

# Praktikum iz programiranja 3



2024/25



# Pandas

- **Pandas** je Pythonova biblioteka koja se koristi efikasnije predstavljanje tabelarnih podataka.

```
pip install pandas
```

```
import pandas as pd
```

# DataFrame i rad sa kolinama tabela

```
import pandas as pd
podaci = [{"Ana", "ž", 13, 46, 160},
          {"Bojan", "m", 14, 52, 165},
          {"Vlada", "m", 13, 47, 157},
          {"Gordana", "ž", 15, 54, 165},
          {"Dejan", "m", 15, 56, 163},
          {"Đorđe", "m", 13, 45, 159},
          {"Elena", "ž", 14, 49, 161},
          {"Žaklina", "ž", 15, 52, 164},
          {"Zoran", "m", 15, 57, 167},
          {"Ivana", "ž", 13, 45, 158},
          {"Jasna", "ž", 14, 51, 162}]
```

```
tabela = pd.DataFrame(podaci)
print(tabela)
```

```
0 1 2 3 4
0 Ana ž 13 46 160
1 Bojan m 14 52 165
2 Vlada m 13 47 157
3 Gordana ž 15 54 165
4 Dejan m 15 56 163
5 Đorđe m 13 45 159
6 Elena ž 14 49 161
7 Žaklina ž 15 52 164
8 Zoran m 15 57 167
9 Ivana ž 13 45 158
10 Jasna ž 14 51 162
RangeIndex(start=0, stop=5, step=1)
>>>|
```

# DataFrame i rad sa kolinama tabela

Dodavanjem:

```
tabela.columns=["Ime", "Pol", "Starost", "Masa", "Visina"]  
print(tabela["Ime"])  
print(tabela.columns)
```

	Ime	Pol	Starost	Masa	Visina
0	Ana	ž	13	46	160
1	Bojan	m	14	52	165
2	Vlada	m	13	47	157
3	Gordana	ž	15	54	165
4	Dejan	m	15	56	163
5	Đorđe	m	13	45	159
6	Elena	ž	14	49	161
7	Žaklina	ž	15	52	164
8	Zoran	m	15	57	167
9	Ivana	ž	13	45	158
10	Jasna	ž	14	51	162

## Kolona Ime

```
0      Ana  
1     Bojan  
2     Vlada  
3  Gordana  
4     Dejan  
5     Đorđe  
6     Elena  
7   Žaklina  
8     Zoran  
9     Ivana  
10    Jasna  
Name: Ime, dtype: object
```

## Lista naziva svih kolona

```
Index(['Ime', 'Pol', 'Starost', 'Masa', 'Visina'], dtype='object')
```

# DataFrame i rad sa kolonama tabela

Pozivom jedne od sledećih funkcija lako možemo da vršimo elementarnu analizu podataka koji su predstavljeni tabelom:

- **sum** -- računa zbir elemenata u koloni (suma);
- **mean** -- računa prosek elemenata u koloni (aritmetička sredina se na engleskom kaže arithmetic mean);
- **median** -- računa medijanu elemenata u koloni;
- **min** -- računa vrednost najmanjeg elementa u koloni (minimum);
- **max** -- računa vrednost najvećeg elementa u koloni (maksimum).

# DataFrame i rad sa kolinama tabela

```
print(tabela["Visina"].min())
print(tabela["Starost"].max())
print(tabela["Visina"].mean())
print(tabela["Visina"].median())
if tabela["Masa"].sum() <= 600:
    print("Mogu svi da stanu u lift.")
else:
    print("Ne. Zajedno su preteški.")
```

```
157
15
161.9090909090909
162.0
Mogu svi da stanu u lift.
```

