

Čiščenje podatkov

Datoteke iz statističnega urada

- ▶ <https://www.stat.si/>
- ▶ Meni zgoraj Področja, izberemo Izobraževanje, v podmeniju izberemo Višješolsko, visokošolsko izobraževanje.
- ▶ Nižje na strani najdemo razdelek Več tabel v podatkovni bazi SI-STAT, kliknemo Izobraževanje.
- ▶ Pomaknemo se za zaslon nižje na strani do drevesnega pogleda izberemo: Razžiri vse, Izobraževanje, Višješolsko, visokošolsko izobraževanje, Vpisani v visokošolsko izobraževanje, tabela s šifro 0955201S.

Izbor oblike podatkov

- ▶ Izberemo vse možnosti v 5 dimenzijah podatkov (razen agregatov SKUPAJ).
- ▶ Nižje pri gumbu Izpis podatkov izberemo Prikaži tabelo na zaslonu, prikaz 2.
- ▶ S pomočjo operacij vrtenja tabel spravimo podatke v željeno obliko in izvozimo.

Tabele s hierarhičnimi indeksi

► Primer:

Študenti visokošolskega študija na univerzah in samostojnih visokošolskih zavodih po: VISOKOŠOLSKI ZA SPOL													
				2004		2005		2006		2007		2008	
				Moški	Ženske	Moški	Ženske	Moški	Ženske	Moški	Ženske	Moški	Ženske
Samostojni visokošolski zavodi	Visokošolsko strokovno (prejšnje)	1. letnik	Redni	122	152	141	218	95	163	-	1	6	5
			Izredni	191	231	218	323	147	259	-	-	-	7
		2. letnik	Redni	130	101	148	166	80	176	23	43	-	4
			Izredni	206	205	374	547	236	502	174	321	5	4
		3. letnik	Redni	67	109	100	80	81	119	51	135	22	45
			Izredni	198	351	285	470	301	588	300	607	170	352
		4. letnik	Redni	-	-	4	7	2	3	-	-	-	6
			Izredni	5	8	1	-	-	1	-	-	1	3

- S hierarhičnimi indeksi je težko delati.
- R nima dobre podpore (npr. Python Pandas ima).

“Navadne tabele”

- ▶ Tabele poskušamo spraviti v obliko “ena meritev-ena vrstica”.
- ▶ Meritev je število vpisanih.
- ▶ Vsi drugi podatki določajo dimenzije, pri katerih je opravljena meritev.
- ▶ Tabelo želimo v taki obliki:
 - ▶ VISOKOŠOLSKI ZAVOD
 - ▶ VRSTA IZOBRAŽEVANJA
 - ▶ LETNIK
 - ▶ NAČIN ŠTUDIJA
 - ▶ ŠTUDIJSKO LETO
 - ▶ ST_STUDENTOV
- ▶ S takimi podatki je v R lažje delati.

Obdelava in pridobivanje podatkov

- ▶ Ročno zavrtimo tabelo.
- ▶ Izvozimo v obliki CSV (ločeno s podpičjem) z glavo.
- ▶ Za namene demonstracije dela s čiščenjem podatkov, bomo uporabili še “staro” datoteko, ki ima še kompleksnejšo strukturo in dodatno dimenzijo SPOL (`viri/0955201ss.csv`).
- ▶ Ogledamo si format CSV v tekstovni datoteki.
- ▶ Ogledamo si še datoteko v Excelu.
- ▶ Prve 4 vrstice so nepomembne.
- ▶ Tabela nima glave.
- ▶ Vsebina v prvih nekaj stolpcih je še vedno podana hierarhično.
- ▶ Prazna polja so označena kot “-”

Branje tabele

- ▶ Uporabili bom paket readr
- ▶ RStudio CheatSheets -
<https://www.rstudio.com/resources/cheatsheets/>
- ▶ Pričakuje se, da na koncu predavanj obvladate cheatsheets-e
Data Import, Data Transformation, RMarkdown, RStudio,
Shiny, Data Visualization (skoraj vse :)
- ▶ Poskusimo naivno prebrati CSV datoteko

```
uvoz <- read_csv2("0955201ss.csv")
```

- ▶ Problem: kodna tabela

```
uvoz <- read_csv2("0955201ss.csv",  
                  locale=locale(encoding="cp1250"))
```

