

# Poizvedovanje z jezikom SQL

**Definicija.** Stavek SELECT v jeziku SQL DML (brez agregacije in grupiranja).

```
SQL
5. SELECT [*|ALL|DISTINCT] A, B AS C -- projekcija in preimenovanje
1.   FROM t tabela ... -- preimenovanje (in stiki) tabel
2.   WHERE A = 1 -- selekcija po atributih
      /* GROUP BY ... */
      /* HAVING ... */
3.   ORDER BY A, B [ASC|DESC] -- urejanje izpisa
4.   LIMIT [0,] 10; -- omejevanje izpisa
```

## Projekcija in selekcija

**Projekcija.** Izpiši vrednosti stolpcev  $A$  in  $B$  v tabeli  $t$  (brez podvojenih vrstic).

$$\pi_{A,B}(t)$$

```
SQL
SELECT [ALL|DISTINCT] A, B
      FROM t;
```

Izpiši vse vrednosti v tabeli  $t$ .

$$t$$

```
SQL
SELECT *
      FROM t;
```

**Preimenovanje.** Izpiši vrednosti stolpcev  $A$  in  $B$  v tabeli  $t$  kot Foo in Bar.

$$\rho_{(Foo,Bar)}(\pi_{A,B}(t))$$

```
SQL
SELECT A AS Foo, B AS Bar
      FROM t;
```

**Selekcija.** Izpiši vse vrstice v tabeli  $t$  kjer  $A > 100$  in  $B \neq 1$ .

$$\sigma_{A>100 \wedge B \neq 1}(t)$$

```
SELECT *
FROM t
WHERE A > 100 AND B != 1;
```

SQL

V logičnih pogojih lahko uporabljamo konjunkcijo  $\wedge$ , disjunkcijo  $\vee$  in negacijo  $\neg$ ...

```
AND, OR, NOT
```

SQL

...ter različne operatorje primerjanja.

```
=, !=, <>, <, >, <=, >=, BETWEEN ... AND ..., IS NULL, LIKE "%..."
```

SQL

**Domena.** Facebook&Twitter v datoteki [facebook\\_twitter.sql](#).

facebook (#OID, #PID)  
twitter (#OID, #SID)  
oseba (ID, Ime, Rojen, #SID)  
stan (SID, Stan)

**1. naloga.** (Facebook&Twitter) Z uporabo jezika SQL izpišite identifikatorje, uporabniška imena in rojstne datume vseh polnoletnih oseb.

$$\rho_{(ID, User\ name, Birth)}(\pi_{ID, Ime, Rojen}(\sigma_{Rojen \leq 1.11.2007}(o)))$$

Shema rezultata poizvedbe naj bo enaka spodnji.

ID	User name	Birth
...	...	...

**2. naloga.** (Facebook&Twitter) Z uporabo jezika SQL izračunajte starost vseh oseb.

**Domena.** WikiLeaks v datoteki [wikileaks.sql](#).

embassy (id, name)  
privacy (id, classification)

cable (`id`, date, header, content, #privacy\_id, #embassy\_id)  
reference (#`cable_id`, #`ref_cable_id`)

**3. naloga.** (*WikiLeaks*) Z uporabo jezika SQL poiščite identifikatorje, datume, naslove in vsebino zadnjih pet depeš ambasade z identifikatorjem 140.

**4. naloga.** (*WikiLeaks*) Z uporabo jezika SQL izpišite vse podatke o ambasadi v Ljubljani.

## Stične operacije

**Kartezični produkt.** Izpiši vse pare vrstic v tabelah  $t_1$  in  $t_2$ .

$$t_1 \times t_2$$

```
SELECT *
  FROM t1, t2;
```

SQL

**Naravni stik.** Izpiši naravni stik tabel  $t_1$  in  $t_2$ .

$$t_1 \bowtie t_2$$

```
SELECT *
  FROM t1 NATURAL JOIN t2;
```

SQL

**Ekvistik.** Izpiši ekvistik tabel  $t_1$  in  $t_2$  po stolpcu  $A$ .

$$t_1 \bowtie_A t_2$$

```
SELECT *
  FROM t1 [INNER] JOIN t2 USING (A);
-- ali
SELECT *
  FROM t1, t2
 WHERE t1.A = t2.A;
```

SQL

**Pogojni stik.** Poišči pare vrstic v tabelah  $t_1$  in  $t_2$ , kjer je  $t_1.A$  večji ali enak  $t_2.B$ .

$$t_1 \bowtie_{t_1.A \geq t_2.B} t_2$$

```

SELECT *
  FROM t1 [INNER] JOIN t2 ON t1.A >= t2.B;
-- ali
SELECT *
  FROM t1, t2
 WHERE t1.A >= t2.B;

```

SQL

**Odpri stik.** Izpiši vse vrstice v tabeli  $t_1$  in pare vrstic v tabeli  $t_2$ , kjer je  $t_1.A$  večji ali enak  $t_2.B$ .