# Pandas+matplotlib

- Histogramom predstaviti visinu dece u grupi

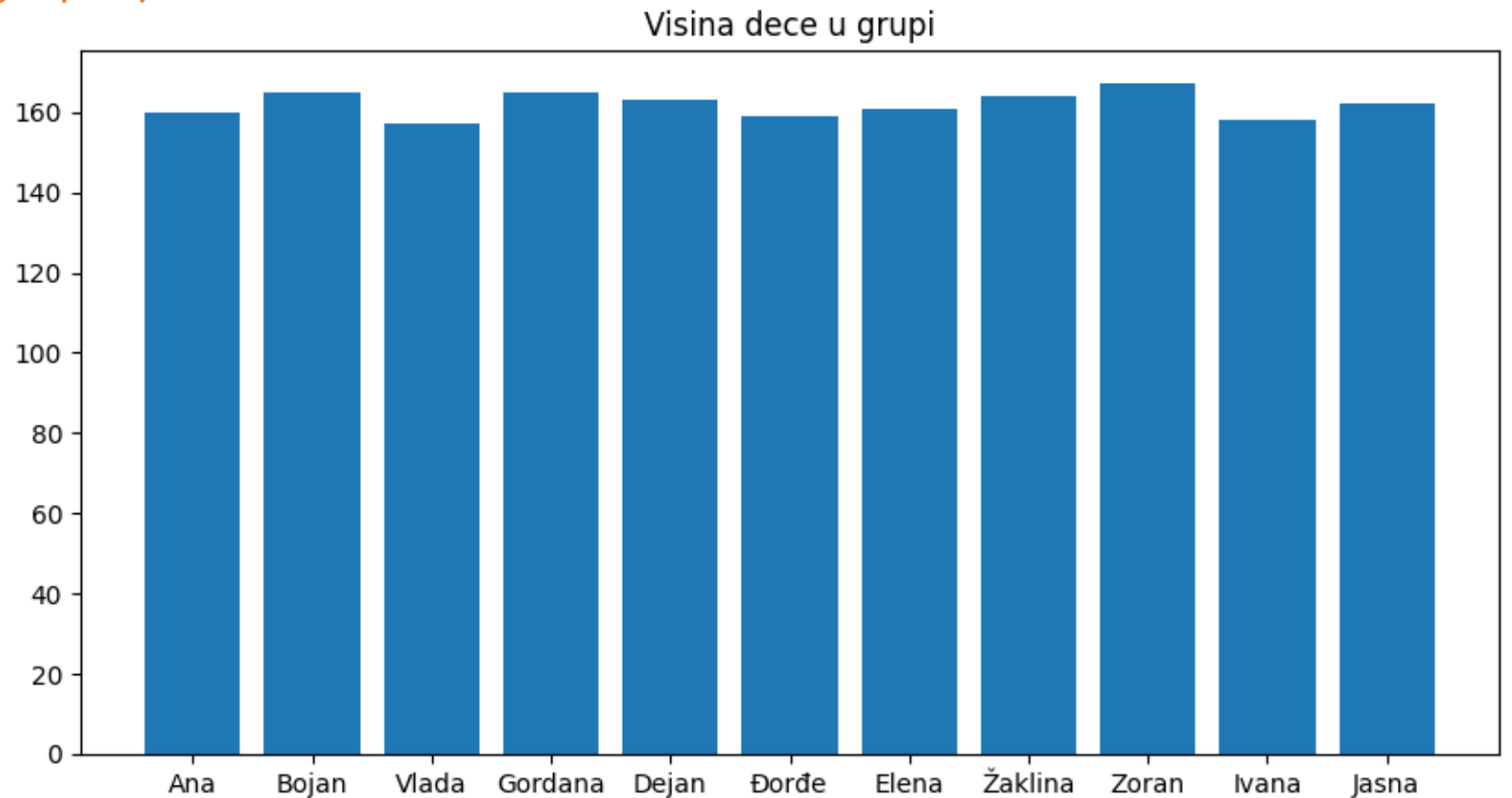
```
plt.figure(figsize=(10,5))
```

```
plt.bar(tabela["Ime"], tabela["Visina"])
```

```
plt.title("Visina dece u grupi")
```

```
plt.show()
```

```
plt.close()
```



# Pandas+matplotlib

- Predstaviti težinu i starost dece u grupi tako da težina bude predstavljena crvenim stubićima, a starost zelenom. Prikazati i legendu da bi bilo jasno šta koja boja predstavlja.

```
plt.figure(figsize=(10,5))
```

