

Základy príkazu Select

24. februára 2015

SELECT [ALL | DISTINCT]

... všetky riadky, alebo potlačiť duplicity

{ * | [stlpec_vyraz [[AS] alias_stlpca]][,..] }

... všetky stĺpce, alebo vybrané

FROM meno_tabulky [alias_tabulky] [,...]

... zoznam tabuliek, pohľadov..., z ktorých čerpáme dáta
... spojenia tabuliek

[**WHERE** podmienky]

... podmienky na vymedzenie riadkov

[**GROUP BY** zoznam_stlpcov]

... grupovanie riadkov pre agregáčné funkcie

[**HAVING** podmienky]

... podmienka s agregáčnými funkciami

[**ORDER BY** meno_stlpca [ASC | DESC] [,...]]

Výpis celej tabuľky

```
SQL> SELECT *
      2 FROM os_udaje;
```

| ROD_CISLO | MENO | PRIEZVISKO | ULICA | PSC | OBEC |
|-------------|-----------|-------------|--------------------|-------|----------------------|
| ----- | | | | | |
| 841106/3456 | Peter | Novak | Kamenna 27 | 97401 | Banska Bystrica |
| 840312/7845 | Stanislav | Steinmuller | Zelena 9 | 91501 | Nove Mesto nad Vahom |
| 860907/1259 | Janos | Toth | Slnečne namestie | 94501 | Komarno |
| 850130/3695 | Marek | Ratroch | Stred 49/7 | 01701 | Povazska Bystrica |
| 841201/1248 | Bohuslav | Biely | Juh 2100/456 | 91101 | Trencin |
| 830514/5341 | Branislav | Balaz | Tahanovce 38/12 | 04000 | Kosice |
| 781015/4431 | Peter | Kapustny | Javorova 2 | 01001 | Zilina |
| 800407/3522 | Marek | Durica | Precin 124 | 01701 | Precin |
| 791229/5431 | Martin | Kluciar | A. Bernolaka 14/20 | 01001 | Zilina |
| 871124/3578 | Lukas | Satrapa | Dolna 12 | 02201 | Cadca |
| 871203/5472 | Jan | Krnac | Prievoznicka | 03401 | Ruzomberok |
| 890310/2145 | Juraj | Papun | Kosicka cesta | 07101 | Michalovce |
| ----- | | | | | |
| ROD_CISLO | MENO | PRIEZVISKO | ULICA | PSC | OBEC |
| ----- | | | | | |
| 911001/3623 | Andrej | Janci | Tatranska 22 | 05801 | Poprad |
| 901130/4454 | Zdeno | Svetkovsky | Janka Borodaca 12 | 97101 | Prievidza |
| 921225/7452 | Rastislav | Kontros | Kolarovce 12 | 01401 | Kolarovce |
| 900913/3326 | Frantisek | Murgas | Namestie SNP 15 | 97401 | Banska Bystrica |
| 870913/3326 | Frantisek | Murgas | Fatranska 13 | 01008 | Zilina |
| 890608/4543 | Lubos | Lehotsky | Orlove 16 | 01701 | Orlove |

Výpis vybraných stĺpcov tabuľky

[1.1]

| os_udaje | | | |
|------------|--------------|----|------|
| rod_cislo | Char(11) | NN | (PK) |
| meno | Varchar2(15) | NN | |
| priezvisko | Varchar2(15) | NN | |
| ulica | Varchar2(20) | | |
| psc | Char(5) | | |
| obec | Varchar2(50) | | |

Výpis vybraných stĺpcov tabuľky

[1.1]

```
SELECT meno, priezvisko  
FROM os_udaje;
```

| os_udaje | | | |
|------------|--------------|----|------|
| rod_cislo | Char(11) | NN | (PK) |
| meno | Varchar2(15) | NN | |
| priezvisko | Varchar2(15) | NN | |
| ulica | Varchar2(20) | | |
| psc | Char(5) | | |
| obec | Varchar2(50) | | |

Výpis vybraných stĺpcov tabuľky

[1.1]

```
SELECT meno, priezvisko
FROM os_udaje;
```

| os_udaje | | |
|------------|--------------|---------|
| rod_cislo | Char(11) | NN (PK) |
| meno | Varchar2(15) | NN |
| priezvisko | Varchar2(15) | NN |
| ulica | Varchar2(20) | |
| psc | Char(5) | |
| obec | Varchar2(50) | |

```
SQL> SELECT meno, priezvisko
2 FROM os_udaje;
```

| MENO | PRIEZVISKO |
|-----------|-------------|
| ----- | ----- |
| Peter | Novak |
| Stanislav | Steinmuller |
| Janos | Toth |
| Marek | Ratroch |
| Bohuslav | Biely |
| Branislav | Balaz |
| Peter | Kapustny |
| Lukas | Satrapa |
| Jan | Krnac |
| Juraj | Papun |
| Andrej | Janci |
| Zdeno | Svetkovsky |
| Rastislav | Kontros |
| Frantisek | Murgas |
| Frantisek | Murgas |

