

# Delo z datumi

- Povečaj datum jadralcev za 3 dni.

```
SELECT dan, dan + 3, DATE(dan + 3)
FROM rezervacija;
```

- Zgornja rešitev je zavajajoča, v splošnem ne deluje.

```
SELECT dan, dan + 3, DATE(dan + 3),
       DATE(dan + 20), DATE(dan + 100)
FROM rezervacija;
```

# Delo z datumi

- Pravilno povečavanje datuma.

```
SELECT dan, DATE_ADD(dan, INTERVAL 3 DAY),  
        DATE_ADD(dan, INTERVAL 3 WEEK),  
        DATE_ADD(dan, INTERVAL 3 MONTH),  
        DATE_ADD(dan, INTERVAL 3 YEAR)  
FROM rezervacija;
```

- Analogno deluje tudi funkcija `DATE_SUB()` za odštevanje intervala.

# Delo z datumi

- Kateri dan je danes? Kaj pa čas?

```
SELECT NOW(), CURDATE(), DATE(NOW());
```

- Polepšan izpis – formatiranje izpisa.

```
SELECT DATE_FORMAT(CURDATE(), '%d-%m-%Y'),  
       DATE_FORMAT(CURDATE(), '%D-%M-%Y');
```

- Število preteklih dni od leta 2000.

```
SELECT DATEDIFF(CURDATE(), '2000-01-01') AS days;
```

# Delo z datumi

- Izločanje določenih vrednosti iz datuma.

```
SELECT dan, WEEK(dan), WEEKDAY(dan), QUARTER(dan),  
        DAY(dan), MONTH(dan), YEAR(dan)  
FROM rezervacija;
```

# Delo z vrednostjo NULL

- Tisti, ki delate na svoji bazi uporabite naslednje ukaza, da dodamo NULL vrednosti.

```
ALTER TABLE jadralec  
ADD COLUMN rang char(1) AFTER rating;  
UPDATE jadralec SET rang = 'C' WHERE jid = 22;  
UPDATE jadralec SET rang = 'B' WHERE jid = 31 OR jid = 32;  
UPDATE jadralec SET rang = 'A' WHERE jid = 58 OR jid = 71;
```

- Ko končate z vajami, lahko spremembe odstranite z:

```
ALTER TABLE jadralec  
DROP COLUMN rang;
```

# Delo z vrednostjo NULL

- Izberi jadralce, ki imajo neznano vrednost ranga.

```
SELECT *  
FROM jadralec  
WHERE rang IS NULL;
```

# Delo z vrednostjo NULL

- Izberi vse jadralce, ki niso 'A' ranga.

```
SELECT *  
FROM jadralec  
WHERE rang != 'A';
```

- Zakaj ni jadralcev z vrednostjo NULL?  
(Trinivojska logika)

```
SELECT *  
FROM jadralec  
WHERE rang != 'A' OR rang IS NULL;
```

# IF stavek v SELECT delu

- Dodaj stolpec, ki določa pokaže polnoletnost jadralcev pri izpisu.

```
SELECT jadralec.*,  
       IF(starost >= 18, 'Polnoleten', 'Mladoleten') AS  
       Polnoletnost  
FROM jadralec;
```



# Relacijski podatkovni model (RPM)

- Relacije in operacije nad njimi predstavljajo formalno logično osnovo številnih povpraševalnih jezikov (npr. SQL); formalna osnova omogoča številne možnosti optimizacije povpraševanj!
- Dve vrsti operacij:
  - Relacijska algebra: operativna; opišemo načrt izvajanja operacij (SQL)
  - Relacijski račun: neoperativen, deklarativen; opišemo želen rezultat (QBE)

# Osnovni koncepti RPM

- Relacija in relacijska shema
- Atribut
- Vrednostna množica (območje) atributa
- Odvisnosti med atributi

# Relacija

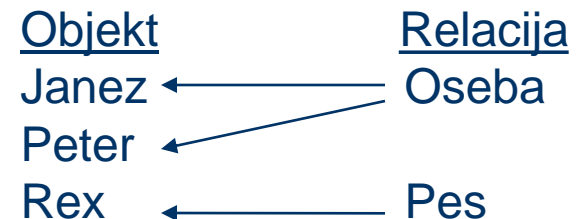
- Preslikava kartezičnega produkta vrednostnih množic  $D_i$ .

$$r : D_1 \times D_2 \times \dots \times D_n \rightarrow \{res, ni\ res\}$$

- Množica resničnih trditev:

je\_oseba={Janez, Peter}

je\_pes={Rex}



# Predstavitev relacije

- Predikatni zapis:
    - Shema: je\_oseba(oseba)
    - je\_oseba(Janez)
    - je\_oseba(Peter)
  - Predikatni zapis: opis objektov z atributi:
    - Shema: je\_oseba(ime, priimek, kraj bivanja)
    - je\_oseba(Janez, Novak, Ljubljana)
    - je\_oseba(Peter, Klepec, Celje)
- ⇒ Naštejemo n-terice, za katere velja relacija
- ⇒ Kakšen je pomen gornjih relacij?