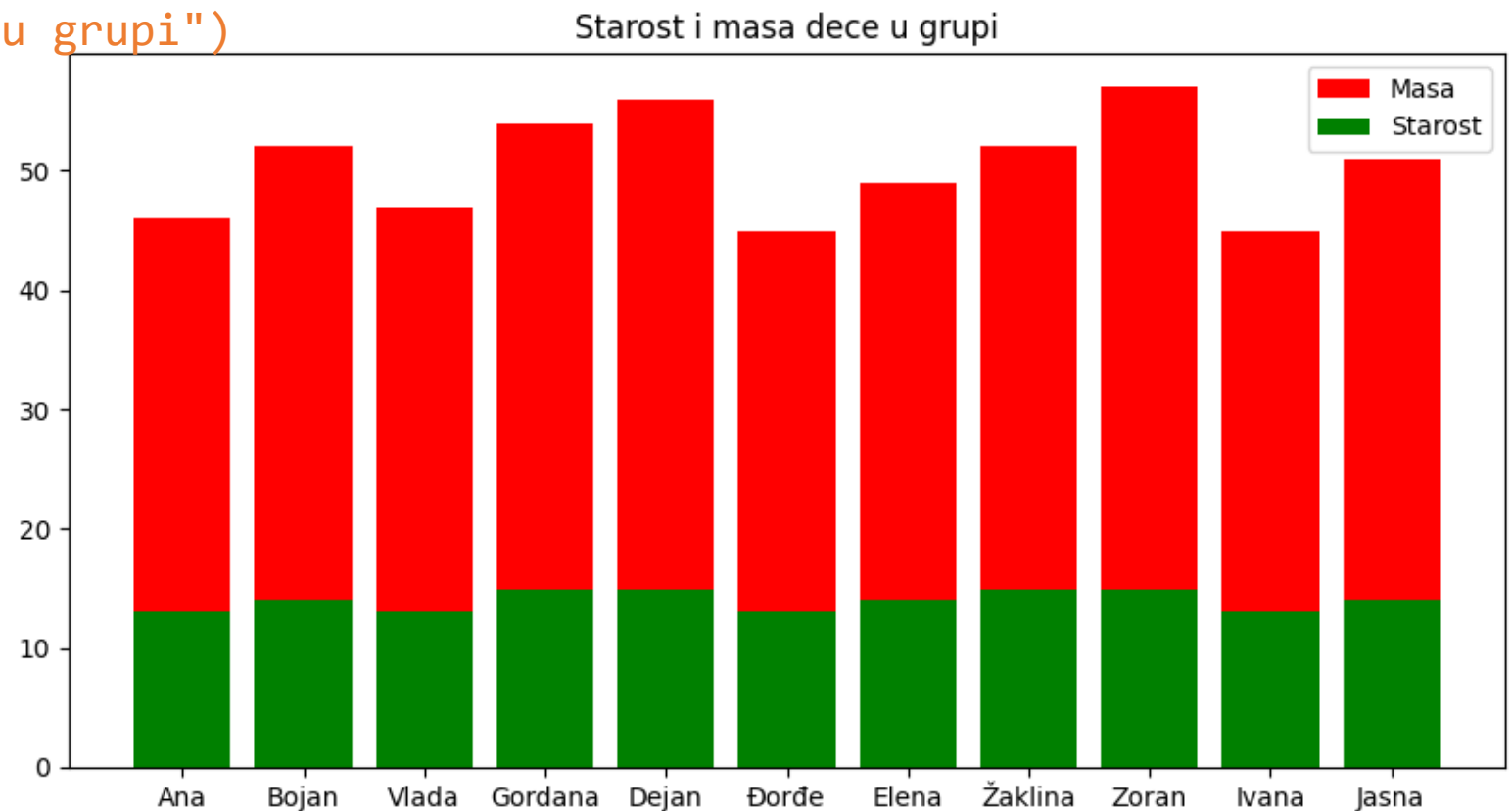
```
plt.bar(tabela["Ime"], tabela["Masa"], color="r", label="Masa")
```

```
plt.bar(tabela["Ime"], tabela["Starost"], color="g", label="Starost")
```

```
plt.title("Starost i masa dece u grupi")
```

```
plt.legend()
```

```
plt.show()
```