Branje tabele

- ▶ Problem? Pogledamo:

```
problems(uvoz)
```

- ▶ Vizualno pregledamo, kaj se je uvozilo:

```
View(uvoz)
```


Branje tabele

- ▶ V uvozu imamo samo en stolpec!
- ▶ Definirajmo stolpce

```
stolpci <- c("VISOKOSOLSKI_ZAVOD", "VRSTA_IZOBRAZEVANJA",  
"LETNIK", "NACIN_STUDIJA", "SPOL" ,  
"STUDIJSKO_LETO", "ST_STUDENTOV")
```

```
uvoz <- read_csv2("0955201ss.csv",  
                  locale=locale(encoding="cp1250"),  
                  col_names=stolpci)
```



```
problems(uvoz)
```



```
View(uvoz)
```

Branje tabele

- ▶ Izpustiti moramo prve 4 vrstice

```
uvoz <- read_csv2("0955201ss.csv",  
                  locale=locale(encoding="cp1250"),  
                  col_names=stolpci,  
                  skip=4)
```

- ▶ Zadnja prebrana vrstica (po branju), ki nas še zanima je 7162

```
uvoz <- read_csv2("0955201ss.csv",  
                  locale=locale(encoding="cp1250"),  
                  col_names=stolpci,  
                  skip=4,  
                  n_max=7162)
```

Branje tabele

- Znak “-” pri številu študentov bi radi interpretirali kot NA

```
uvoz <- read_csv2("0955201ss.csv",  
                  locale=locale(encoding="cp1250"),  
                  col_names=stolpci,  
                  skip=4,  
                  n_max=7162,  
                  na=c("", " ", "-"))
```

Obdelava podatkov

- ▶ Zaradi 'hierarhično kompresiranega uvoza' bi radi, da se vsi dimenzijski stolpci ponavljajo v vrsticah do naslednjega vnosa
- ▶ Izjema je stolpec STUDIJSKO_LETO

```
podatki <- uvoz %>%  
  fill(1:5) %>%  
  drop_na(STUDIJSKO_LETO)
```

Pregled tipov stolpcev

- ▶ Preverimo tipe stolpcev

```
sapply(podatki, class)
```

- ▶ Tipi so ustrezni.
- ▶ Če bi v kakšnem številskem stolpcu imeli kak “zatipk”, bi verjetno uvozili ta stolpec kot tip character.
- ▶ Tak stolpec bi najprej obdelali kot vektor nizov in potem stolpec pretvorili (npr. s funkcijo `parse_integer()` iz paketa `readr`).

Relacijski model

- ▶ Podatke organiziramo kot množico večih tabel (`data.frame`-ov).
- ▶ Model je v osnovi star več kot 40 let.
- ▶ Uporablja se v večini večjih poslovnih sistemov, relacijskih podatkovnih bazah
- ▶ Enostaven za razumevanje, pregleden.
- ▶ Omogoča zmogljive poizvedbe v standardiziranem jeziku SQL (na podatkovnih bazah) in podobno zmogljive poizvedbe/operacije v R.
- ▶ Podpira učinkovite implementacije (pri podatkovnih bazah in v R).

Relacijska algebra

- ▶ Podatke organiziramo kot množico večih tabel (`data.frame`-ov)
- ▶ *Relacija* = tabela
- ▶ *Relacijska algebra* je matematični opis operacij nad relacijami (tabelami).
- ▶ Operatorji so operacije, ki sprejmejo relacije (tabele) in vrnejo (nove) relacije (tabele).
- ▶ *Shema relacije* = definicija tabele (imena + tipi).

Operatorji relacijske algebre

- ▶ Operatorji so odvisni od shem relacij nad katerimi jih izvajamo.
- ▶ $\sigma_p(R)$ - izberi vrstice v relaciji R , ki ustrezajo pogoju p .
 - ▶ Pogoj je lahko logični izraz.
 - ▶ Shema vrnjene tabele je ista.
 - ▶ Primer: operator $[\text{pogoj},]$ v kombinaciji z logičnim indeksom
- ▶ $\pi_{a_1, a_2, \dots, a_n}(R)$ - izberi stolpce z imeni a_1, a_2, \dots, a_n relacije R in vrni novo tabelo s shemo, ki jo določajo definicije teh stolpcev.
 - ▶ Število vrstic ostane nespremenjeno.
 - ▶ Primer: operator $[\text{vektor_imen}]$.