# Predstavitev relacije s tabelo

Oseba
Janez (v celoti)
Peter (v celoti)

Ime	Priimek	Naslov
Janez	Novak	Ljubljana
Peter	Klepec	Celje

Objekti (elementi množice).

Atributni opis objektov (elementov):

**ena vrstica = en objekt !**

Tabela predstavlja množico  
resničnih trditev:

**tabela = relacija!**

# Pomen relacije

- Relacija v dobesednem pomenu:
  - Elementi relacije (objekti, vrstice) izpolnjujejo določene pogoje. (coln in jadralec)
- Relacija v povezovalnem pomenu:
  - Elementi v vrstici (celice) relacije (tabele) so med seboj v nekem razmerju. (rezervacija)
  - Uporaba za povezovanje elementov drugih relacij (tabel) med seboj. (rezervacija)

zakonec(Janez, Micka)                      Janez <sup>zakonec</sup> ————— Micka

# Atribut

- Vsaka n-terica v relaciji predstavlja določen objekt.
- Vsak objekt opišemo z lastnostmi – atributi.
- Atribut kot preslikava lastnosti objekta v pripadajočo domeno:

$$A_i : O \rightarrow D_i$$

# Relacijska shema

- Vsaki relaciji  $r$  pripada natanko ena relacijska shema, sestavljena iz oznake sheme  $R$  in iz oznak imen in domen atributov. Domene atributov lahko izpustimo.

$$R(A_1 : D_1, A_2 : D_2, \dots, A_n : D_n)$$

- Eni shemi lahko pripada več relacij!
- Shema relacije = glava tabele.



# Odvisnosti med atributi

- Omejevanje vrednosti relacij
  - Funkcionalne
  - Večvrednostne
  - Stične
- Veljajo v shemi  $R$ ; torej v vseh relacijah  $r$ , katerih shema je  $R$ .

# Funkcionalne odvisnosti

- Množica atributov  $\{X\}$  funkcionalno določa množico atributov  $\{Y\}$  če v nobeni relaciji s shemo  $R$  ne obstajata  $n$ -terici, ki bi se ujemali v vrednosti atributov  $\{X\}$  in ne ujemali v vrednosti atributov  $\{Y\}$ .
- Zapišemo  $\{X\} \rightarrow \{Y\}$  ali krajše  $X \rightarrow Y$ .
- Množico vseh funkcionalnih odvisnosti v shemi  $R$  označimo s  $F(R)$ .

$$X \rightarrow Y \in F(R) \Leftrightarrow \forall r (Sh(r) = R \Rightarrow \forall t \forall u (t \in r \wedge u \in r \wedge t.X = u.X \Rightarrow t.Y = u.Y))$$

# Ključ relacijske sheme

- Relacija je množica, torej morajo biti vsi elementi ( $n$ -terice) unikatni.
- Minimalna podmnožica atributov, ki enolično identificira vsako  $n$ -terico je ključ.
- Ključ:
  1.  $X \rightarrow R$
  2.  $\neg \exists A : A \subseteq X \wedge (X - A) \rightarrow R$
- Nadključ: ključ, ki vsebuje vsaj en ključ.

# Primarni in tuji ključ

- Primarni ključ: Ključ, ki ga izberemo za enolično identifikacijo vsake n-terice.  
V relacijski shemi primarni ključ podčrtamo.
- Tuji ključ: Množica atributov v relaciji, ki je enaka ključu druge relacije (ali iste).  
V relacijski shemi tuje ključe označimo z #.

Primer: rezervacija(#jid, #cid, dan)

(jid, cid, dan) so skupaj primarni ključ.

(jid) je tuji ključ, ki kaže na relacijo jadralec.

(cid) je tuji ključ, ki kaže na relacijo coln.