Alias tabuľky a stĺpcov

| Princíp | Príklad |
|---|-----------------------|
| Ak nie je zavedený alias tabuľky | |
| nazov_stlpca | meno |
| nazov_tabulky.nazov_stlpca | os_udaje.meno |
| schema.nazov_tabulky.nazov_stlpca | vajsova.os_udaje.meno |
| Ak je zavedený alias tabuľky | |
| nazov_stlpca | meno |
| alias_tabulky.nazov_stlpca | ou.meno |

Bez aliasu tabulky

```
select meno  
  from os_udaje;
```

```
select os_udaje.meno  
  from os_udaje;
```

```
select vajsova.os_udaje.meno  
  from vajsova.os_udaje;
```


S aliasom tabulky

```
select meno  
  from os_udaje ou;
```

```
select ou.meno  
  from os_udaje ou;
```

```
select ou.meno  
  from vajsova.os_udaje ou;
```

Alias stĺpca

V časti Select za výrazom môžeme dopísať alias stĺpca, ktorý sa stane názvom stĺpca vo výsledku. Tento alias nie je možné použiť v inej časti selectu.

SELECT vyraz **AS** alias_stlpca

```
SQL> select os_cislo, rocnik + 1 AS novy_rocnik
      2 from student;
```

| OS_CISLO | NOVY_ROCNIK |
|----------|-------------|
| 501512 | 4 |
| 501469 | 3 |
| 501414 | 3 |
| 550545 | 2 |
| 550020 | 4 |
| 501567 | 1 |
| 501319 | 3 |

Alias stĺpca nie je možné použiť v iných častiach toho istého Selectu

```
1 select os_cislo, rocnik + 1 AS novy_rocnik
2 from student
3* where novy_rocnik = 1
SQL> /
where novy_rocnik = 1
      *
ERROR at line 3:
ORA-00904: "NOVY_ROCNIK": invalid identifier
```

Alias stĺpca a uloženie výsledku do tabuľky

```

1 create table pom_tab
2 AS
3 select os_cislo, rocnik + 1 AS novy_rocnik
4* from student
SQL> /

```

Table created.

```
SQL> desc pom_tab
```

| Name | Null? | Type |
|-------------|----------|------------|
| OS_CISLO | NOT NULL | NUMBER(38) |
| NOVY_ROCNIK | | NUMBER |

Alias tabuľky a stĺpcov - príklad

```
SELECT ou.rod_cislo AS rc , meno, priezvisko AS priezv
FROM os_udaje ou ;
```

| RC | MENO | PRIEZV |
|-------------|-----------|-------------|
| ----- | ----- | ----- |
| 841106/3456 | Peter | Novak |
| 840312/7845 | Stanislav | Steinmuller |
| 860907/1259 | Janos | Toth |
| 850130/3695 | Marek | Ratroch |
| 841201/1248 | Bohuslav | Biely |
| 830514/5341 | Branislav | Balaz |
| 781015/4431 | Peter | Kapustny |
| 800407/3522 | Marek | Durica |
| 791229/5431 | Martin | Kluciar |
| 871124/3578 | Lukas | Satrapa |
| 871203/5472 | Jan | Krnac |
| 890310/2145 | Juraj | Papun |
| 911001/3623 | Andrej | Janci |
| 901130/4454 | Zdeno | Svetkovsky |
| 921225/7452 | Rastislav | Kontros |
| 900913/3326 | Frantisek | Murgas |
| 870913/3326 | Frantisek | Murgas |
| 890608/4543 | Lubos | Lehotsky |
| 860103/2238 | Rudolf | Kovac |
| 896123/5471 | Zuzana | Slamova |
| 855122/8569 | Erika | Lipovska |
| 830914/7748 | Robert | Mazur |
| 840410/6777 | Alojz | Gazo |
| 840409/7900 | Zoltan | Sim |
| 830420/8088 | Zdenko | Olzbut |
| 830301/7789 | Martin | Skuta |
| 831002/8463 | Jozef | Brna |

Potlačenie duplicít

```
SELECT DISTINCT zoznam_stlpcov
FROM ....
```

```
1 select st_skupina
2   from student
3* order by st_skupina
SQL> /
```

```
ST_SKU
```

```
-----
```

```
5PI011
```

```
5PI021
```

```
5PI021
```

```
5ZI000
```

```
5ZI000
```

```
5ZI000
```

```
5ZI012
```

```
5ZI021
```

```
5ZI030
```

```
5ZI031
```

```
5ZIA21
```

```
5ZM021
```

```
5ZM021
```

```
5ZM023
```

```
5ZM031
```

```
...
```

Potlačenie duplicít

```
SELECT DISTINCT zoznam_stlpcov
FROM ....
```

```
1 select st_skupina
2   from student
3* order by st_skupina
SQL> /
```

ST_SKU

5PI011

5PI021

5PI021

5ZI000

5ZI000

5ZI000

5ZI012

5ZI021

5ZI030

5ZI031

5ZIA21

5ZM021

5ZM021

5ZM023

5ZM031

...

```
1 select DISTINCT st_skupina
2   from student
3* order by st_skupina
SQL> /
```

ST_SKU

5PI011

5PI021

5ZI000

5ZI012

5ZI021

5ZI030

5ZI031

5ZIA21

5ZM021

5ZM023

5ZM031

5ZN021

5ZP012

5ZP021

5ZR013

...

