

Procedure

- Shranjeni podprogrami, ki jih lahko kličemo
- Parametre lahko določimo kot vhodne (IN), izhodne (OUT) ali vhodno-izhodne (IN OUT)
- Kličemo jih s `CALL Ime_Procedure(...)`

Primer procedure

Napišite shranjeno proceduro, ki vrne vse podatke o čolnih z dolžino med "spodnja" in "zgornja" meja.

```
DELIMITER //
CREATE PROCEDURE colni_razpon (
    IN spodnja INTEGER,
    IN zgornja INTEGER)
BEGIN
    -- Pri proceduri se SELECT stavki izpišejo
    SELECT *
    FROM coln
    WHERE dolzina BETWEEN spodnja AND zgornja;
END //
DELIMITER ;
-- Klic procedure
CALL colni_razpon(20,40);
```

Funkcije

- Podobno kot procedure, le-da privzeto vračajo eno vrednost
- Vsi parametri so privzeto IN
- Uporaba `SELECT Ime_Funkcije(...)`

Primer funkcije

Napišite funkcijo, ki vrne število čolnov z dolžino med "spodnja" in "zgornja" meja.

```
DELIMITER //
```

```
CREATE FUNCTION colni_razpon_fun (  
    spodnja INTEGER,  
    zgornja INTEGER) RETURNS INTEGER
```

```
BEGIN  
    DECLARE X INTEGER; -- Deklaracija spremenljivke  
    SELECT COUNT(*) INTO X -- INTO shrani vrednost v X  
    FROM coln  
    WHERE dolzina BETWEEN spodnja AND zgornja;  
    RETURN X; -- Funkcija mora imeti RETURN  
END //
```

```
DELIMITER ;  
SELECT colni_razpon_fun(20,40); -- Klic funkcije
```

Dostop do parametrov procedure

- Napišite shranjeno proceduro, ki vrne število čolnov z dolžino med "spodnja" in "zgornja" meja v zadnjem parametru "stevilo".

```
DELIMITER //
CREATE PROCEDURE colni_stevilo (
    IN spodnja INTEGER,
    IN zgornja INTEGER,
    OUT stevilo INTEGER)
BEGIN
    SELECT COUNT(*) INTO stevilo
    FROM coln
    WHERE dolzina BETWEEN spodnja AND zgornja;
END //
DELIMITER ;
-- Klic procedure in vračanje vrednosti
CALL colni_stevilo(20,40,@s);
SELECT @s;
```

Brisanje procedur in funkcij

- `DROP PROCEDURE [IF EXISTS] Ime_procedure;`
- `DROP FUNCTION [IF EXISTS] Ime_Funkcije;`

Bazni prožilci

- Podobni proceduram, le da nimajo argumentov in se avtomatsko kličejo ob ažuriranju tabele.
- Lahko se kličejo pri vstavljanju (INSERT), brisanju (DELETE) ali posodabljanju (UPDATE).
- Stavčni in vrstični prožilci.

Jadralec(jid, ime, rating, starost)
Coln(cid, ime, dolzina, barva)
Rezervacija(jid, cid, dan)

Vaja

- Kreirajte tabelo *MladoletniJadralci* s funkcionalnostjo materializiranega pogleda, ki hrani jadralce mlajše od 18 let. Dodajte bazne prožilce za vstavljanje, brisanje in posodabljanje vrstic originalne tabele *jadralci*.

Jadralec(jid, ime, rating, starost)
Coln(cid, ime, dolzina, barva)
Rezervacija(jid, cid, dan)

Ustvarjanje tabele

```
CREATE TABLE MladoletniJadralci (  
    jid INTEGER PRIMARY KEY  
        AUTO_INCREMENT,  
    ime VARCHAR(20),  
    rating INTEGER,  
    starost REAL  
)AS SELECT jid, ime, rating, starost  
FROM jadralec  
WHERE starost <18;
```

Nalogo lahko zelo poenostavite, če dodate dodatno omejitev CHECK pri gradnji tabele MladoletniJadralci. Katero?

Jadralec(jid, ime, rating, starost)
Coln(cid, ime, dolzina, barva)
Rezervacija(jid, cid, dan)

Bazni prožilec za vstavljanje

```
DELIMITER //  
CREATE TRIGGER MladoletniJadralci_Insert  
AFTER INSERT ON jadralec  
FOR EACH ROW  
BEGIN  
IF NEW.starost <18 THEN  
    INSERT INTO MladoletniJadralci VALUES(NEW.jid, NEW.ime,  
                                           NEW.starost, NEW.rating);  
END IF;  
END //  
DELIMITER ;
```

Privzeti besedi NEW in OLD

Pri gradnji baznih prožilcev imate na voljo dve besedi NEW in OLD.

Pomen NEW in OLD glede na čas proženja baznega prožilca:

-INSERT

NEW – vrednosti nove vrstice, ki jo vstavljamo

OLD – ne obstaja

-DELETE

NEW – ne obstaja

OLD – vrednosti izbrisane vrstice

-UPDATE

NEW - vrednosti vrstice PO izvedenem UPDATE stavkom

OLD – vrednosti vrstice PRED izvedenem UPDATE stavkom

Jadralec(jid, ime, rating, starost)
Coln(cid, ime, dolzina, barva)
Rezervacija(jid, cid, dan)

Vaje procedure, funkcije in bazni prožilci

1. Tabeli MladoletniJadralci dodajte še bazne prožilce za brisanje in spreminjanje vrstic iz tabele jadralci.
2. Napišite proceduro, ki bo izbrisala vsa naselje s populacijo nič in vam v zadnjem parametru "stevilo" vrnila stevilo izbranih naselji.
3. Napišite proceduro StatistikaAlianse(ime), ki vrne šifro in ime alianse, imena njenih članov in za vsakega člana število njegovih naselji.
4. Napišite funkcijo NajBarva(jid), ki vam za jadralca s šifro jid vrne barvo čolna, ki ga je največkrat rezerviral.
 - Dodatno:* Če ima več najljubših barv naj vrne vse ločene z vejico!
Pomoč: GROUP_CONCAT(barva SEPARATOR ',')
5. Tabeli Naselje dodajte bazne prožilce, ki izpiše napako, če vrednosti x in y nista med -400 in 400 ali če je population negativen.
 - Napako sprožite z: SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = 'napaka'

```
Jadralec(jid, ime, rating, starost)  
Coln(cid, ime, dolzina, barva)  
Rezervacija(jid, cid, dan)
```

Vaje procedure, funkcije in bazni prožilci

1. Napišite proceduro premesti(emp_no, dept_no, datum), ki bo premestila zaposlenega z trenutnega delovnega mesta na nov oddelek. Parameter datum je začetek dela na tem odelku.
2. Napišite funkcijo izplacila(emp_no, start_date, end_date), ki vrne vsoto vseh izplacil v tem obdobju. Predpostavite, da je izplacilo izdano točno 8 dni po to_date v tabeli salaries.
3. Napišite bazni prožilec, ki prepreči vnos v tabelo salaries, če se izplačilo že prekriva z obstoječimi datumi. From_date in to_date sta lahko enaka.