Operatorji relacijske algebre

- ▶ $\rho_{a/b}(R)$ - spremeni ime stolpcu a v b . Vrni enako tabelo (glede vrstic), le z drugo shemo.
 - ▶ Primer: preimenovanje stolpca.
- ▶ $R \cup S$ - vrni relacijo z unijo vrstic, če imata relaciji R in S enaki shemi.
- ▶ $R \setminus S$ - vrni relacijo z razliko vrstic, če imata relaciji R in S enaki shemi.
- ▶ $R \times S$ - vrni kartezični produkt relacij (vsaka vrstica R z vsako vrstico S).
 - ▶ Shema rezultata sta združeni shemi.
 - ▶ Ni tako uporabna operacija, so pa uporabne ustrezne učinkovite izvedbe s filtriranjem (združitve).

Operacija JOIN

$$R \bowtie S = \pi_{schema(R) \cup schema(S)}(\sigma_{R.a_1=S.a_1 \wedge R.a_2=R.a_2 \wedge \dots}(R \times S))$$

Employee

Name	Empld	DeptName
Harry	3415	Finance
Sally	2241	Sales
George	3401	Finance
Harriet	2202	Sales

Dept

DeptName	Manager
Finance	George
Sales	Harriet
Production	Charles

Employee \bowtie Dept

Name	Empld	DeptName	Manager
Harry	3415	Finance	George
Sally	2241	Sales	Harriet
George	3401	Finance	George
Harriet	2202	Sales	Harriet

Vir: Wikipedia.

- ▶ “Sklic” med tabelami.
- ▶ Levi join, desni join, navaden join.

Paket *dplyr*

- ▶ Podpira operacije iz relacijske algebre.
- ▶ Učinkovita implementacija.
- ▶ Alternativa: uporaba paketa `data.table`
- ▶ <https://cran.rstudio.com/web/packages/dplyr/vignettes/introduction.html>
- ▶ <https://cran.r-project.org/web/packages/dplyr/vignettes/two-table.html>

Operatorji v *dplyr*

- ▶ Osnovni relacijski operatorji:
 - ▶ `filter(p)` - $\sigma_p(R)$
 - ▶ `select(a_1, a_2, \dots, a_n)` - $\pi_{a_1, a_2, \dots, a_n}(R)$
 - ▶ `rename()` - $\rho_{a/b}(R)$
 - ▶ `union(x, y)` - $R \cup S$
 - ▶ `setdiff(x, y)` - $R \setminus S$
 - ▶ `inner_join(x, y)` - združitev po skupnih stolpcih
- ▶ Dodatni praktično uporabni operatorji
 - ▶ `arrange(...)` - urejanje vrstic glede na izbrane stolpce
 - ▶ `mutate(...)` - preimenovanje stolpcev in dodajanje novih, ki so funkcije obstoječih.
 - ▶ `distinct()` - ohranjanje enoličnih vrstic
 - ▶ `summarize(...)` - uporaba združevalne funkcije na nekem stolpcu
 - ▶ `group_by(...)` - združevanje po vrsticah glede na enake vrednosti v stolpcih

Popravljanje stolpca LETNIK

- ▶ V stolpcu LETNIK bi želeli krajši zapis
- ▶ Poglejmo kaj imamo v stolpcu (histogram)

```
table(podatki$LETNIK)
```

- ▶ Obstoječa imena v stolpcu LETNIK

```
> imena <- c("1. letnik", "2. letnik", "3. letnik",  
"4. letnik", "5. letnik", "6. letnik", "Absolventi")
```

- ▶ Radi bi jih poenostavili imena v:

```
> letniki <- c("1", "2", "3", "4", "5", "6", "Abs")
```

Popravljanje stolpca LETNIK

- ▶ Ustvarimo “relacijo” s temi dvema stolpci.

```
> tab2 <- data.frame(  
  letnik=letniki,  
  ime=imena,  
  stringsAsFactors = FALSE)
```

- ▶ Uporabimo operacijo `inner_join` po stolpcih LETNIK IN ime

```
> require(dplyr)  
> zdruzena <- podatki %>%  
  inner_join(tab2, c("LETNIK"="ime"))
```

- ▶ Lahko bi še odstranili stolpec LETNIK in stolpec letnik preimenovali ...