Základné typy spojení

```
tabulka_A {LEFT | RIGHT | FULL } JOIN tabulka_B  
  { ON ( podmienky_spojenia)  
    |  
    USING( zoznam_stlpcov)  
  }
```

- **USING** - je možné použiť len v prípade, že atribúty primárneho a cudzieho kľúča majú rovnaké meno. Pri stĺpcoch použitých v klauzule USING nie je možné používať alias tabuľky.
- **ON** - je možné využiť vždy. V tomto prípade je nutné používať alias tabuľky.

USING - spojenia

```
SELECT rod_cislo , st.os_cislo, ou.meno, ou.priezvisko  
FROM os_udaje ou JOIN student st USING ( rod_cislo );
```

USING - spojenia

```
SELECT rod_cislo , st.os_cislo, ou.meno, ou.priezvisko  
FROM os_udaje ou JOIN student st USING ( rod_cislo ) ;
```

```
SELECT rod_cislo , st.os_cislo, ou.meno, ou.priezvisko,  
sto.popis_odboru, sto.popis_zamerania  
FROM os_udaje ou JOIN student st USING ( rod_cislo )  
JOIN st_odbory sto USING ( st_odbor, st_zameranie ) ;
```

ON - spojenia

```
SELECT ou.rod_cislo, st.os_cislo, ou.meno, ou.priezvisko  
FROM os_udaje ou JOIN student st  
ON ( ou.rod_cislo = st.rod_cislo) ;
```

ON - spojenia

```
SELECT ou.rod_cislo, st.os_cislo, ou.meno, ou.priezvisko  
FROM os_udaje ou JOIN student st  
    ON ( ou.rod_cislo = st.rod_cislo) ;
```

```
SELECT ou.rod_cislo, st.os_cislo, ou.meno, ou.priezvisko,  
    sto.popis_odboru, sto.popis_zamerania  
FROM os_udaje ou JOIN student st ON ( ou.rod_cislo = st.rod_cislo )  
    JOIN st_odbory sto ON ( st.st_odbor = sto.st_odbor  
        AND st.st_zameranie = sto.st_zameranie) ;
```

Porovnanie USING a ON

- USING

```
SELECT rod_cislo , st.os_cislo, ou.meno, ou.priezvisko  
FROM os_udaje ou JOIN student st USING ( rod_cislo ) ;
```

- ON

```
SELECT ou.rod_cislo , st.os_cislo, ou.meno, ou.priezvisko  
FROM os_udaje ou JOIN student st  
ON ( ou.rod_cislo = st.rod_cislo) ;
```

Triedenie

```
SELECT [DISTINCT] { * | [ stlpec_vyraz [ [AS] alias_stlpca] ][,..]
FROM  meno_tabulky [alias_tabulky]  [...] [JOIN ....]
[ WHERE podmienky ]

[ GROUP BY zoznam_stlpcov ]
[ HAVING podmienky ]

[ ORDER BY  meno_stlpca [ ASC | DESC ] [...] ]
```

... triedenie výpisu

Triedenie - príklad

```
SELECT meno, priezvisko, rocnik
FROM os_udaje ou JOIN student st USING (rod_cislo)
```

- ORDER BY priezvisko, meno
- ORDER BY priezvisko ASC, meno ASC
- ORDER BY rocnik DESC, priezvisko, meno

| MENO | PRIEZVISKO | R |
|-----------|------------|---|
| ----- | ----- | - |
| Branislav | Balaz | 0 |
| Branislav | Balaz | 2 |
| Jozef | Brna | 0 |
| Dusan | Bucany | 2 |
| Jaroslav | Cipak | 2 |
| Marek | Durica | 2 |
| Marek | Durica | 1 |
| Alojz | Gazo | 2 |
| Miroslav | Gmuca | 2 |
| Andrej | Janci | 1 |
| Peter | Kapustny | 3 |
| Martin | Kluciar | 0 |
| Martin | Kluciar | 0 |
| Martin | Kluciar | 1 |
| Lubomir | Kominek | 2 |
| Rastislav | Kontros | 2 |
| Peter | Korenciak | 0 |
| Rudolf | Kovac | 1 |
| ... | | |

Triedenie - príklad

```
SELECT meno, priezvisko, rocnik
FROM os_udaje ou JOIN student st USING (rod_cislo)
```

