

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

SQL (stik v WHERE delu stavka SQL)

- Poišči imena jadralcev, ki so rezervirali čoln s šifro 103.

```
SELECT ime
FROM jadralec, rezervacija
WHERE jadralec.jid = rezervacija.jid
      AND rezervacija.cid = 103;
```

```
+-----+
| ime    |
+-----+
| Darko  |
| Lojze  |
| Henrik |
+-----+
```

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

Alternativna sintaksa za stik: operator JOIN

```
SELECT jadralec.ime  
FROM jadralec, rezervacija  
WHERE jadralec.jid=rezervacija.jid AND rezervacija.cid = 103;
```

$$\Pi_{jadralec.ime}(\sigma_{jadralec.jid=rezervacija.jid \wedge rezervacija.cid=103}(jadralec \times rezervacija))$$

```
SELECT jadralec.ime      -- Naravni stik  
FROM jadralec NATURAL JOIN rezervacija  
WHERE rezervacija.cid = 103;
```

$$\Pi_{jadralec.ime}(jadralec \bowtie \sigma_{rezervacija.cid=103}(rezervacija))$$

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

Alternativna sintaksa za stik: operator JOIN

```
SELECT jadralec.ime      -- Ekvistik
FROM jadralec JOIN rezervacija USING(jid)
WHERE rezervacija.cid = 103;
```

$$\Pi_{jadralec.ime}(jadralec \underset{jid}{\bowtie} \sigma_{rezervacija.cid=103}(rezervacija))$$

```
SELECT jadralec.ime      -- Pogojni stik
FROM jadralec JOIN rezervacija ON (rezervacija.jid = jadralec.jid)
WHERE rezervacija.cid = 103;
```

$$\Pi_{jadralec.ime}(jadralec \underset{jadralec.jid=rezervacija.jid}{\bowtie} \sigma_{rezervacija.cid=103}(rezervacija))$$

```
SELECT jadralec.ime      -- Pogojni stik s sestavljenim pogojem
FROM jadralec JOIN rezervacija ON (rezervacija.jid = jadralec.jid AND
                                   rezervacija.cid = 103);
```

$$\Pi_{jadralec.ime}(jadralec \underset{jadralec.jid=rezervacija.jid \wedge rezervacija.cid=103}{\bowtie} rezervacija)$$

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

Zunanji stiki

```
SELECT jadralec.ime -- levi zunanji stik
FROM   jadralec LEFT JOIN rezervacija USING(jid)
WHERE  rezervacija.cid = 103;
```

$$\Pi_{jadralec.ime}(jadralec \bowtie_{jid} \sigma_{rezervacija.cid}(rezervacija))$$

```
SELECT jadralec.ime -- desni zunanji stik
FROM   jadralec RIGHT OUTER JOIN rezervacija ON (rezervacija.jid =
        jadralec.jid)
WHERE  rezervacija.cid = 103;
```

$$\Pi_{jadralec.ime}(jadralec \bowtie_{jadralec.jid=rezervacija.jid} \sigma_{rezervacija.cid=103}(rezervacija))$$

```
SELECT jadralec.ime -- levi in desni zunanji stik skupaj
FROM   jadralec FULL OUTER JOIN rezervacija USING(jid)
WHERE  rezervacija.cid = 103;
```

-- MariaDB ne podpira polnega zunanjega stika.

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

SQL: delo z množicami - vsebovanost elementa

- Poišči imena jadralcev z ratingi 1, 3 ali 7.

```
SELECT ime
FROM jadralec
WHERE (rating = 1) OR
      (rating = 3) OR
      (rating = 7);
```

```
+-----+
| ime    |
+-----+
| Darko  |
| Borut  |
| Henrik |
| Anze   |
| Bine   |
+-----+
```

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

SQL: vsebovanost elementa

- Poišči imena jadralcev, z ratingi 1, 3 ali 7.

$$\Pi_{ime}(\sigma_{rating \in \{1,3,7\}}(jadralec))$$

```
SELECT ime
FROM jadralec
WHERE rating IN (1,3,7);
```

Množica v SQL. Namesto seznama imamo lahko tudi rezultat neke poizvedbe.

```
+-----+
| ime    |
+-----+
| Darko  |
| Borut  |
| Henrik |
| Anze   |
| Bine   |
+-----+
```