# Učitavanje podataka iz .csv datoteka

- Unošenje podataka u tabele je najkompleksniji posao obrade podataka
- Prekucavanje podataka
- Moderna obrada podataka se sve više usmerava na analizu ogromnih količina podataka (big data)
- Podaci se sakupljaju automatski i programi za prikupljanje podataka generišu velike tabele podataka koje posle treba obrađivati
- Najjednostavniji format za tabelarno predstavljanje podataka je CSV
- CSV datoteka je tekstualna datoteka u kojoj redovi odgovaraju redovima tabele, a podaci unutar istog reda su razdvojeni zarezima
- Posmatrajmo fajl *Top 25 YouTubers.csv*

```
RANK,GRADE,NAME,VIDEOS,SUBSCRIBERS,VIEWES  
1,A++,T-Series,13629,105783888,76945588449  
2,A,PewDiePie,3898,97853589,22298927681  
3,A+,5-Minute Crafts,3341,58629572,14860695079  
4,A++,Cocomelon - Nursery Rhymes,441,53163816,33519273951  
5,A++,SET India,31923,51784081,36464793233  
6,A+,Canal KondZilla,1100,50560964,25446405744
```

...

# Učitavanje podataka iz .csv datoteka

- Kako podatke iz .csv fajla učitati u pandas tabelu?

```
import pandas as pd
```

```
Top25 = pd.read_csv("Top 25 YouTubers.csv")
```

```
print(Top25)
```

	RANK	GRADE	NAME	VIDEOS	SUBSCRIBERS	VIEWES
0	1	A++	T-Series	13629	105783888	76945588449
1	2	A	PewDiePie	3898	97853589	22298927681
2	3	A+	5-Minute Crafts	3341	58629572	14860695079
3	4	A++	Cocomelon - Nursery Rhymes	441	53163816	33519273951
4	5	A++	SET India	31923	51784081	36464793233
5	6	A+	Canal KondZilla	1100	50560964	25446405744
6	7	A+	WWE	42404	46098586	34085586984
7	8	B+	Justin Bieber	134	45873439	625649566
8	9	A	Dude Perfect	209	43796634	8354321862
9	10	A+	Badabun	4406	41131131	13175713909
10	11	A+	Ed Sheeran	151	40883164	17420235764
11	12	A+	Zee Music Company	4305	40675210	18597258212
12	13	B+	HolaSoyGerman.	137	39549835	3963774207
13	14	B	EminemMusic	117	38321579	723182792
14	15	A	Marshmello	323	36937062	6495653129
15	16	B+	Ariana Grande	128	36748165	852378143
16	17	B+	whinderssonnunes	363	36059417	2965932071
17	18	A	JuegaGerman	1635	35648722	9524890177
18	19	A-	elrubiusOMG	809	35324033	7772447040
19	20	B	Taylor Swift	166	34920060	255089844
20	21	B-	Katy Perry	97	34416819	361332307
21	22	A	Felipe Neto	1872	33549096	7458531306
22	23	A	JustinBieberVEVO	122	33514535	18873475304
23	24	A	Fernanfloo	534	33378699	7201866552
24	25	A	TheEllenShow	10542	33362512	16519572219

Tabele mogu imati dosta podataka  
Ne štampati celu tabelu.

# Učitavanje podataka iz .csv datoteka

- Prikazati prvih 5 redova, ili proizvoljan broj redova

```
import pandas as pd
```

```
Top25 = pd.read_csv("Top 25 YouTubers.csv")
```

```
print(Top25.head())
```

```
print(Top25.head(7))
```

```
      RANK  GRADE      NAME  VIDEOS  SUBSCRIBERS  VIEWS
0        1   A++    T-Series   13629   105783888  76945588449
1        2     A   PewDiePie    3898    97853589  22298927681
2        3   A+   5-Minute Crafts   3341    58629572  14860695079
3        4  A++ Cocomelon - Nursery Rhymes    441    53163816  33519273951
4        5  A++      SET India   31923    51784081  36464793233
```

```
      RANK  GRADE      NAME  VIDEOS  SUBSCRIBERS  VIEWS
0        1   A++    T-Series   13629   105783888  76945588449
1        2     A   PewDiePie    3898    97853589  22298927681
2        3   A+   5-Minute Crafts   3341    58629572  14860695079
3        4  A++ Cocomelon - Nursery Rhymes    441    53163816  33519273951
4        5  A++      SET India   31923    51784081  36464793233
5        6   A+   Canal KondZilla    1100    50560964  25446405744
6        7   A+      WWE       42404    46098586  34085586984
```