- ORDER BY priezvisko, meno
- ORDER BY priezvisko ASC, meno ASC
- ORDER BY rocnik DESC, priezvisko, meno

| MENO | PRIEZVISKO | R |
|-----------|------------|---|
| ----- | ----- | - |
| Branislav | Balaz | 0 |
| Branislav | Balaz | 2 |
| Jozef | Brna | 0 |
| Dusan | Bucany | 2 |
| Jaroslav | Cipak | 2 |
| Marek | Durica | 2 |
| Marek | Durica | 1 |
| Alojz | Gazo | 2 |
| Miroslav | Gmuca | 2 |
| Andrej | Janci | 1 |
| Peter | Kapustny | 3 |
| Martin | Kluciar | 0 |
| Martin | Kluciar | 0 |
| Martin | Kluciar | 1 |
| Lubomir | Kominek | 2 |
| Rastislav | Kontros | 2 |
| Peter | Korenciak | 0 |
| Rudolf | Kovac | 1 |
| ... | | |

Triedenie - príklad

```
SELECT meno, priezvisko, rocnik
FROM os_udaje ou JOIN student st USING (rod_cislo)
```

- ORDER BY priezvisko, meno
- ORDER BY priezvisko ASC, meno ASC
- ORDER BY rocnik DESC, priezvisko, meno

| MENO | PRIEZVISKO | R |
|-----------|-------------|---|
| ----- | ----- | - |
| Peter | Kapustny | 3 |
| Jan | Krnac | 3 |
| Peter | Novak | 3 |
| Branislav | Balaz | 2 |
| Dusan | Bucany | 2 |
| Jaroslav | Cipak | 2 |
| Marek | Durica | 2 |
| Alojz | Gazo | 2 |
| Miroslav | Gmuca | 2 |
| Lubomir | Kominek | 2 |
| Rastislav | Kontros | 2 |
| Lubos | Lehotsky | 2 |
| Erika | Lipovska | 2 |
| Peter | Minarik | 2 |
| Frantisek | Murgas | 2 |
| Zdenko | Olzbut | 2 |
| Juraj | Papun | 2 |
| Zoltan | Sim | 2 |
| Stanislav | Steinmuller | 2 |

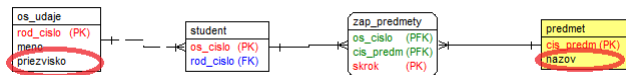
Podmienky

- vyraz1

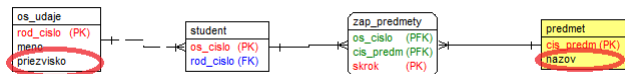
| |
|----|
| > |
| >= |
| = |
| <> |
| != |
| < |
| <= |

 vyraz2
- vyraz1 **BETWEEN** min_hodnota **AND** max_hodnota
- vyraz **IS** [NOT] **NULL**
- vyraz [NOT] **IN** (mnozina)
- [NOT] **EXISTS** (mnozina)
- podmienka1 **AND** podmienka2 | podmienka1 **OR** podmienka2

Vypíšte názvy predmetov študenta s priezviskom 'Balaz'.

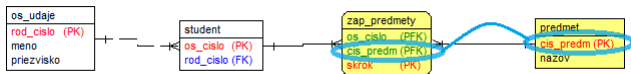


Vypíšte názvy predmetov študenta s priezviskom 'Balaz'.



```
SELECT pr.nazov
FROM predmet pr
```

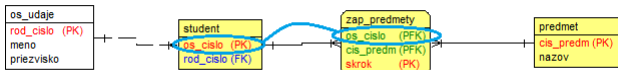
Vypíšte názvy predmetov študenta s priezviskom 'Balaz'.



```

SELECT pr.nazov
FROM   predmet pr JOIN zap_predmety zp USING ( cis_predm )
  
```

Vypíšte názvy predmetov študenta s priezviskom 'Balaz'.



```

SELECT pr.nazov
FROM   predmet pr JOIN zap_predmety zp USING ( cis_predm )
      JOIN student st USING ( os_cislo )
  
```

Vypíšte názvy predmetov študenta s priezviskom 'Balaz'.



```

SELECT pr.nazov
FROM   predmet pr JOIN zap_predmety zp USING ( cis_predm )
      JOIN student st USING ( os_cislo )
      JOIN os_udaje ou USING ( rod_cislo )
WHERE  ou.priezvisko = 'Balaz';
  
```

Štandardné funkcie

- Podľa dátových typov: TO_CHAR, TO_DATE, ROUND, SUBSTR, LENGTH, ...
- Operátor LIKE
- podmienky sú vo WHERE

Agregačné funkcie

- COUNT, SUM, MIN, MAX, AVG
- podmienky sú v HAVING
- spojené väčšinou s GROUP BY