# Operacije nad relacijami – relacijska algebra

- Tradicionalni operatorji za delo z množicami: unija  $\cup$ , presek  $\cap$ , razlika  $-$ , kartezični produkt  $\times$
- Posebni relacijski operatorji: selekcija  $\sigma$ , projekcija  $\pi$ , stik  $\bowtie$  ali  $| \times |$ , deljenje  $/$

# Množiški operatorji

Relacija r:

A	B	C
a	b	c
d	a	f
c	b	d

Relacija s:

D	E	F
b	g	a
d	a	f

Pomembna kompatibilnost atributov!

# Unija, presek, razlika

Relacija  $r \cup s$ :

G	H	I
a	b	c
d	a	f
c	b	d
b	g	a
<del>d</del>	<del>a</del>	<del>f</del>

Relacija  $r \cap s$ :

G	H	I
d	a	f

Relacija  $r - s$ :

G	H	I
a	b	c
c	b	d

# Kartezični produkt

Velja asociativnost:  $(r \times s) \times t = r \times (s \times t)$ .

Relacija  $r \times s$ :

A	B	C	D	E	F
a	b	c	b	g	a
d	a	f	b	g	a
c	b	d	b	g	a
a	b	c	d	a	f
d	a	f	d	a	f
c	b	d	d	a	f



# Relacijski operatorji

- Projekcija  $\pi$ : zmanjševanje števila stolpcev
- Selekcija  $\sigma$ : zmanjševanje števila vrstic
- Stik  $| \times |$ : zmanjševanje števila stolpcev in vrstic kartezičnega produkta; zelo pogosta operacija, ki jo lahko realiziramo z drugimi operatorji
- Deljenje  $/$ , ki ga lahko realiziramo z drugimi operatorji

# Projekcija $\pi$

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

A	B
a	b
d	a
c	b

$$\pi_B(r)$$

B
b
a
<del>b</del>

Sintaksa:  $\pi_{A_1,A_2,\dots,A_k}$  - naštejemo attribute

Včasih se lahko zmanjša tudi število vrstic!

# Selekcija $\sigma$

$$\sigma_{B < b}(r)$$

A	B	C
d	a	f

$$\sigma_{B=b \wedge C=d}(r)$$

A	B	C
c	b	d

Sintaksa:  $\sigma_P(r)$

Logični pogoj  $P$  je lahko poljubno kompleksen!

# Pogojni (theta) stik

$$r \mid \underset{\theta}{\times} \mid s = r \mid \underset{P}{\times} \mid s \equiv \sigma_P(r \times s)$$

- Alternativna sintaksa:  $\mid \times \mid$  je isto kot  $\bowtie$

# Pogojni stik (1. korak)

$$r \quad \left| \times \right| \quad S =$$

$(B=D) \vee (C=c)$

A	B	C	D	E	F
a	b	c	b	g	a
d	a	f	b	g	a
c	b	d	b	g	a
a	b	c	d	a	f
d	a	f	d	a	f
c	b	d	d	a	f

# Pogojni stik (2. korak)

$$r \quad \left| \begin{array}{c} \times \\ \times \end{array} \right| \quad S =$$

$(B=D) \vee (C=c)$

A	B	C	D	E	F
a	b	c	b	g	a
<del>d</del>	<del>a</del>	<del>f</del>	<del>b</del>	<del>g</del>	<del>a</del>
c	b	d	b	g	a
a	b	c	d	a	f
<del>d</del>	<del>a</del>	<del>f</del>	<del>d</del>	<del>a</del>	<del>f</del>
<del>c</del>	<del>b</del>	<del>d</del>	<del>d</del>	<del>a</del>	<del>f</del>

# Ekvistik in naravni stik

- Ekvistik: v pogoju lahko od operatorjev nastopajo samo enačaji.
- Naravni stik: ekvistik po vseh istoimenskih atributih
  - Oznaka brez pogoja  $P$ :  $| \times |$  ali  $\bowtie$
  - Ker je nekaj atributov po naravnem stiku odveč, jih izločimo