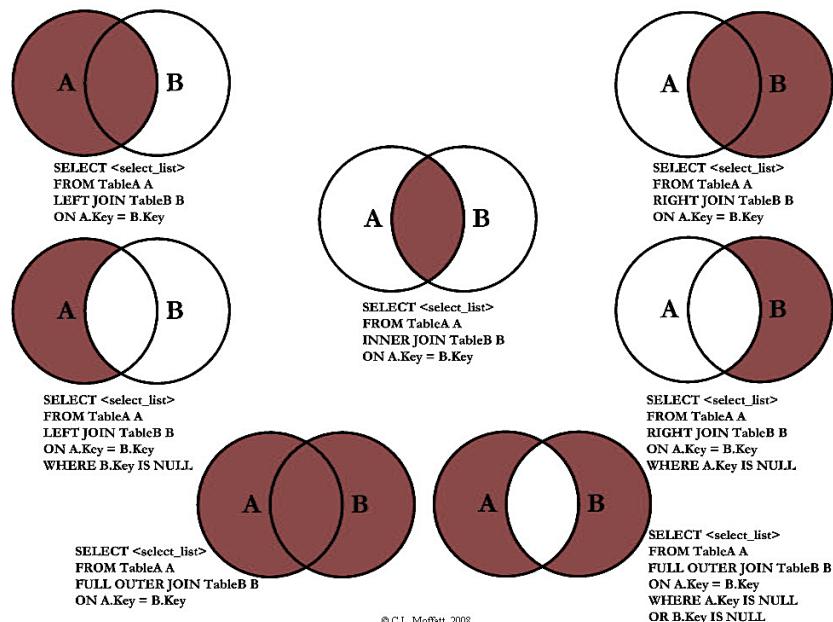
$$t_1 \bowtie_{t_1.A \geq t_2.B} t_2$$

```

SELECT *
  FROM t1 LEFT [OUTER] JOIN t2 ON t1.A >= t2.B;

```

SQL



**Domena.** Facebook&Twitter v datoteki [facebook\\_twitter.sql](#).

facebook (#OID, #PID)  
 twitter (#OID, #SID)  
 oseba (#ID, Ime, Rojen, #SID)  
 stan (#SID, Stan)

**5. naloga.** (*Facebook&Twitter*) Z uporabo jezika SQL izpišite vse podatke o osebah na Facebooku.

$$\rho_{(ID, User\ name, Status)}(\pi_{ID, Ime, Stan}(o \bowtie s \bowtie_{ID=OID} f))$$

Shema rezultata poizvedbe naj bo enaka spodnji.

ID	User name	Status
...	...	...

**6. naloga.** (*Facebook&Twitter*) Z uporabo jezika SQL poiščite identifikatorje oseb, ki *niso* zgolj na Twitterju. Nalogo rešite z uporabo odprtrega stika [OUTER JOIN](#) (glej sliko).

**Domena.** WikiLeaks v datoteki [wikileaks.sql](#).

embassy (id, name)  
privacy (id, classification)  
cable (id, date, header, content, #privacy\_id, #embassy\_id)  
reference (#cable\_id, #ref\_cable\_id)

**7. naloga.** (*WikiLeaks*) Z uporabo jezika SQL izpišite podatke zaupnih depeš ambasade v Ljubljani leta 2010.

**8. naloga.** (*WikiLeaks*) Z uporabo jezika SQL poiščite dneve, ko je ista ambasada odposlala vsaj dve depeši. Shema rezultata poizvedbe naj bo enaka spodnji.

Embassy	Day
...	...

## Operacije množic in deljenje

**Unija.** Izpiši vse vrstice v tabelah  $t_1$  in  $t_2$ .

$$t_1 \cup t_2$$

```
SELECT *
  FROM t1
UNION [DISTINCT|ALL]
SELECT *
  FROM t2;
```

SQL

**Presek.** Izpiši vse enake vrstice v tabelah  $t_1$  in  $t_2$ ,...

$$t_1 \cap t_2$$

```
SQL
SELECT *
  FROM t1
INTERSECT -- ni podprt?
SELECT *
  FROM t2;
```

pri čimer je  $A$  primarni ključ tabeli  $t_1$  in  $t_2$ .

```
SQL
SELECT *
  FROM t1
 WHERE t1.A IN ( -- ugnezdena poizvedba
      SELECT t2.A FROM t2);
```

**Razlika.** Izpiši vrstice, ki so zgolj v tabeli  $t_1$ , pri čimer je  $A$  primarni ključ.

$$t_1 - t_2$$

```
SQL
SELECT *
  FROM t1
MINUS -- ni podprt?
SELECT *
  FROM t2;
-- ali
SELECT *
  FROM t1
 WHERE t1.A NOT IN (SELECT t2.A FROM t2);
```

