UPPER ()</u>
<u>UNHEX ()</u>

Funkce ASCII nebo ORD

ASCII nebo ORD vrátí hodnotu prvního znaku, funguje i pro čísla ;)

SYNTAXE

```
SELECT jmeno, ASCII(jmeno), PLAT1, ASCII(PLAT1) FROM funkce  
SELECT ASCII('€')
```

SYNTAXE

```
SELECT jmeno, ORD(jmeno), PLAT1, ORD(PLAT1) FROM funkce  
SELECT ASCII("j"), ORD("j")
```

Funkce CHAR

Zobrazí daný znak zadaný jako ASCII číslo

SYNTAXE

```
SELECT CHAR(67,72,65,82);  
SELECT CHAR(226);  
>> problémy s nastaveným kodováním budeme se ještě bavit v sekci nastavení
```

Funkce BIN, HEX, OCT

Převod desítkové číslo na binární dvojkové, šestnáctkové

SYNTAXE

```
select pocet, BIN(pocet), plat1, BIN(Plat1) FROM funkce;  
SELECT BIN(255);
```

SYNTAXE

```
SELECT OCT(255);  
SELECT OCT(7), OCT(9)
```

SYNTAXE

```
SELECT HEX(255);
```

Funkce CHAR_LENGTH nebo CHARACTER_LENGTH()

vrátí délku (kolik znaků je) daného řetězce.

SYNTAXE

```
SELECT jmeno, CHAR_LENGTH(jmeno), PLAT1, CHAR_LENGTH(PLAT1) FROM funkce
```

>> podobná LENGHT

Nejsou tam mezery?

Funkce LENGTH

Počet znaků (bere i Text) vrací v bajtech

```
SELECT LENGTH(jmeno) FROM funkce  
SELECT jmeno, LENGTH(jmeno), PLAT1, LENGTH(PLAT1) FROM funkce
```

Rozdíl LENGTH a CHAR_LENGTH

- LENGTH() vrací v bajtech.
- CHAR_LENGTH() vrací ve znacích.

SYNTAXE

```
select length('E'), char_length('E');
```

Otázka? Co se vrátí?

Funkce CONCAT

Slučování řetězců ... V jiných SQL fungují dvě svislé čárky || v MySQL zatím ne

SYNTAXE

```
SELECT CONCAT(jmeno,prijmeni) FROM funkce  
SELECT CONCAT(jmeno," ",prijmeni) FROM funkce;
```

Funkce CONTACT_WS

Spojování řetězce s oddělovačem. (TEXTJOIN v Excel)

CONCAT_WS(separator, string1, string2, ...)

- separator oddělovač přidané mezi strunami, zatímco spojování.
- string1 první řetězec, který má být připojen.
- string2 Druhý řetězec, který má být připojen. Až N počet řetězců lze zadat tímto způsobem.

SYNTAXE

```
SELECT CONCAT_WS(" ",jmeno,prijmeni) FROM funkce;
```

Funkce ELT

vrací řetězec na požadované pozici ze seznamu argumentů.

ELT(index number, string1, string2, string3,...)

- index number celé číslo
- string1, string2, string3,... řetězce.

SYNTAXE

```
SELECT ELT(2,'Muni','je','super');  
SELECT ELT(5,'Muni','je','super');
```

- Proč druhý Select nefunguje?

Funkce FIELD

řekne pozici, kde je hledaný řetězec ... Opak od ELT,

FIELD(search string, string1, string2, string3.....)

- hledaný řetězec Řetězec, který se nachází v následujícím seznamu řetězců specifikovaných jako argumenty.
- string1 první řetězec, který má být zkontrolován, zda je obsahující první argument (tj. hledaný řetězec).
- string2 Druhý řetězec, který má být kontrolován, pokud je obsahuje první argument (tj. vyhledávání řetězec).

- string3 třetí řetězec, který má být kontrolován, pokud je obsahuje první argument (tj. vyhledávání řetězec). Až N počet řetězců lze zadat tímto způsobem.

SYNTAXE

```
SELECT FIELD('je', 'MUNI', 'je', 'super');
SELECT FIELD('neni', 'MUNI', 'je', 'super');
```

Co se stane pokud nenajde?