Poizvedbe

- ▶ Uporabi operacije iz relacijske algebre pravimo *poizvedba*.
- ▶ Ime izhaja iz relacijskih podatkovnih baz in jezika SQL, ki izvaja operacije in iz obstoječih tabel (relacij) preko operacij proizvajajo nove tabele (relacije).
- ▶ Vrni tabelo z vrsticami, ki pripadajo ženskam.

```
> filter(zdruzena, SPOL=="Ženske")
```

- ▶ Ekvivalentno operatorju [pogoj,]
- ▶ Bolj po dplyr-jevske:

```
> zdruzena %>% filter(SPOL=="Ženske")
```

Poizvedbe

- ▶ Vse vrstice, v katerih so ženske vpisane po letu 2011.

```
> združena %>% filter(SPOL=="Ženske" &  
  STUDIJSKO_LETO > 2011)
```

- ▶ Namesto operatorja "&" lahko pogoje ločimo z vejico.

```
> združena %>% filter(SPOL=="Ženske",  
  STUDIJSKO_LETO > 2011)
```


Operacija arrange

- ▶ Uredi po stolpcu ST_STUDENTOV.

```
> zdruzena %>% arrange(ST_STUDENTOV)
```

- ▶ Uredi po stolpcih STUDIJSKO_LETO in potem po stolpcu ST_STUDENTOV, in sicer padajoče.

```
> zdruzena %>% arrange(STUDIJSKO_LETO, desc(ST_STUDENTOV))
```

Operaciji select in rename

- ▶ Izberi samo stolpce STUDIJSKO_LETO, ST_STUDENTOV in SPOL

> združena %>%

```
select(STUDIJSKO_LETO, ST_STUDENTOV, SPOL)
```

- ▶ Ob tem še preimenuj stolpec STUDIJSKO_LETO v LETO.

> združena %>%

```
select(LETO=STUDIJSKO_LETO, ST_STUDENTOV, SPOL)
```

- ▶ Preimenuj stolpec STUDIJSKO_LETO v LETO

> združena %>% rename(LETO=STUDIJSKO_LETO)

Združevanje po vrsticah

- ▶ Za katera leta imamo podatke?

```
> zdruzena %>%  
  select(STUDIJSKO_LETO) %>%  
  distinct()
```

- ▶ Koliko študentov je bilo vpisanih vsako leto?

```
> zdruzena %>%  
  group_by(STUDIJSKO_LETO) %>%  
  summarize(VPIS=sum(ST_STUDENTOV, na.rm=TRUE))
```

Združevanje po vrsticah

- ▶ Najprej smo združili vrstice po istih vrednostih v stolpcu `STUDIJSKO_LETO`, potem pa uporabili združevalno funkcijo na nekem od preostalih stolpcev.
- ▶ V rezultatu so le smiselni stolpci.
- ▶ Združevalne funkcije: `min(x)`, `max(x)`, `mean(x)`, `sum(x)`, `sd(x)`, `median(x)`, `n(x)`, `n_distinct(x)`, `first(x)`, `last(x)` in `nth(x, n)`

Združevanje

- ▶ Koliko je bilo vpisanih po spolih za posamezna leta?

```
> zdruzena %>%  
  group_by(SPOL, STUDIJSKO_LETO) %>%  
  summarize(VPIS=sum(ST_STUDENTOV, na.rm=TRUE))
```

Združevanje

- Koliko žensk in koliko moških je bilo vpisanih na posameznih vrstah študija na univerzi in kakšni so njihovi deleži?

```
> zdruzena %>%  
  filter(VISOKOSOLSKI_ZAVOD == "Univerze - SKUPAJ") %>%  
  select(VRSTA_IZOBRZEVANJA, SPOL, ST_STUDENTOV) %>%  
  group_by(VRSTA_IZOBRZEVANJA, SPOL) %>%  
  summarize(STEVILO=sum(ST_STUDENTOV, na.rm=TRUE)) %>%  
  pivot_wider(names_from=SPOL, values_from=STEVILO) %>%  
  arrange(VRSTA_IZOBRZEVANJA) %>%  
  mutate(  
    deležMoški=round(Moški/(Moški + Ženske), 2),  
    deležŽenske=round(Ženske/(Moški + Ženske), 2)  
  )
```

Organizacija podatkov

- ▶ Podatke si skušamo organizirati v obliko, ki se ji reče Tidy data.
- ▶ Stolpci lahko predstavljajo spremenljivke ali meritve.
- ▶ Spremenljivke opisujejo parametre pri katerih je izvedene meritev ("dimenzije").
- ▶ Definicija "Tidy data":
 - ▶ vsaka spremenljivka tvori stolpec,
 - ▶ za vsako meritev imamo eno vrstico,
 - ▶ vsak tip meritve je v ločeni tabeli.