**Operatorji.** Deljenje lahko implementiramo z dvema operatorjema `NOT EXISTS`.

```
SQL
... WHERE [NOT] EXISTS (SELECT...); -- neprazna tabela
... WHERE A [NOT] IN (SELECT A FROM...); -- vsebovanost v tabeli
... WHERE A >= ALL (SELECT A FROM...); -- največja vrednost A
... WHERE A > ANY (SELECT A FROM...); -- ne najmanjša vrednost A
```

**Domena.** Facebook&Twitter v datoteki `facebook_twitter.sql`.

facebook (#OID, #PID)  
twitter (#OID, #SID)  
oseba (ID, Ime, Rojen, #SID)

stan (SID, Stan)

**9. naloga.** (*Facebook&Twitter*) Z uporabo jezika SQL poiščite vse podatke najstarejših oseb.

$$o - \rho_{o_1}(o) \triangleright_{o_1.Rojen > o_2.Rojen} \rho_{o_2}(o)$$

**Domena.** Trgovina v datoteki `trgovina.sql`.

stranka (id, ime, priimek, mesto, popust)

agent (id, ime, priimek, mesto, marza)

izdelek (id, ime, zaloga, cena)

narocilo (id, #stranka\_id, #agent\_id, #izdelek\_id, datum, kolicina)

**10. naloga.** (*Trgovina*) Z uporabo jezika SQL izpišite imena in priimke strank, ki so naročile izdelek z identifikatorjem 2. Nalogo rešite brez stikov, dočim naj bo shema rezultata poizvedbe enaka spodnji.



**Domena.** GSM v datoteki `gsm.sql`.

prodaja (operater, telefon)

kupuje (stranka, operater)

najraje (stranka, telefon)

**11. naloga.** (*GSM*) Z uporabo jezika SQL poiščite stanke, ki kupujejo pri vseh operaterjih (deljenje).

Nalogo rešite z uporabo operatorja `NOT EXISTS` in dveh ugnezdenih poizvedb.

## Agregacija in grupiranje

**Definicija.** Stavek SELECT v jeziku SQL DML z agregacijo (in grupiranjem).

```
7. SELECT [A, B,] SUM(C) -- agregacija atributov
1.   FROM t tabela ...
2.   WHERE A != 1 AND C > 0 -- selekcija po atributih
3.   [GROUP BY A, B] -- grupiranje po atributih
4.   HAVING SUM(C) > 100 -- selekcija po agregacijah
5.   ORDER BY [B,] SUM(C);
6.   LIMIT 1;
```

SQL

**Agregacija.** Izračunaj najmanjšo, največjo in povprečno vrednost stolpca  $A$  v tabeli  $t$ .

$$\tau_{\min A, \max A, \text{avg } A}(t)$$

```
SELECT MIN(A), MAX(A), AVG(A) -- SUM(A) / COUNT(A)
  FROM t;
```

SQL

V agregacijah lahko uporabljamo funkcije za izračun vsote, števila vrednosti brez `NULL`, najmanjše in največje vrednosti, povprečja, standardnega odklona, variance itd.

```
SUM, COUNT, MIN, MAX, AVG, STD, VARIANCE ...
```

SQL

**Grupiranje.** Za vsako vrednost stolpca  $A$  poiščite število vrstic v tabeli  $t$ ...

$$_A \tau_{COUNT B}(t)$$

```
SELECT A, COUNT(*)
  FROM t
 GROUP BY A;
```

SQL

**Selekcija.** ...ter izpišite vrednosti  $A$ , pri katerih je število vrstic večje od 100.

$$\pi_A(\sigma_{C>100}(\rho_{(A,C)}(_A \tau_{COUNT B}(t))))$$

```
SELECT A
  FROM t
 GROUP BY A
 HAVING COUNT(*) > 100;
```

SQL

**Domena.** *Facebook&Twitter* v datoteki `facebook_twitter.sql`.

facebook (#OID, #PID)  
twitter (#OID, #SID)  
oseba (ID, Ime, Rojen, #SID)  
stan (SID, Stan)

**12. naloga.** (*Facebook&Twitter*) Z uporabo jezika SQL za vsako osebo izpišite identifikator, uporabniško ime in število sledenj na Twitterju (*ne* sledilcev).

$ID, \text{Ime} \tau \text{COUNT OID}(o \rtimes_{ID=t.SID} t)$

**13. naloga.** (*Facebook & Twitter*) Z uporabo jezika SQL izpišite število vseh prijateljstev na Facebooku in število vseh sledenj na Twitterju.

---

**Domena.** WikiLeaks v datoteki `wikileaks.sql`.

embassy (id, name)  
privacy (id, classification)  
cable (id, date, header, content, #privacy\_id, #embassy\_id)  
reference (#cable\_id, #ref\_cable\_id)

**14. naloga.** (*WikiLeaks*) Z uporabo jezika SQL poiščite dneve, ko je bilo odposlanih več kot 300 depeš (ozioroma največ depeš v enem dnevu).

**15. naloga.** (*WikiLeaks*) Z uporabo jezika SQL izpišite število in seznam referenc prvih 100 depeš z največjim številom referenc.

---

Lovro Šubelj ©