Špeciálne funkcie

DECODE - prekódovanie hodnôt

DECODE (vyraz, IF-hodnota, THEN-hodnota, [,...] , ELSE-hodnota)

```
SQL> select os_cislo
1      , decode ( st.stav, 'S', 'student', 'K','ukonceny', 'P', 'preruseny', 'vyluceny') as stav
2 from student st;
```

OS_CISLO STAV

```
-----
501512 student
501469 student
501414 student
550545 preruseny
550020 student
501567 vyluceny
501319 student
501555 student
501402 student
501096 vyluceny
501103 ukonceny
501333 student
...
```

Funkcie pre číselné dátové typy

ROUND - zaokrúhlenie

```
ROUND( cislo, [ pocet_des_miest] )
```

TRUNC - useknutie podľa parametra

```
TRUNC( cislo, [presnost])
```

MOD - zvyšok po delení

```
MOD( delenec, delitel)
```

ROUND - zaokrúhlenie

```
ROUND( cislo, [ pocet_des_miest] )
```

```
SQL> select round(2.5), round(21.3), round(13.6) from dual;
```

```
ROUND(2.5) ROUND(21.3) ROUND(13.6)
```

```
-----  
              3              21              14
```

```
SQL> select round(13.2458878, 2 ) from dual;
```

```
ROUND(13.2458878,2)
```

```
-----  
              13.25
```

```
SQL> select round(657981.124, -2) from dual;
```

```
ROUND(657981.124,-2)
```

```
-----  
              658000
```

TRUNC - useknutie podľa parametra

```
TRUNC( cislo, [presnost])
```

```
SQL> select trunc( 153.3 ), trunc (169.9)
       2 from dual;
```

```
TRUNC(153.3) TRUNC(169.9)
```

```
-----
              153              169
```

```
SQL> select trunc( 15.26589, 1), trunc (15.26589, - 1 )
       2 from dual;
```

```
TRUNC(15.26589,1) TRUNC(15.26589,-1)
```

```
-----
              15.2              10
```

MOD - zvyšok po delení

MOD(delenec, delitel)

```
SQL> select mod(5,2) ,mod(16,4) from dual;
```

| MOD(5,2) | MOD(16,4) |
|----------|-----------|
| 1 | 0 |

Funkcie pre reťazcové dátové typy

SUBSTR - podreťazec

SUBSTR(reťazca, pozícia, dĺžka)

LENGTH - dĺžka reťazca

LENGTH(reťazec)

CONCAT, operátor || - spojenie reťazcov

CONCAT(reťazec1, reťazec2)
reťazec1 || reťazec2 [|| reťazec3 ...]

Operátor LIKE

stlpec LIKE vzor

UPPER - prevod na veľké písmo

UPPER(reťazec)

LOWER - prevod na malé písmo

LOWER(reťazec)

SUBSTR - podreťazec

SUBSTR(reťazca, pozícia, dĺžka)

kde

- **pozícia** začína od 1.
- V prípade zápornej pozície, pozícia je odrátaná od konca reťazca.
- Ak nie je udaná dĺžka reťazca, vypíše sa až do konca reťazca.

```
SQL> select substr( '5ZI021', 2, 1 ) as pracovisko  
2* from dual;
```

P
-
Z

```
SQL> select st_skupina, substr(st_skupina, 5,2) from student;
```

```
ST_SKU SUBSTR(S
```

```
-----
```

```
5ZI031 31
```

```
5ZP021 21
```

```
5ZI021 21
```

```
5ZI012 12
```

```
5ZM031 31
```

```
5ZI000 00
```

```
SQL> select st_skupina, substr(st_skupina, -2) from student
```

```
ST_SKU SUBSTR(S
```

```
-----
```

```
5ZI031 31
```

```
5ZP021 21
```

```
5ZI021 21
```

```
5ZI012 12
```

```
5ZM031 31
```

```
5ZI000 00
```


*LENGTH - dĺžka reťazca***LENGTH(reťazec)**

```
SQL> select priezvisko , length(priezvisko) from os_udaje;
```

| PRIEZVISKO | LENGTH(PRIEZVISKO) |
|-------------|--------------------|
| ----- | ----- |
| Novak | 5 |
| Steinmuller | 11 |
| Toth | 4 |
| Ratroch | 7 |
| Biely | 5 |
| Balaz | 5 |
| Kapustny | 8 |

CONCAT, operátor || - spojenie reťazcov

CONCAT(ret'azec1, ret'azec2)

```
SQL> select concat( meno, priezvisko )  
      2 from os_udaje;
```

```
CONCAT(MENO,PRIEZVISKO)
```

```
-----  
PeterNovak  
StanislavSteinmuller  
JanosToth  
MarekRatroch  
BohuslavBiely  
BranislavBalaz  
PeterKapustny  
MarekDurica  
MartinKluciar  
LukasSatrapa  
JanKrnac  
JurajPapun
```

```
SQL> select concat(concat(meno, ' ') , priezvisko ) as cele_meno  
2 from os_udaje;
```