# Učitavanje podataka iz .csv datoteka

- Prikazati poslednjih 5 redova, ili proizvoljan broj poslednjih redova

```
import pandas as pd
```

```
Top25 = pd.read_csv("Top 25 YouTubers.csv")
```

```
print(Top25.tail())
```

```
print(Top25.tail(7))
```

	RANK	GRADE	NAME	VIDEOS	SUBSCRIBERS	VIEWES
20	21	B-	Katy Perry	97	34416819	361332307
21	22	A	Felipe Neto	1872	33549096	7458531306
22	23	A	JustinBieberVEVO	122	33514535	18873475304
23	24	A	Fernanfloo	534	33378699	7201866552
24	25	A	TheEllenShow	10542	33362512	16519572219

	RANK	GRADE	NAME	VIDEOS	SUBSCRIBERS	VIEWES
18	19	A-	elrubiusOMG	809	35324033	7772447040
19	20	B	Taylor Swift	166	34920060	255089844
20	21	B-	Katy Perry	97	34416819	361332307
21	22	A	Felipe Neto	1872	33549096	7458531306
22	23	A	JustinBieberVEVO	122	33514535	18873475304
23	24	A	Fernanfloo	534	33378699	7201866552
24	25	A	TheEllenShow	10542	33362512	16519572219

# Prikaz učitanih podataka

- Prikazati podatke o broju pretplatnika stubičastim dijagramom

```
import pandas as pd
```

```
import matplotlib.pyplot as plt
```

```
Top25 = pd.read_csv("Top 25 YouTubers.csv")
```

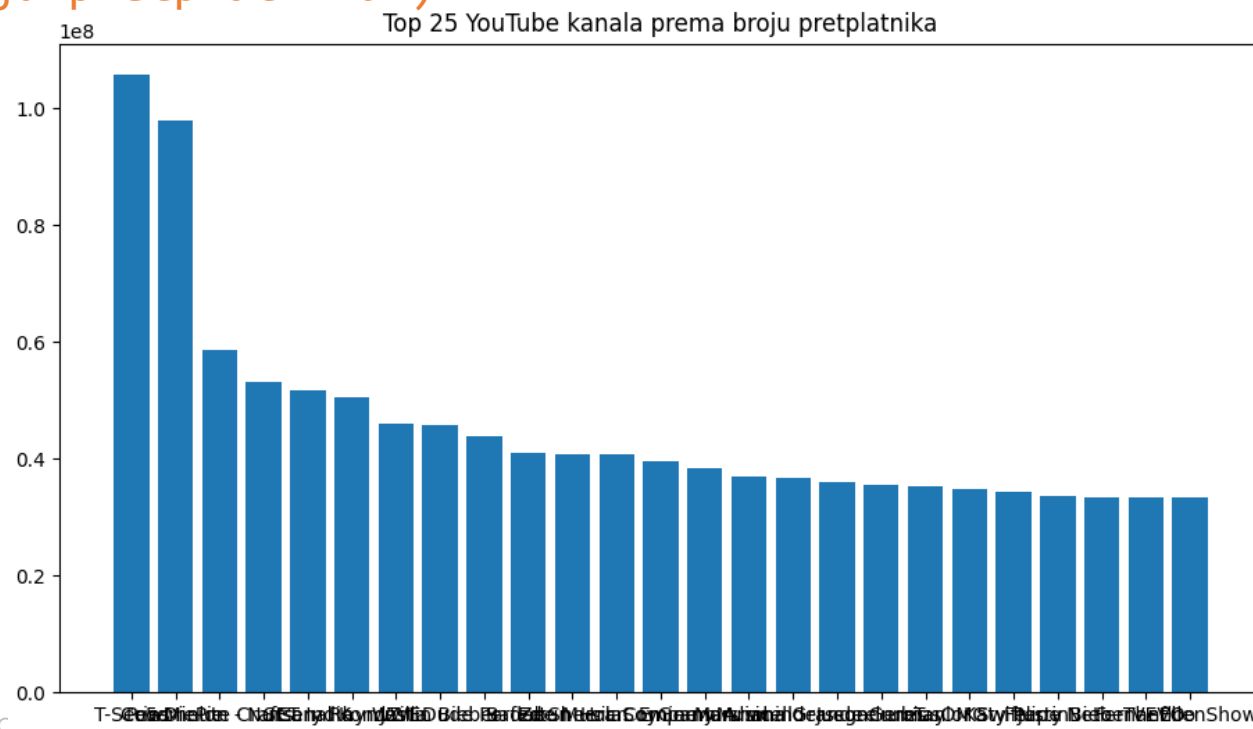
```
plt.figure(figsize=(15,10))
```

```
plt.bar(Top25["NAME"], Top25["SUBSCRIBERS"])
```

```
plt.title("Top 25 YouTube kanala prema broju pretplatnika")
```