# Naravni stik (1. korak)

A	B	C
a	b	f
x	b	c



B	C	D
b	f	e
b	c	y

=

A	B	C	B	C	D
a	b	f	b	f	e
x	b	c	b	c	y



# Naravni stik (2. korak)

A	B	C
a	b	f
x	b	c



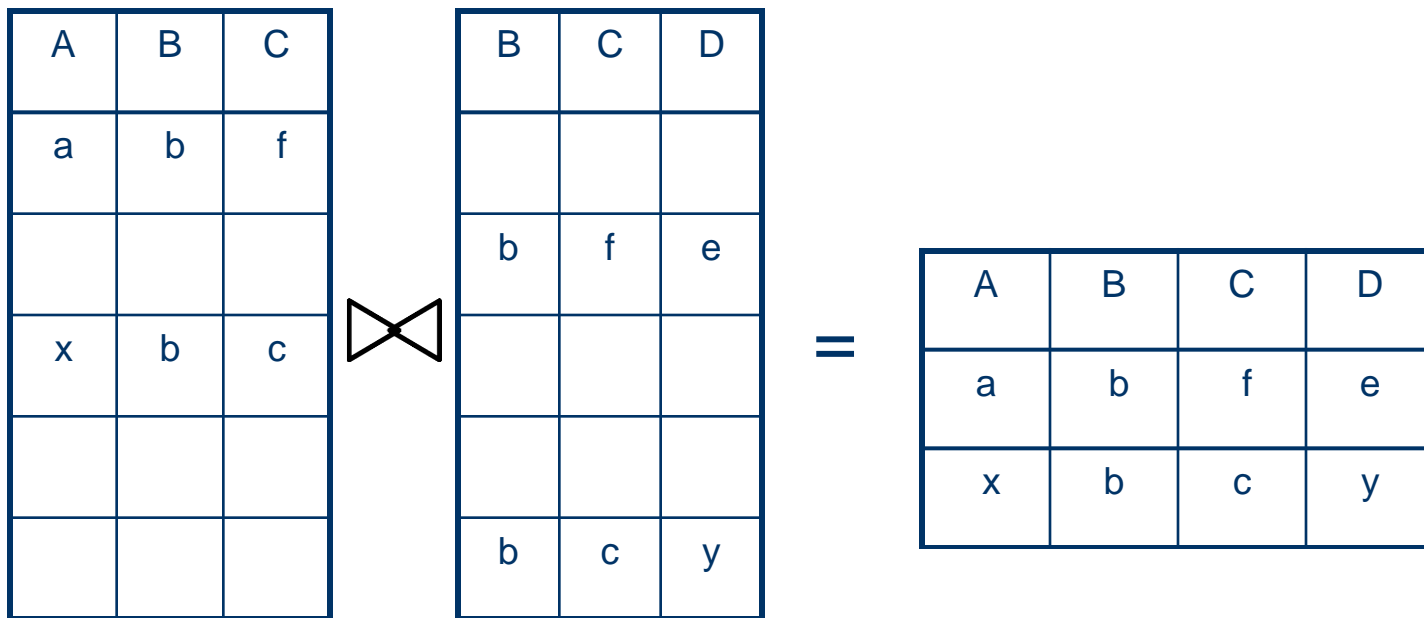
B	C	D
b	f	e
b	c	y

=

A	B	C	B	C	D
a	b	f	b	f	e
x	b	c	b	c	y

Odvečna stolpca!

# Naravni stik (3. korak)



# Primeri relacijske algebre

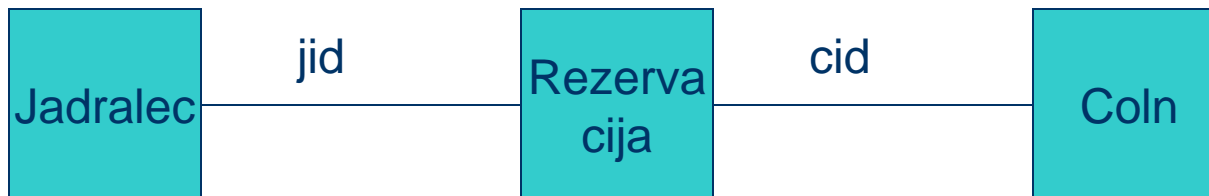
- Sheme za primere rel. algebre:

Jadralec(jid, ime, rating, starost)

Coln(cid, ime, dolzina, barva)

Rezervacija(jid, cid, dan)

- Pomen in povezava relacij:



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

# Projekcija

- Poišči (izpiši) šifre in imena vseh jadralcev.
- Poišči barve vseh čolnov.

$\pi_{jid, ime}(\text{jadralec})$

$\pi_{barva}(\text{coln})$

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

# Selekcija

- Poišči (izpiši) šifre in imena vseh jadralcev, starejših od 50 let.
- Poišči barve vseh čolnov krajših od 40 čevljev.

$$\pi_{jid, ime}(\sigma_{starost > 50}(\text{jadralec}))$$

$$\pi_{barva}(\sigma_{dolzina < 40}(\text{coln}))$$

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

# Stik

- Poišči vse pare imen jadralcev in čolnov, kjer je jadralec rezerviral ustrezen čoln.