CELE_MENO

Peter Novak
Stanislav Steinmuller
Janos Toth
Marek Ratroch
Bohuslav Biely
Branislav Balaz
Peter Kapustny
Marek Durica
Martin Kluciar
Lukas Satrapa
Jan Krnac
Juraj Papun

Reťazce je možné spojiť nielen pomocou funkcie CONCAT, ale aj pomocou operátora ||.

```
ret'azec1 || ret'azec2 [ || retazec3 ...]
```

```
SQL> select  meno || ' ' || priezvisko as cele_meno
          2  from os_udaje;
```

```
CELE_MENO
```

```
-----
Peter Novak
Stanislav Steinmuller
Janos Toth
Marek Ratroch
Bohuslav Biely
Branislav Balaz
Peter Kapustny
Marek Durica
Martin Kluciar
Lukas Satrapa
Jan Krnac
Juraj Papun
```

*UPPER - prevod na veľké písmo***UPPER(ret'azec)**

```
SQL> select priezvisko, upper(priezvisko) from os_udaje;
```

| PRIEZVISKO | UPPER(PRIEZVISK |
|-------------|-----------------|
| ----- | ----- |
| Novak | NOVAK |
| Steinmuller | STEINMULLER |
| Toth | TOTH |
| Ratroch | RATROCH |
| Biely | BIELY |
| Balaz | BALAZ |
| Kapustny | KAPUSTNY |
| Durica | DURICA |
| Kluciar | KLUCIAR |
| Satrapa | SATRAPA |

LOWER - prevod na malé písmo

LOWER(ret'azec)

```
SQL> select priezvisko, lower(priezvisko) from os_udaje;
```

| PRIEZVISKO | LOWER(PRIEZVISK |
|-------------|-----------------|
| ----- | ----- |
| Novak | novak |
| Steinmuller | steinmuller |
| Toth | toth |
| Ratroch | ratroch |
| Biely | biely |
| Balaz | balaz |
| Kapustny | kapustny |
| Durica | durica |
| Kluciar | kluciar |

Operátor *LIKE*

stĺpec **LIKE** vzor

| Vzor | Význam |
|------|------------------------|
| % | ľubovoľný počet znakov |
| _ | jeden znak |

```
select meno, priezvisko
from os_udaje
where priezvisko LIKE 'K%';
```

| MENO | PRIEZVISKO |
|-----------|------------|
| Peter | Kapustny |
| Martin | Kluciar |
| Jan | Krnac |
| Rastislav | Kontros |
| Rudolf | Kovac |
| Lubomir | Kominek |
| Peter | Korenciak |

```
-- znak 3 na ľubovoľnej pozícii
SQL> select distinct st_skupina
      2   from student
      3  where st_skupina like '%3%';
```

```
ST_SKU
-----
5ZI031
5ZI030
5ZM023
5ZSN23
5ZM031
5ZR032
5ZR013
```

```
-- znak 3 po štyroch ľubovoľných znakoch
SQL> select distinct st_skupina
      2   from student
      3  where st_skupina like '___3%';
```

```
ST_SKU
-----
5ZI031
5ZI030
5ZM031
5ZR032
```


Funkcie pre dátumové dátové typy

SYSDATE - aktuálny dátum a čas

SYSDATE

TO_DATE - prevod z reťazca na dátum

TO_DATE(reťazec, formát)

TO_CHAR - prevod z dátumu na reťazec

TO_CHAR(dátum, formát)

ADD_MONTHS - pridaj k dátumu zadaný počet mesiacov

ADD_MONTHS(dátum, pocet_mesiacov)

LAST_DAY - posledný deň v mesiaci

LAST_DAY(dátum)

TRUNC - oseknutie dátumu podľa formátu

TRUNC (dátum, [format])

***SYSDATE** - aktuálny dátum a čas*

SYSDATE

```
SQL> select sysdate from dual;
```

```
SYSDATE
```

```
-----
```

```
10-FEB-15
```

TO_CHAR - prevod dátumu na reťazec

TO_CHAR(datum, format)

```
SQL> select to_char( sysdate, 'DD.MM.YYYY') from dual;
```

```
TO_CHAR(SY
-----
10.02.2015
```

```
SQL> select to_char(sysdate, 'DD.MM.YYYY HH24:MI:SS') as datum
       2 from dual;
```

```
DATUM
-----
10.02.2015 14:12:02
```

```
SQL> select sysdate, to_char( sysdate, 'D') , to_char( sysdate, 'DD')
       2 from dual;
```

```
SYSDATE    T TO
-----  -  --
10-FEB-15 3 10
```

| Formát | Význam |
|--------|-----------------------------|
| D | Poradové číslo dňa v týždni |
| DD | Číslo dňa v mesiaci |
| MM | Číslo mesiaca |
| YYYY | Rok - štyri číslice |
| RRRR | Rok - štyri číslice |
| HH | Hodiny - 12 hodinový formát |
| HH24 | Hodiny - 24 hodinový formát |
| MI | Minúty |
| SS | Sekundy |