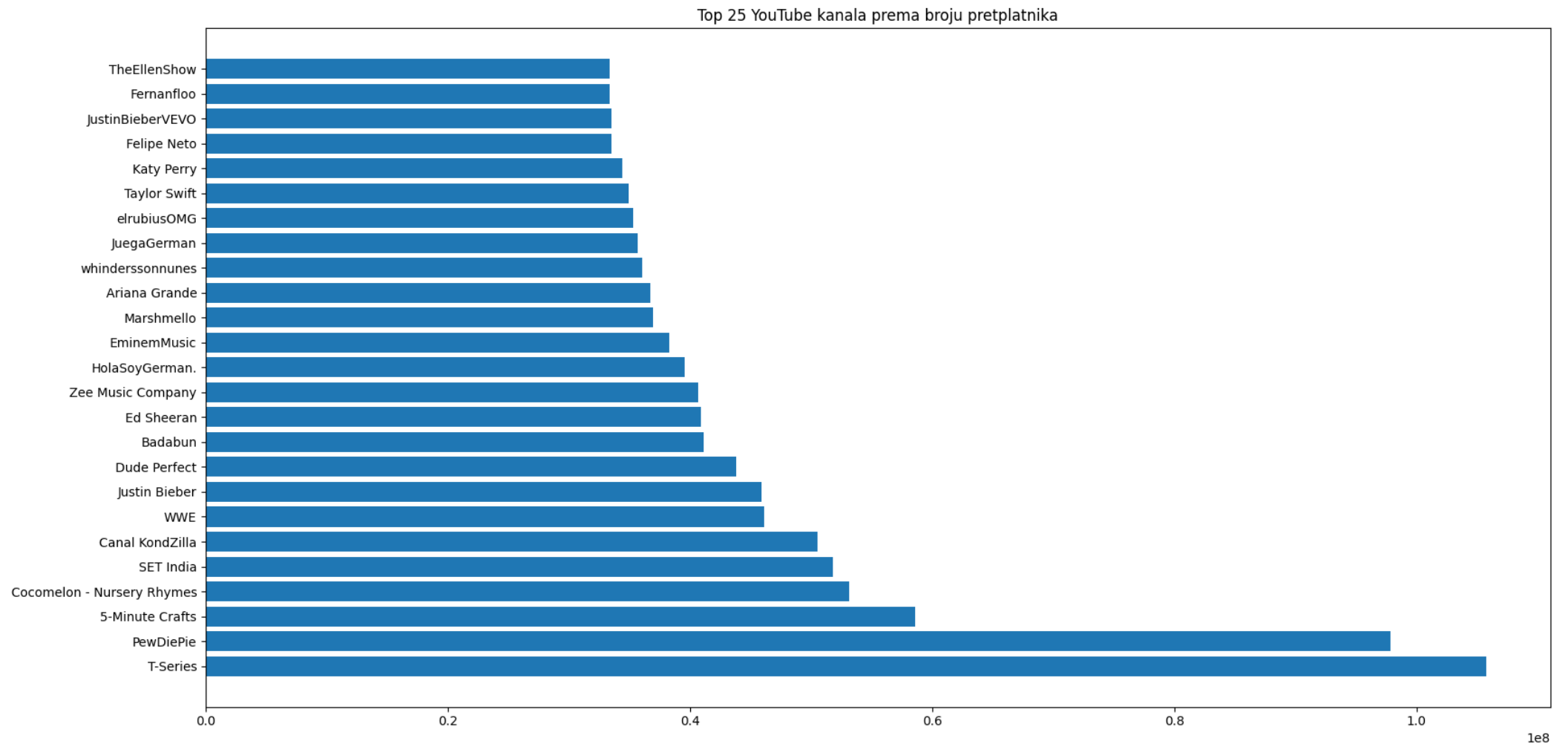
```
plt.show()
```

```
plt.close()
```



# Prikaz učitanih podataka

- Ne vide se nazivi kanala, zato umesto funkcije `bar`, koristi se funkcija `barh`  
`plt.barh(Top25["NAME"], Top25["SUBSCRIBERS"])`



# Učitavanje podataka iz .csv datoteka

- Posmatrajmo fajl *StanovnistvoSrbije2017.csv*
- Zaglavlje tabele sadrži kolone (Старост, Мушко, Женско)
- Zatim ćemo u tabelu dodati novu kolonu koja predstavlja odnos muškaraca i žena po uzrastima

```
import pandas as pd
import matplotlib.pyplot as plt
stanovnistvo = pd.read_csv("StanovnistvoSrbije2017.csv")
stanovnistvo1 = stanovnistvo.set_index("Старост")

stanovnistvo1["М/Ж"] = 0.0
for i in stanovnistvo1.index:
    stanovnistvo1.loc[i, "М/Ж"] = stanovnistvo1.loc[i, "Мушко"] /
    stanovnistvo1.loc[i, "Женско"]