Funkce FIND_IN_SET

řekne pozici, kde je hledaný řetězec z více hodnot. Jen jiná syntaxe:

FIND_IN_SET (search string, string list)

- hledaný řetězec Řetězec, který se nachází v následujícím seznamu řetězců specifikovaných jako argumenty.
- stringList první seznam řetězců

SYNTAXE

```
SELECT FIND_IN_SET('je', 'muni, je, fakt, super, skola');
SELECT FIND_IN_SET('je', 'muni, je, je, fakt, super, skola')
```

Co když budou dva?

Funkce INSERT

Funkce vloží řetězec do existujícího, odstranění od znaku daný počet znaků a doplní mezery.

INSERT (ori_string, in_pos, length, new_string)

- ori_string původního řetězce.
- in_pos Pozice vložení v rámci původního řetězce.
- Délka Počet znaků, které mají být odstraněny z původního řetězce.
- new_string Řetězec má být vložen.

SYNTAXE

```
SELECT INSERT('MujOriginalniText', 4, 10, 'neoriginalni');
```

>> vylepšení

```
SELECT INSERT('MujOriginalniText', 4, 10, ' neoriginalni ') /* mezery */ ;
```

>> vylepšení kde využijete jaké texty můžete upravovat, co potřebujete za další údaje, když budete nahrazovat?

- Pozice znaků?

Funkce INSTR

Pozice hledaného řetězce.

INSTR (ori str, substr)

- ori str řetězec, který má být vyhledán.
- sub_str Řetězec který se má hledat v rámci ori_str.

SYNTAXE

```
SELECT INSTR('Abcde', 'c');
```

Funkce LOWER ... LCASE

Všechna malá

SYNTAXE

```
SELECT LOWER(jmeno) FROM funkce;
```

Funkce UPPER UCASE

Všechna velká

SYNTAXE

```
SELECT UPPER(jmeno) FROM funkce;
```

Úkol vytvoříte první velké písmeno a ostatní malí

Viz sloupec jméno – předpokládejte že někdo napíše JANA ...

Funkce LEFT / RIGHT

Počet znaku zprava/zleva

Sčítání

```
SELECT Left(jmeno,2) from funkce  
SELECT Right(jmeno,2) from funkce;
```

Další ukázky ... iniciály

```
SELECT Left(jmeno,1), Left(prijmeni,1) from funkce;
```

Spoj a mám iniciály

```
SELECT CONCAT(Left(jmeno,1), Left(prijmeni,1)) from funkce;
```

Úkol vytvoříte první velké písmeno a ostatní malí

Jak vytvořit Iniciály pokud je v buňce celé jméno a oddělovač je mezera? ...

Co vám chybí? Najít znak?

Funkce LOCATE

Na které pozici se nachází první znak (znaky) ... NAJÍT Excel

Můžu vložit od které pozice

SYNTAXE

```
select jmeno, LOCATE('a',jmeno) FROM funkce;  
SELECT LOCATE('a', 'abeceda'), LOCATE('a', 'abeceda', 2) ;
```

Co velká a malá písmena?

Funkce LPAD podobná RPAD

Doplň znaky řetězec jiným řetězcem do požadovaného počtu.

LPAD(str, len, padstr) RPAD(str, len, padstr)

- str Skutečné řetězec.
- len Číslo udávající celkový délku řetězce (ve znacích).
- padstr Řetězec, který se používá pro levé/pravé doplnění znaků do celkového počtu.

SYNTAXE

```
SELECT LPAD('Ahoj',10,'*'), LPAD('Hello',4,'*');
SELECT RPAD('Ahoj',10,'*'), RPAD('Hello',4,'*');
```

Kde využijete, se využívá? Banka? Textově čísta

Funkce LTRIM, RTRIM

odebere úvodní mezery (zleva)

odebere koncové mezery (zprva)

SYNTAXE

```
SELECT '  Ahoj', LTRIM('  Ahoj');
```

SYNTAXE

```
SELECT '  Ahoj  ', RTRIM('  Ahoj ');
```

Kde využijete?

Funkce MID nebo SUBSTR, SUBSTRING

Zobrazí požadované znaky

MID (str, pos, len)

- **stř** Řetězec, ze kterého budou vytaženy podřetězec.
- **pos** Celé číslo označující polohu (v rámci STR) od místa, kde těžba by měla začít.
- **len** Celé číslo udávající délku extrahovaného řetězce.

Najdi první a druhy

```
SELECT MID(jmeno,1,2) from funkce;
SELECT SUBSTR('ABECEDA',4,3), MID('ABECEDA',4,3);
```

Tip kde můžete využít? Iniciály?