TO_DATE - prevod z reťazca na dátum

TO_DATE(reťazec, formát)

```
SQL> select to_date ( '31.12.2015', 'DD.MM.YYYY') from dual;
```

```
TO_DATE('
-----
31-DEC-15
```

ADD_MONTHS - pridaj k dátumu zadaný počet mesiacov

ADD_MONTHS(dátum, pocet_mesiacov)

```
SQL> select sysdate, add_months( sysdate, 2), add_months(sysdate, -2) from dual;
```

| SYSDATE | ADD_MONTH | ADD_MONTH |
|-----------|-----------|-----------|
| ----- | ----- | ----- |
| 10-FEB-15 | 10-APR-15 | 10-DEC-14 |

LAST_DAY - posledný deň v mesiaci

LAST_DAY(dátum)

```
SQL> select sysdate, last_day( sysdate ) from dual;
```

```
SYSDATE    LAST_DAY(  
-----  
10-FEB-15  28-FEB-15
```

```
SQL> select last_day( to_date( '15.5.2015', 'DD.MM.YYYY' ) ) from dual;
```

```
LAST_DAY(  
-----  
31-MAY-15
```

TRUNC - useknutie dátumu

TRUNC (datum, [format])

```

1 select to_char( sysdate , 'DD.MM.YYYY HH24:MI:SS') as d1,
2         to_char( trunc(sysdate) , 'DD.MM.YYYY HH24:MI:SS') as d2,
3         to_char( trunc(sysdate, 'DD') , 'DD.MM.YYYY HH24:MI:SS') as d3
4* from dual
SQL> /

```

| D1 | D2 | D3 |
|---------------------|---------------------|---------------------|
| 10.02.2015 13:16:34 | 10.02.2015 00:00:00 | 10.02.2015 00:00:00 |

```

1 select to_char( sysdate , 'DD.MM.YYYY HH24:MI:SS') as d1,
2         to_char( trunc(sysdate, 'MI') , 'DD.MM.YYYY HH24:MI:SS') as d4,
3         to_char( trunc(sysdate, 'YYYY') , 'DD.MM.YYYY HH24:MI:SS') as d5
4* from dual
SQL> /

```

| D1 | D4 | D5 |
|---------------------|---------------------|---------------------|
| 10.02.2015 13:17:17 | 10.02.2015 13:17:00 | 01.01.2015 00:00:00 |

EXTRACT - získanie časti dátumu

EXTRACT (co FROM datum)

```
1* select sysdate, extract ( month from sysdate ) from dual
SQL> /
```

| SYSDATE | EXTRACT(MONTHFROMSYSDATE) |
|-----------|---------------------------|
| ----- | ----- |
| 10-FEB-15 | 2 |

Aritmetika s dátumami

K dátumu je možné pripočítať / odpočítať počet dní a výsledok bude typu DATE.

```
1* select sysdate, sysdate+1 from dual
SQL> /
```

| SYSDATE | SYSDATE+1 |
|-----------|-----------|
| 10-FEB-15 | 11-FEB-15 |

Rozdiel medzi dvoma dátumami je počet dní.

```
1 select sysdate - to_date('03.05.2014', 'DD.MM.YYYY') as rozdiel1,
2         trunc( sysdate ) - to_date('03.05.2014', 'DD.MM.YYYY') as rozdiel2
3* from dual
SQL> /
```

| ROZDIEL1 | ROZDIEL2 |
|------------|----------|
| 283.469815 | 283 |

Podmienky s NULL hodnotami

vyraz IS [NOT] NULL

```
SQL> select os_cislo, meno, priezvisko
  2  from os_udaje ou JOIN student st USING ( rod_cislo )
  3  where ukoncenie IS NULL ;
```

| OS_CISLO | MENO | PRIEZVISKO |
|----------|-----------|-------------|
| 550020 | Peter | Kapustny |
| 501333 | Martin | Kluciar |
| 501402 | Marek | Durica |
| 501555 | Marek | Durica |
| 501319 | Branislav | Balaz |
| 501469 | Stanislav | Steinmuller |

```
1  select os_cislo, meno, priezvisko
2  from os_udaje ou JOIN student st USING ( rod_cislo )
3* where ukoncenie IS NOT NULL
SQL> /
```

| OS_CISLO | MENO | PRIEZVISKO |
|----------|-----------|------------|
| 501103 | Martin | Kluciar |
| 501096 | Martin | Kluciar |
| 500438 | Miroslav | Gmuca |
| 500433 | Juraj | Tuma |
| 500422 | Martin | Skuta |
| 500428 | Peter | Stelbasky |
| 500432 | Zdenko | Olzbut |
| 501567 | Branislav | Balaz |
| 500429 | Peter | Minarik |