print(stanovnistvo1.head(10))
```

	Мушко	Женско	М/Ж
Старост			
0	33145	31444	1.054096
1	33252	31105	1.069024
2	33807	31475	1.074091
3	34076	31952	1.066475
4	33436	31643	1.056663
5	34278	32505	1.054545
6	33773	31523	1.071376
7	33892	32185	1.053037
8	34706	32396	1.071305
9	34519	32177	1.072785

# Učitavanje podataka iz .csv datoteka

- Dodata je linija  $y=1$  kako bi se uočilo u kom trenutku broj muškaraca postaje manji od broja žena

```
plt.figure(figsize=(20,5))
```

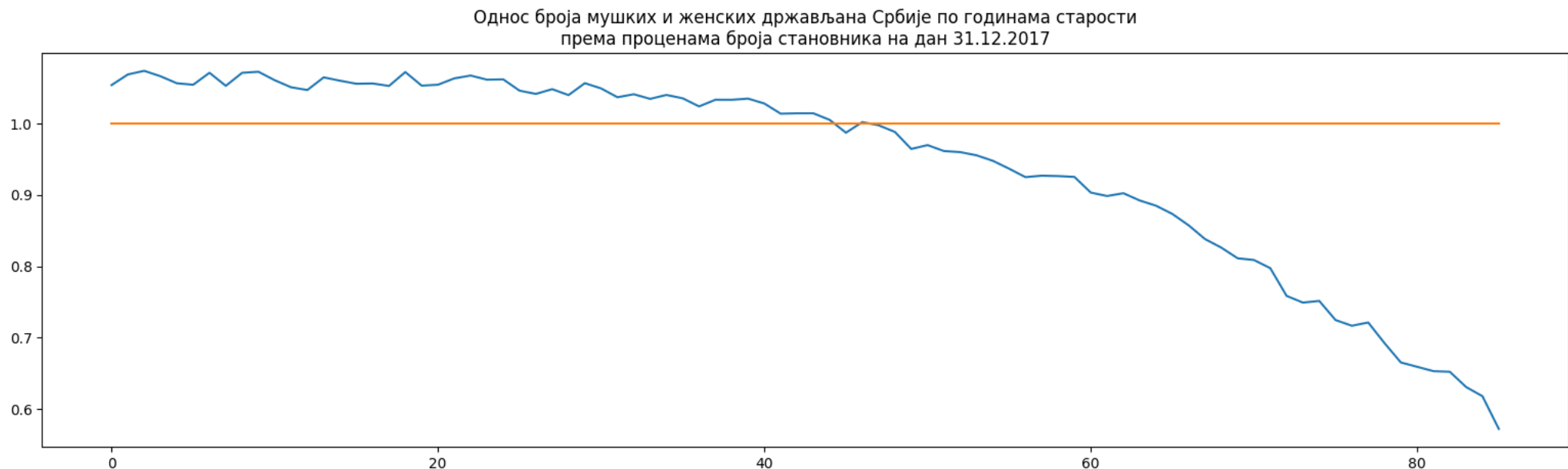
```
plt.plot(stanovništvo1.index, stanovništvo1["M/Ж"])
```

```
plt.plot(stanovništvo1.index, [1.0] * len(stanovništvo1.index))
```

```
plt.title("Однос броја мушких и женских држављана Србије по годинама старости\nпрема проценама броја становника на дан 31.12.2017")
```

```
plt.show()
```

```
plt.close()
```