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

SQL (gnezdene poizvedbe)

- Poišči imena jadralcev, ki so rezervirali čoln s šifro 103.

$$\pi_{ime} \left(\sigma_{jid \in \{ \pi_{jid} (\sigma_{cid=103} (\text{Rezervacija}) \}} (\text{Jadralec}) \right)$$

```
SELECT ime
FROM jadralec
WHERE jid IN
  (SELECT jid
   FROM rezervacija -- Mnozica rezervacij
   WHERE cid=103); -- colna 103
```

```
+-----+
| ime    |
+-----+
| Darko  |
| Lojze  |
| Henrik |
+-----+
```

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

SQL (gnezdene poizvedbe, kvantifikacija)

- Poišči imena jadralcev, ki so rezervirali kakšen rdeč čoln.

$$\Pi_{ime}(\sigma_{jid \in \{\Pi_{jid}(\sigma_{cid \in \{\Pi_{cid}(\sigma_{barva='rdeca'}(coln))\}}(rezervacija))\}}(jadralec))$$

```
SELECT ime
FROM jadralec
WHERE jid IN      -- Mnozica rezervacij
  (SELECT jid      -- rdecih colnov
   FROM rezervacija
   WHERE cid IN    -- Mnozica rdecih colnov
    (SELECT cid
     FROM coln
     WHERE barva='rdeca'));
```

```
+-----+
| ime    |
+-----+
| Darko  |
| Lojze  |
| Henrik |
+-----+
```


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

SQL (gnezdene poizvedbe, kvantifikacija)

- Poišči imena jadralcev, ki nikoli niso rezervirali nobenega rdečega čoln. Kvantificirano: nobenega!

```
SELECT ime
FROM jadralec
WHERE jid NOT IN      -- Mnozica rezervacij
      (SELECT jid      -- rdecih colnov
       FROM rezervacija
       WHERE cid IN    -- Mnozica rdecih colnov
            (SELECT cid
             FROM coln
             WHERE barva='rdeca'));
```

```
+-----+
| ime    |
+-----+
| Borut  |
| Andrej |
| Rajko  |
| Zdravko|
| Henrik |
| Anze   |
| Bine   |
+-----+
```

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

SQL (gnezdene poizvedbe, kvantifikacija)

- Poišči imena jadralcev, ki vsaj enkrat niso rezervirali rdečega čoln.

```
SELECT DISTINCT j.ime
FROM jadralec j, rezervacija r, coln c
WHERE j.jid=r.jid AND r.cid=c.cid AND
      c.barva <> 'rdeca';
```

```
+-----+
| ime    |
+-----+
| Darko  |
| Lojze  |
| Henrik |
+-----+
```

- Zakaj imamo manj imen v rezultatu?
- Lahko so kdaj rezervirali rdeč čoln
- V prejšni poizvedbi tudi tisti, ki niso še nič rezervirali!
- Kako popraviti?

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

SQL (gnezdene poizvedbe, kvantifikacija)

- Popravek prejšnje naloge: poišči imena jadralcev, ki so že rezervirali kakšen čoln, vendar nikoli rdečega!

```
+-----+  
|  ime  |  
+-----+  
| Henrik |  
+-----+
```

```
SELECT ime  
FROM jadralec NATURAL JOIN rezervacija  
WHERE jid NOT IN      -- Mnozica rezervacij  
  (SELECT jid         -- rdecih colnov  
   FROM rezervacija  
   WHERE cid IN       -- Mnozica rdecih colnov  
    (SELECT cid  
     FROM coln  
     WHERE barva='rdeca'));
```

Gnezdenje v FROM vrstici

- Pozvedbe lahko gnezdimo tudi v FROM vrstici, pri čemer se rezultat poizvedbe naprej obravnava kot (začasna) tabela in ga je zato potrebno ustrezno poimenovati.
- Vsi atributi v SELECT vrstici gnezdene poizvedbe morajo imeti eksplicitno določena imena.

```
SELECT stik.*
FROM (SELECT j.jid, r.cid, j.starost
      FROM jadralec j, rezervacija r
      WHERE j.jid = r.jid) AS stik
WHERE stik.starost > 40;
```