S NULL hodnotami nikdy nepoužívajte relačné operátory, pretože výsledkom bude vždy FALSE.

```
1 select os_cislo, meno, priezvisko
2 from os_udaje ou JOIN student st USING ( rod_cislo )
3* where ukoncenie = NULL
SQL> /
```

no rows selected

```
1 select os_cislo, meno, priezvisko
2 from os_udaje ou JOIN student st USING ( rod_cislo )
3* where ukoncenie <> NULL
SQL> /
```

no rows selected

Funkcia NVL - nahradenie null hodnoty

```
SQL> select vysledok, nvl(vysledok, 'F')  
2 from zap_predmety;
```

```
V N  
- -  
C C  
F  
F  
A A  
C C  
D D  
F  
F  
A A
```

COALESCE- nahradenie null hodnoty

`coalesce (vyraz1, vyraz2 [,...])`

```
SQL> select vysledok, coalesce(vysledok, 'F') from zap_predmety;
```

```
V C
- -
C C
  F
  F
A A
C C
D D
  F
  F
A A
```

vyraz [NOT] IN (mnozina)
alebo

[NOT] EXISTS (vymenovana_mnozina)

- Vymenovaná množina
- Množina udaná pomocou vnoreného selectu.

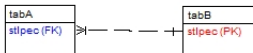
Vymenovaná množina

```
select meno, priezvisko  
from os_udaje JOIN student USING (rod_cislo)  
where rocnik IN ( 1, 2);
```

```
select meno, priezvisko  
from os_udaje JOIN student USING (rod_cislo)  
where rocnik NOT IN ( 1, 2);
```

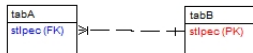
Množina udaná pomocou vnoreného selectu - Jednoduchý primárny kľúč.

[1,1]



alebo

[1,1]



```
SELECT ....  
FROM tabA  
WHERE tabA.stlpec [NOT] IN  
      ( select tabB.stlpec  
        FROM tabB  
      )
```

```
SELECT ....  
FROM tabA  
WHERE [ NOT ] EXISTS  
      ( select 'x'  
        FROM tabB  
        WHERE tabA.stlpec = tabB.stlpec  
      )
```

Množina udaná pomocou vnoreného selectu.

- IN

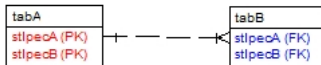
```
select ou.meno, ou.priezvisko
  from os_udaje ou JOIN student st USING (rod_cislo)
 where st.os_cislo NOT IN
       ( select zp.os_cislo
         from zap_predmety zp
         where zp.cis_predm = 'BI06'
       );
```

- EXISTS

```
select ou.meno, ou.priezvisko
  from os_udaje ou JOIN student st USING (rod_cislo)
 where NOT EXISTS
       ( select 'x'
         from zap_predmety zp
         where zp.cis_predm = 'BI06'
           and zp.os_cislo = st.os_cislo
       );
```

Množina udaná pomocou vnoreného selectu - Kompozitný primárny kľúč.

[1,1]



```
SELECT ....
FROM tabA
WHERE tabA.stlpecA [NOT] IN
    ( select tabB.stlpecA
      FROM tabB
    )
AND tabA.stlpecB [NOT] IN
    ( select tabB.stlpecB
      FROM tabB
    )
```

```
SELECT ....
FROM tabA
WHERE [ NOT ] EXISTS
    ( select 'x'
      FROM tabB
      WHERE tabA.stlpecA = tabB.stlpecA
        AND tabA.stlpecB = tabB.stlpecB
    )
```

Min - minimum

Min(vyraz)

Max - maximum

Max(vyraz)

Sum - súčet

Sum(vyraz)

Avg - priemer

Avg(vyraz)

Count - počet riadkov

Count(* | [DISTINCT] vyraz)

Min - minimum

Min(vyraz)

```
SQL> select min(os_cislo) from student;
```

```
MIN(OS_CISLO)
```

```
-----
```

```
500422
```

Max - maximum

Max(vyraz)

```
SQL> select max(os_cislo) from student;
```

```
MAX(OS_CISLO)
```

```
-----
```

```
550945
```

Sum - súčet

Sum(vyraz)

```
SQL> select sum(ects)
      2  from zap_predmety
      3  where os_cislo = 501103
      4  ;
```

SUM(ECTS)

19

Count - počet riadkov

- Počet všetkých riadkov skupiny

Count(*)

```
SQL> select count(*) from zap_predmety;
```

```
COUNT(*)
-----
      484
```

- Počet riadkov skupiny, kde stlpec nemá NULL hodnotu.

Count(stlpec)

```
SQL> select count(vysledok) from zap_predmety;
```

```
COUNT(VYSLEDOK)
-----
      295
```

- Počet unikátnych hodnôt stlpca v skupine.

Count(DISTINCT stlpec)

```
SQL> select count(DISTINCT vysledok) from zap_predmety;
```

```
COUNT(DISTINCTVYSLEDOK)
-----
```