Najbolj pogosti problemi

- ▶ Imena stolpcev so vrednosti, ne pa imena spremenljivk.
- ▶ V enem stolpcu hranimo več spremenljivk.
- ▶ Spremenljivke hranimo tako v vrsticah kot v stolpcih.
- ▶ Več vrst meritev (podatkov) v eni tabeli.
- ▶ Več istovrstnih meritev v večih tabelah.
- ▶ Hadley Wickham, Tidy Data, Journal of Statistical Software, August 2014, Volume 59, Issue 10

Imena stolpcev so vrednosti

religion	<\$10k	\$10–20k	\$20–30k	\$30–40k	\$40–50k	\$50–75k
Agnostic	27	34	60	81	76	137
Atheist	12	27	37	52	35	70
Buddhist	27	21	30	34	33	58
Catholic	418	617	732	670	638	1116
Don't know/refused	15	14	15	11	10	35
Evangelical Prot	575	869	1064	982	881	1486
Hindu	1	9	7	9	11	34
Historically Black Prot	228	244	236	238	197	223
Jehovah's Witness	20	27	24	24	21	30
Jewish	19	19	25	25	30	95

row	a	b	c
A	1	4	7
B	2	5	8
C	3	6	9

(a) Raw data

row	column	value
A	a	1
B	a	2
C	a	3
A	b	4
B	b	5
C	b	6
A	c	7
B	c	8
C	c	9

(b) Molten data

Operacija "pivot_longer"

- ▶ paket tidyr
- ▶ alternative gather (tidyr), melt (reshape2)

```
head(airquality)
```

##	Ozone	Solar.R	Wind	Temp	Month	Day
## 1	41	190	7.4	67	5	1
## 2	36	118	8.0	72	5	2
## 3	12	149	12.6	74	5	3
## 4	18	313	11.5	62	5	4
## 5	NA	NA	14.3	56	5	5
## 6	28	NA	14.9	66	5	6

Operacija "pivot_longer"

```
airquality %>%  
  pivot_longer(  
    c(-Month, -Day),  
    names_to="MERITEV",  
    values_to="VREDNOST"  
  ) %>% head
```

```
## # A tibble: 6 x 4  
##   Month   Day MERITEV VREDNOST  
##   <int> <int> <chr>      <dbl>  
## 1     5     1 Ozone       41  
## 2     5     1 Solar.R    190  
## 3     5     1 Wind        7.4  
## 4     5     1 Temp        67  
## 5     5     2 Ozone       36  
## 6     5     2 Solar.R    118
```

En stolpec več spremenljivk

country	year	column	cases
AD	2000	m014	0
AD	2000	m1524	0
AD	2000	m2534	1
AD	2000	m3544	0
AD	2000	m4554	0
AD	2000	m5564	0
AD	2000	m65	0
AE	2000	m014	2
AE	2000	m1524	4
AE	2000	m2534	4
AE	2000	m3544	6
AE	2000	m4554	5
AE	2000	m5564	12
AE	2000	m65	10
AE	2000	f014	3

(a) Molten data

country	year	sex	age	cases
AD	2000	m	0–14	0
AD	2000	m	15–24	0
AD	2000	m	25–34	1
AD	2000	m	35–44	0
AD	2000	m	45–54	0
AD	2000	m	55–64	0
AD	2000	m	65+	0
AE	2000	m	0–14	2
AE	2000	m	15–24	4
AE	2000	m	25–34	4
AE	2000	m	35–44	6
AE	2000	m	45–54	5
AE	2000	m	55–64	12
AE	2000	m	65+	10
AE	2000	f	0–14	3

(b) Tidy data

- Obdelavo nizov si bomo ogledali kasneje (regularni izrazi)