AS opcijsko, zaradi preglednosti

Začasna tabela stik ----->

Rdeče vrstice bodo izločene

+-----+-----+-----+			
jid cid starost			
+-----+-----+-----+			
22	101	45	
22	102	45	
22	103	45	
22	104	45	
31	102	55.5	
31	103	55.5	
31	104	55.5	
64	101	35	
64	102	35	
74	103	35	
+-----+-----+-----+			

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

SQL (Korelirane gnezdene poizvedbe)

- Poišči imena jadralcev, ki so rezervirali čoln številka 103.

```
SELECT j.ime
FROM jadralec j
WHERE EXISTS
    (SELECT *
     FROM rezervacija r
     WHERE r.cid = 103 AND
           r.jid = j.jid);
```

-- Neprazna mnozica
-- rezervacij colna
-- 103 za vsakega
-- jadralca
-- posebej

```
+-----+
| ime    |
+-----+
| Darko  |
| Lojze  |
| Henrik |
+-----+
```

- Problem: neučinkovitost, zato se jim izognemo, kadar je le mogoče.

Operatorji za delo z množicami

- UNION: unija \cup
UNION ALL: unija \cup s ponavljanjem elementov
- INTERSECT: presek \cap
- MINUS ali EXCEPT: razlika $-$
- IN, NOT IN (tabela): pripadnost \in in \notin
- ALL, ANY: kvantifikatorja \forall in \exists
- EXISTS, NOT EXISTS (tabela): (ne)praznost množice
- UNIQUE (tabela): enoličnost elementov v tabeli
- Operatorji IN, NOT IN in EXISTS so osnova za gnezdenje poizvedb

SQL: operatorji za delo z množicami

- Unija: UNION, UNION ALL (ohrani duplikate)
- Presek: INTERSECT (*)
- Razlika: MINUS ali EXCEPT (*)
- Sintaksa:
SELECT ...
<OPERATOR>
SELECT ...;
- Kompatibilnost tabel (ali rezultatov SELECT stavka):
isto število stolpcev, istoležni stolpci istega tipa
- *Starejše verzije MariaDB strežnika ne podpirajo

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

SQL: unija množic

- Poišči imena jadralcev, ki so rezervirali rdeč ALI zelen čoln.

$$\pi_{jadralec.ime}(\sigma_{barva=rdeca}(\text{Coln}) \mid \times_{cid} \mid \text{Rezervacija} \mid \times_{jid} \mid \text{Jadralec}) \cup$$

$$\pi_{jadralec.ime}(\sigma_{barva=zelen}(\text{Coln}) \mid \times_{cid} \mid \text{Rezervacija} \mid \times_{jid} \mid \text{Jadralec})$$

```
SELECT DISTINCT j.ime
FROM jadralec j, rezervacija r, coln c
WHERE j.jid=r.jid AND r.cid=c.cid AND
      c.barva='rdeca'
```

UNION

```
SELECT DISTINCT j.ime
FROM jadralec j, rezervacija r, coln c
WHERE j.jid=r.jid AND r.cid=c.cid AND
      c.barva='zelen';
```

```
+-----+
| ime    |
+-----+
| Darko  |
| Lojze  |
| Henrik |
+-----+
```


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

SQL: presek množic

- Poišči imena jadralcev, ki so rezervirali rdeč IN zelen čoln.

$$\pi_{jadralec.ime}(\sigma_{barva=rdeca}(\text{Coln}) \mid \times_{cid} \mid \text{Rezervacija} \mid \times_{jid} \mid \text{Jadralec}) \cap$$

$$\pi_{jadralec.ime}(\sigma_{barva=zelen}(\text{Coln}) \mid \times_{cid} \mid \text{Rezervacija} \mid \times_{jid} \mid \text{Jadralec})$$

```
SELECT DISTINCT j.ime
FROM jadralec j, rezervacija r, coln c
WHERE j.jid=r.jid AND r.cid=c.cid AND
      c.barva='rdeca'
```

INTERSECT

```
SELECT DISTINCT j.ime
FROM jadralec j, rezervacija r, coln c
WHERE j.jid=r.jid AND r.cid=c.cid AND
      c.barva='zelen';
```

```
+-----+
| ime    |
+-----+
| Darko  |
| Lojze  |
| Henrik |
+-----+
```

