

Korisničke definisane funkcije

- Sačuvana funkcija (*Stored function*) je poseban tip sačuvanog programa koji **vraća samo jednu vrednost**.
- Kreirane funkcije se obično koriste da enkapsuliraju (zatvore) opšte formule ili pravila u poslovanju koja se često pozivaju kroz SQL iskaze, najčešće u select iskazu
- U funkcijama ne mogu da se koriste kursori, transakcije, ažuriranje tabela i pisanje agregatnih funkcija kao što je npr. sum funkcija
- Funkcija sadrži samo ulazne (in) parametre, ne sadrži out parametre što je slučaj sa procedurama

```
DELIMITER $$  
  
CREATE FUNCTION function_name( param1, param2,... ) RETURNS datatype [NOT] DETERMINISTIC  
  
BEGIN  
  
  -- statements  
  
END $$  
  
DELIMITER ;
```

Struktura funkcije

- Prvi korak je definsanje naziva za funkciju
- Drugi korak je lista ulaznih parametara
- Treći korak je tip podataka za vrednost koju funkcija vraća
- Četvri korak je izbor determinističke funkcije koja uvek vraća isti rezlutat za iste ulazne parametre dok nedeteministička (default) funkcija vraća različit rezlutat za iste ulazne parametre
- Peti korak je logika funkcije koja mora da sadrži RETURN iskaz unutar BEGIN i END bloka
 - Return iskaz vraća vrednost i izlazi se iz funkcije

```
DELIMITER $$  
  
CREATE FUNCTION function_name( param1, param2,... ) RETURNS datatype [NOT] DETERMINISTIC  
  
BEGIN  
  
-- statements  
  
END $$  
  
DELIMITER ;
```

Struktura funkcije



SQL

```
DELIMITER $$
```

```
CREATE FUNCTION function_name(parameter data_type) RETURNS data_type
```

```
DECLARE variable_name data_type
```



```
BEGIN
```

```
    SELECT ...
```

```
    RETURN variable_name
```

```
END$$
```

```
DELIMITER ;
```

here you have no OUT parameters to define between the parentheses after the object's name

all parameters are IN, and since this is well known, you need not explicitly indicate it with the word, 'IN'



SQL

```
DELIMITER $$
```

```
CREATE FUNCTION function_name(parameter data_type) RETURNS data_type
```

```
DECLARE variable_name data_type
```



```
BEGIN
```

```
    SELECT ...
```

```
    RETURN variable_name
```

```
END$$
```

```
DELIMITER ;
```

although there are no OUT parameters, there is a 'return value'

it is obtained after running the query contained in the body of the function

Primer korisničke funkcije

- Kreirati funkciju za bazu **ig_clone** koja će na osnovu username korisnika da vrati broj slika koje je taj korisnik postavio

```
select username, broj_slika(username)
from users;
```

```
delimiter $$
create function broj_slika(v_username varchar(40))
returns integer
deterministic
begin
declare v_broj_slika integer;
select count(*) into v_broj_slika
from users inner join photos on users.id=photos.user_id
where username=v_username;
return v_broj_slika;
end$$
delimiter ;
```

Primer korisničke funkcije

- Kreirati funkciju koja će na osnovu unete vrednosti za cenu da vrati jednu od tri vrednosti
 - Jeftino ukoliko je cena manja od 3\$
 - Povoljno ukoliko je cena između 3\$ i 8\$
 - Skupo ukoliko je cena veća od 8\$

```
SHOW FUNCTION STATUS
```

```
WHERE db = 'online_shop';
```

```
select name, price, opis_cene(price)  
from products;
```

```
delimiter $$
```

```
create function opis_cene(price decimal(10,2))
```

```
returns varchar(20)
```

```
deterministic
```

```
begin
```

```
declare opis varchar(20);
```

```
if price < 3
```

```
then set opis = 'Jeftino';
```

```
Else if price < 7
```

```
then set opis = 'Povoljno';
```

```
Else set opis='Skupo';
```

```
end if;
```

```
return (opis);
```

```
end$$
```

```
delimiter ;
```