Spremenljivke v vrsticah in stolpcih

id	year	month	element	d1	d2	d3	d4	d5	d6	d7	d8
MX17004	2010	1	tmax	—	—	—	—	—	—	—	—
MX17004	2010	1	tmin	—	—	—	—	—	—	—	—
MX17004	2010	2	tmax	—	27.3	24.1	—	—	—	—	—
MX17004	2010	2	tmin	—	14.4	14.4	—	—	—	—	—
MX17004	2010	3	tmax	—	—	—	—	32.1	—	—	—
MX17004	2010	3	tmin	—	—	—	—	14.2	—	—	—
MX17004	2010	4	tmax	—	—	—	—	—	—	—	—
MX17004	2010	4	tmin	—	—	—	—	—	—	—	—
MX17004	2010	5	tmax	—	—	—	—	—	—	—	—
MX17004	2010	5	tmin	—	—	—	—	—	—	—	—

id	date	element	value
MX17004	2010-01-30	tmax	27.8
MX17004	2010-01-30	tmin	14.5
MX17004	2010-02-02	tmax	27.3
MX17004	2010-02-02	tmin	14.4
MX17004	2010-02-03	tmax	24.1
MX17004	2010-02-03	tmin	14.4
MX17004	2010-02-11	tmax	29.7
MX17004	2010-02-11	tmin	13.4
MX17004	2010-02-23	tmax	29.9
MX17004	2010-02-23	tmin	10.7

(a) Molten data

id	date	tmax	tmin
MX17004	2010-01-30	27.8	14.5
MX17004	2010-02-02	27.3	14.4
MX17004	2010-02-03	24.1	14.4
MX17004	2010-02-11	29.7	13.4
MX17004	2010-02-23	29.9	10.7
MX17004	2010-03-05	32.1	14.2
MX17004	2010-03-10	34.5	16.8
MX17004	2010-03-16	31.1	17.6
MX17004	2010-04-27	36.3	16.7
MX17004	2010-05-27	33.2	18.2

(b) Tidy data

Več vrst meritev v eni tabeli

year	artist	time	track	date	week	rank
2000	2 Pac	4:22	Baby Don't Cry	2000-02-26	1	87
2000	2 Pac	4:22	Baby Don't Cry	2000-03-04	2	82
2000	2 Pac	4:22	Baby Don't Cry	2000-03-11	3	72
2000	2 Pac	4:22	Baby Don't Cry	2000-03-18	4	77
2000	2 Pac	4:22	Baby Don't Cry	2000-03-25	5	87
2000	2 Pac	4:22	Baby Don't Cry	2000-04-01	6	94
2000	2 Pac	4:22	Baby Don't Cry	2000-04-08	7	99
2000	2Ge+her	3:15	The Hardest Part Of ...	2000-09-02	1	91
2000	2Ge+her	3:15	The Hardest Part Of ...	2000-09-09	2	87
2000	2Ge+her	3:15	The Hardest Part Of ...	2000-09-16	3	92
2000	3 Doors Down	3:53	Kryptonite	2000-04-08	1	81
2000	3 Doors Down	3:53	Kryptonite	2000-04-15	2	70
2000	3 Doors Down	3:53	Kryptonite	2000-04-22	3	68
2000	3 Doors Down	3:53	Kryptonite	2000-04-29	4	67
2000	3 Doors Down	3:53	Kryptonite	2000-05-06	5	66

- Vsaka meritev v svoji tabeli (normalizacija).

id	artist	track	time	id	date	rank
1	2 Pac	Baby Don't Cry	4:22	1	2000-02-26	87
2	2Ge+her	The Hardest Part Of ...	3:15	1	2000-03-04	82
3	3 Doors Down	Kryptonite	3:53	1	2000-03-11	72
4	3 Doors Down	Loser	4:24	1	2000-03-18	77
5	504 Boyz	Wobble Wobble	3:35	1	2000-03-25	87
6	98°0	Give Me Just One Nig...	3:24	1	2000-04-01	94
7	A*Teens	Dancing Queen	3:44	1	2000-04-08	99
8	Aaliyah	I Don't Wanna	4:15	2	2000-09-02	91
9	Aaliyah	Try Again	4:03	2	2000-09-09	87
10	Adams, Yolanda	Open My Heart	5:30	2	2000-09-16	92
11	Adkins, Trace	More	3:05	3	2000-04-08	81
12	Aguilera, Christina	Come On Over Baby	3:38	3	2000-04-15	70
13	Aguilera, Christina	I Turn To You	4:00	3	2000-04-22	68
14	Aguilera, Christina	What A Girl Wants	3:18	3	2000-04-29	67
15	Alice Deejay	Better Off Alone	6:50	3	2000-05-06	66

Več istovrstnih meritev v večih tabelah

- ▶ Npr. meritve za vsako leto, po osebah, ...
- ▶ Po potrebi dodamo stolpce, ki odražajo delitev in združimo v eno tabelo.
- ▶ Npr. delitev po letih: dodamo stolpec leto.