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

Presek z uporabo gnezdenja

- Poišči imena jadralcev, ki so rezervirali rdeč IN zelen čoln.

$$\pi_{jadralec.ime}(\sigma_{barva=rdeca}(\text{Coln}) \mid \times_{cid} \mid \text{Rezervacija} \mid \times_{jid} \mid \text{Jadralec}) \cap$$

$$\pi_{jadralec.ime}(\sigma_{barva=zelen}(\text{Coln}) \mid \times_{cid} \mid \text{Rezervacija} \mid \times_{jid} \mid \text{Jadralec})$$

```

SELECT DISTINCT j.ime          -- Prva mnozica
FROM jadralec j, rezervacija r, coln c
WHERE j.jid=r.jid AND r.cid=c.cid AND
      c.barva='rdeca'
AND j.ime IN (                  -- Druga mnozica
  SELECT DISTINCT j.ime
  FROM jadralec j, rezervacija r, coln c
  WHERE j.jid=r.jid AND r.cid=c.cid
    AND c.barva='zelena');

```

ime
Darko
Lojze
Henrik

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

SQL: razlika množic

- Poišči imena jadralcev, ki so rezervirali rdeč čoln vendar nikoli zelenega.

$$\pi_{jadralec.ime}(\sigma_{barva=rdeca}(\text{Coln}) \mid \times_{cid} \mid \text{Rezervacija} \mid \times_{jid} \mid \text{Jadralec}) -$$

$$\pi_{jadralec.ime}(\sigma_{barva=zeleni}(\text{Coln}) \mid \times_{cid} \mid \text{Rezervacija} \mid \times_{jid} \mid \text{Jadralec})$$

```
SELECT DISTINCT j.ime
FROM jadralec j, rezervacija r, coln c
WHERE j.jid=r.jid AND r.cid=c.cid AND
      c.barva='rdeca'
```

MINUS -- ali EXCEPT

```
SELECT DISTINCT j.ime
FROM jadralec j, rezervacija r, coln c
WHERE j.jid=r.jid AND r.cid=c.cid AND
      c.barva='zelena';
```

```
+-----+
| ime   |
+-----+
+-----+
```

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

Razlika množic z gnezdenjem

- Poišči imena jadralcev, ki so rezervirali rdeč čoln vendar nikoli zelenega.

$$\pi_{jadralec.ime}(\sigma_{barva=rdeca}(\text{Coln}) \mid \times_{cid} \mid \text{Rezervacija} \mid \times_{jid} \mid \text{Jadralec}) -$$

$$\pi_{jadralec.ime}(\sigma_{barva=zeleni}(\text{Coln}) \mid \times_{cid} \mid \text{Rezervacija} \mid \times_{jid} \mid \text{Jadralec})$$

```
SELECT DISTINCT j.ime
FROM jadralec j, rezervacija r, coln c
WHERE j.jid=r.jid AND r.cid=c.cid AND
      c.barva='rdeca'
```

```
+-----+
|  ime  |
+-----+
+-----+
```

```
AND j.ime NOT IN (
  SELECT DISTINCT j.ime
  FROM jadralec j, rezervacija r, coln c
  WHERE j.jid=r.jid AND r.cid=c.cid AND
        c.barva='zeleni');
```

Urejanje izpisa SELECT stavka

- SELECT stavku dodamo vrstico:
`ORDER BY ime_atributa [ASC ali DESC]`
ali
`ORDER BY številka_atributa [ASC ali DESC]`
ali (za več atributov)
`ORDER BY ime1 ASC, ime2 DESC, ...`
- Lahko urejamo tudi po izrazu ali na novo izračunanem atributu, ki ga ustrezno poimenujemo.
- Urejanje pri množiških operacijah ali gnezdenju ni smiselno, zato ni dovoljeno.

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

Urejanje izpisa

- Izpiši imena jadrancev urejena po količniku med ratingom in starostjo.