Funkce POSITION

Pozice řetězce

POSITION (substr IN str)

- **substr** Řetězec, jehož pozice v jiném řetězci (STR) je možné načíst.
- **IN**
- **str** Řetězec, ve kterém má být vyvolána poloha dílčího (substr).

SYNTAXE

```
SELECT POSITION("a" IN "abeceda");
```

Funkce REPEAT

Opakovat řetězec

REPEAT (str, count)

- **str** Řetězec, který má být opakován.
- **count** Celé číslo udávající, kolikrát je řetězec str opakovat.

SYNTAXE

```
SELECT REPEAT('*',15);
```

Kde lze využít? Grafy?

Úkol – máte platy jak udělat graf kolik lidí vydělávají, chci více hvězdiček pro velké platy méně pro malé? Jak provedete? Něčím vydělíte? Zaokrouhlíte? Co potřebujete zjistíte Max? Min? Lze vypočíst rozdíl? Co vnořený SELECT?

Funkce REPLACE

nahrazuje všechny výskyty řetězce v rámci řetězce. DOSADIT v Excel.

REPLACE (str, find_string, replace_with)

- str řetězec.
- find_string Řetězec, který je přítomen jeden nebo vícekrát v řetězci str.
- replace_with A řetězec, který nahradí pokaždé, když zjistí, že find_string v ul.

SYNTAXE

```
SELECT REPLACE('abceda','a','xxx');
```

Kde využijete?

Funkce REVERSE

Funkce obrátí řetězec dodávaný jako argument

SYNTAXE

```
SELECT REVERSE('Abceda');
```

Funkce SPACE

Počet mezer

SYNTAXE

```
SELECT SPACE(10);
```

Funkce STRCMP

Porovnání řetězců

Strcmp (expr1, expr2)

- expr1 První řetězec pro srovnání.
- expr2 druhý řetězec pro srovnání.

Vrátí hodnotu

- 0 pokud oba řetězce jsou stejné a vrátí
- -1, když je první argument je menší než druhý v závislosti na určeném pořadí, a
- 1, když druhá je menší první.

SYNTAXE

```
SELECT STRCMP ('muj', 'muj');
SELECT STRCMP ('muj', 'můj1');
SELECT STRCMP ('muj', 'můj');
```

Tip: Vyzkoušejte na tabulkách o studentech, zkuste si s tabulkami pohrát, upravit pohlaví vše velkým písmenem. Doplnit sloupce jméno a příjmení – Vždy první písmeno velkým písmenem ostatní malým. To stejný pro město. K platu přidat pruhový graf, využít třeba * nebo jiný ASCII znak.

Propojte přes studium s tabulkou fakulty a i tam upravte názvy fakult, kde vše bude velkým písmenem

Tip: Podobný úkol může být na závěrečném testu.

Pomůcky:

Poznámky v SQL

/* začátek

*/ konec poznámky

```
SELECT actor_id /* název sloupce*/ AS pokus /* alias pro sloupec*/ From /*
poznámka tabulka */ actor /* aneb když so chce někdo psát poznámky do
SELECTU */
```

Na příště se můžete podívat na tabulky v databázi sakila:

<https://demo.phpmyadmin.net/master-config/>

Napadá vás co budete chtít o filmech vědět? Co budete chtít vyhledat?

Tip z COVID

Máte ve dvou sloupcích jméno a příjmení, Jména jsou s počátečním velkým písmenem, včetně háčku a čárek (co když je cizí jméno přehlasované a ...), potřebujete vytvořit emailovou adresu jmeno.prijmeni@nazev.skoly do nového sloupce (všechna písmena chcete malá – at není při importu pro EDU problém). Jak provedete?

Otázka: Co je vhodné zkontrolovat po vytvoření seznamu? (Tip Pepa Novák)

Jak budete řešit?

Můžete využít tabulku funkce a pro test si přidejte řádky (ať můžete otestovat i duplicitu dat):

```
insert into funkce values ('12', 'Janička', 'Kráťžovčková', 'F', '23500', '10000', '2', '2', '1954-12-22', '1954-12-22',
'Brno', 'A b', '0');
```

```
insert into funkce values ('13', 'Janička', 'Kráťžovčková', 'F', '23500', '10000', '2', '2', '1954-12-22', '1954-12-22',
'Brno', 'A b', '0');
```