$$\pi_{ime,ime}(\text{jadralec} \mid \times \mid \text{rezervacija} \mid \times \mid \text{coln})$$

$$\pi_{\substack{jadralec.ime, \\ coln.ime}}(\text{jadralec} \mid \times \mid \text{rezervacija} \mid \times \mid \text{coln})$$

$\substack{jadralec.jid= \\ rezervacija.jid}$ 
 $\substack{rezervacija.cid= \\ coln.cid}$

$$\pi_{\substack{jadralec.ime, \\ coln.ime}}(\text{jadralec} \mid \times \mid \text{rezervacija} \mid \times \mid \text{coln})$$

$\substack{jid}$ 
 $\substack{cid}$

- Poišči vse pare imen jadralcev in čolnov, kjer je jadralec starejši od 50 let rezerviral ustrezen čoln.

$$\pi_{\substack{jadralec.ime, \\ coln.ime}}(\sigma_{starost>50}(\text{jadralec}) \mid \times \mid \text{rezervacija} \mid \times \mid \text{coln})$$

$\substack{jid}$ 
 $\substack{cid}$

# Vaje – SQL

1. Poišči vse evklidske in manhattan razdalje med naselji igralca “hinko”.
2. Izpiši imena mesecev v katerih so bile opravljene rezervacije.
3. Poišči imena in njihova pripadajoča plemena vseh igrancev, ki pripadajo aliansam, ki vsebujejo znak ‘™’.
4. Poišči vsa imena igralcev, ki ne vsebujejo alfa-numeričnih znakov.
5. Ustvari navidezni atribut “Rangiran”, ki vsebuje vrednost ‘Da’, če ima jadralec podan rang, sicer ‘Ne’.
6. Z IF stavkom polepšajte izpis pri 2 in 6 nalogi prejšnjega tedna (jadranci).

# Vaje – SQL (employees)

1. Poišči šifre (emp\_no) vseh, ki so na vsaj enem odelku bili zaposleni manj kot 5 let in na njem ne delajo več.
2. Poišči plače oziroma honorarje, ki so bili izplačani za največ sedem dni.
3. Poišči ime in priimek zaposlenih, ki so bili vodje odelkov manj kot 3 leta.
4. Poišči ime, priimek in ime odelka samo trenutnih vodji odelkov.



# Vaje

1. Poišči alianse, ki imajo vsaj eno naselje na območju  $(-50, -50)$  do  $(50, 50)$ .
2. Poišči alianse, ki imajo naselja izven območja  $(-50, -50)$  do  $(50, 50)$ . (Brez gnezdenja ali agregacijskih operacij)
3. Poišči vse pare naselji, katerim se populacija razlikuje za največ 100.
4. Poišči vse pare jadralcev, ki imajo enak rang.
5. Poišči vse pare jadralcev, ki še nimajo ranga.
6. Poišči pare igralcev in njihovo pleme alianse "Nwm", ki pripadajo istemu plemenu.

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

# Vaja: rešite z uporabo relacijske algebre in uporabe jezika SQL

1. Poišči šifre vseh Henrikov (ime jadrarca).
2. Poišči imena vseh čolnov, daljših od 20 čevljev.
3. Izpiši pare imen (jadrarec, čoln).
4. Izpišite šifre čolnov z nepopolnimi (manjkajočimi) podatki.
5. Izpiši imena vseh doslej rezerviranih čolnov.
6. Kateri izmed jadrarcev še ni rezerviral nobenega čolna? (Z uporabo jezika SQL še ne znamo!)

# Vaja: rešite z uporabo relacijske algebre

1. Poišči imena zaposlenih, ki so bili s svojim prvim dnem dela zaposleni kot vodja oddelka.
2. Poišči imena in priimke zaposlenih, ki nikoli niso delali kot vodje oddelkov.
3. Poišči zaposlene z enakim imenom in priimkom.
4. Poišči ime, priimek in ime odelka samo trenutnih vodji odelkov.
5. Poiščite plače za vodje oddelkov in njihove nazive.

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

## Vaja 2: rešite z uporabo relacijske algebre in uporabe jezika SQL

1. Izpišite šifre naselji, ki pripadajo igralcu „matija“.
2. Izpišite vsa imena naselji, ki se začnejo ali končajo s cifro.
3. Izpišite šifre in imena igralcev, ki pripadajo aliansi „sladoled“ ali „ANIMALS“. Rešite tudi z unijo.
4. Izpišite imena naselji in število prebivalcev naselji, ki **niso** v tretjem kvadrantu.
5. Izpišite imena igralcev brez alianse.
6. Izpišite imena igralcev, ki ne pripadajo plemenom, ki se začnejo na črko „N“.