ROUND(stevilo, mest) zaokroži rezultat na dano število mest.

```
SELECT ime, ROUND(rating/starost,2)
FROM jadralec;
```

ime	ROUND(rating/starost,2)
Darko	0.16
Borut	0.03
Lojze	0.14
Andrej	0.31
Rajko	0.29
Henrik	0.20
Zdravko	0.62
Henrik	0.26
Anze	0.12
Bine	0.05

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

Urejanje izpisa

- Izpiši imena jadrancev urejena po količniku med ratingom in starostjo.

ROUND(stevilo, mest) zaokroži rezultat na dano število mest.

```
SELECT ime, ROUND(rating/starost,2)
FROM jadralec
ORDER BY 2 DESC;
```

ime	ROUND(rating/starost,2)
Zdravko	0.62
Andrej	0.31
Rajko	0.29
Henrik	0.26
Henrik	0.20
Darko	0.16
Lojze	0.14
Anze	0.12
Bine	0.05
Borut	0.03

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

Urejanje izpisa

- Izpiši imena jadrancev urejena po količniku med ratingom in starostjo.

```
SELECT ime, ROUND(rating/starost,2)
        AS kolicnik
FROM jadralec
ORDER BY kolicnik DESC;
```

ime	kolicnik
Zdravko	0.62
Andrej	0.31
Rajko	0.29
Henrik	0.26
Henrik	0.20
Darko	0.16
Lojze	0.14
Anze	0.12
Bine	0.05
Borut	0.03

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

Vaje: stiki in gnezdenje v SQL

Z uporabo gnezdenja in množiških operatorjev napišite poizvedbe, ki rešijo spodnje naloge:

1. Ugotovite ali imata kakšna dva čolna enako ime. (z množicami)
2. Izpišite šifre čolnov daljših od 35 čevljev, ki so jih rezervirali jadranci mlajši od 40 let.
3. Za vse jadrance izpišite podatke o njihovih rezervacijah. Za tiste, ki še niso nič rezervirali, naj bodo polja o rezervacijah prazna. Rešite z uporabo zunanega stika in izpis uredite po `jid` !
4. Za vse jadrance izpišite podatke o njihovih rezervacijah. Za tiste, ki še niso nič rezervirali, naj bodo polja o rezervacijah prazna. Rešite brez uporabe zunanega stika in izpis uredite po `jid` !
5. Preverite, ali pri prejšnji nalogi dobite enak rezultat, kot pri uporabi zunanega stika.

```
Aliansa(aid, alliance),           Pleme(tid, tribe)
Igralec(pid, player, #tid, #aid)
Naselje(id, vid, village, x, y, population, #pid)
```

Vaje: stiki in gnezdenje v SQL

Z uporabo gnezdenja in množiških operatorjev napišite poizvedbe, ki rešijo spodnje naloge:

1. Izpišite imena vseh igralcev, ki pripadajo naslednjim aliansam: GVERILA, IPERIT, *!NFS!*, PaX™, MGP I, -MGP-, sladoled, CYANIDE, *!NFS!*2, SVN, MGP, ANTHRAX, Uporniki, MGP A, GVERIL4, LEGEND, LUDI DDT, T•W in G.R.™
2. Izpišite imena igralcev, ki imajo vsaj eno naselje od točke (0,0) oddaljena za več kot 20 vendar manj kot 50 enot.
3. Izpišite imena igralcev, ki imajo v vsakem kvadrantu vsaj eno naselje.

Employees(emp_no, birth_date, first_name, last_name, gender, hire_date)
Departments(dept_no, dept_name)
Salaries(emp_no, salary, from_date, to_date)
Titles(emp_no, title, from_date, to_date)
Dept_emp(emp_no, dept_no, from_date, to_date)
Dept_manager(emp_no, dept_no, from_date, to_date)

Vaje: stiki in gnezdenje v SQL

Z uporabo gnezdenja in množiških operatorjev napišite poizvedbe, ki rešijo spodnje naloge:

1. Poiščite zaposlene, ki nikoli niso delali kot vodje odelkov.
2. Poiščite šifre zaposlenih, ki so dobili izplačilo za manj kot tri dni in tudi za več kot 364 dni.
3. Poiščite zaposlene, ki so napredovali iz naziva „Assistant Engineer“ na „Engineer“.
4. Poiščite zaposlene, ki so se zaposlili, ko so bili stari 20 ali manj, in tiste zaposlene, ki na enem odelku delajo vsaj 15 let.