# Pisanje podataka u .csv datoteka

- Kako sada izmenjenu tabelu upisati u nov .csv fajl?

```
stanovnistvo1.to_csv("StanovnistvoSrbije2017-Novo.csv", encoding="utf-8")
```

Prvi podatak koji se prosleđuje funkciji `to_csv` predstavlja ime nove datoteke u koju će biti upisani podaci, dok argument `encoding="utf-8"` znači da će podaci biti upisani u datoteku koristeći sistem za kodiranje koji se zove UTF-8. Ovo moramo da navedemo zato što u tabeli imamo podatke koji su zapisani ćirilicom. Čak i da smo koristili latinicu morali bismo da koristimo UTF-8 sistem za kodiranje zbog slova kao što su č, ć, š, ž, đ, a koja ne postoje u engleskom alfabetu. Argument `encoding="utf-8"` možemo da izostavimo samo ako znamo da su za zapisivanje podataka u tabeli korišćeni isključivo simboli iz engleskog alfabeta.

```
Старост,Мушко,Женско,М/Ж
0,33145,31444,1.0540961709706145
1,33252,31105,1.06902427262498
2,33807,31475,1.074090548054011
3,34076,31952,1.0664747120681022
4,33436,31643,1.0566634010681668
5,34278,32505,1.0545454545454545
6,33773,31523,1.071376455286616
7,33892,32185,1.0530371290974057
8,34706,32396,1.071305099394987
9,34519,32177,1.0727849084749976
10,34017,32064,1.0609094311377245
11,34947,33251,1.0510059847824125
-----
```

# Primer

- U tabeli Razred.xlsx nalaze se ocene učenika jednog razreda iz informatike. Podaci su realni, pa su zato anonimizirani (imena učenika su Učenik 1, Učenik 2, itd). Tabela ima zaglavlje, a tekst je unet latiničnim pismom.
  - a) Učitaj ovu tabelu u strukturu podataka DataFrame i prikaži prvih nekoliko redova da razumeš strukturu tabele. (Kolone označene sa "K" predstavljaju ocene iz kontrolnog zadatka, kolone označene sa "P" predstavljaju ocenu iz pismenog zadatka (ovo je latinično slovo P), kolone označene sa "U" predstavljaju ocenu iz usmene provere, a kolona "D" ocenu iz domaćih zadataka.)
  - b) Indeksiraj tabelu kolonom "Ime".
  - c) Izračunaj i ispiši prosečnu ocenu na svakom od tri pismena zadatka (kolone "P1", "P2" i "P3").
  - d) Dodaj tabeli novu kolonu "Prosek" i onda za svakog učenika izračunaj prosek ocena i upiši dobijenu vrednost u ovu kolonu tabele.
  - e) Dodaj tabeli novu kolonu "Ocena" i onda za svakog učenika izračunaj zaključnu ocenu na osnovu proseka, i upiši tu ocenu u ovu kolonu tabele.
  - f) Dobijenu tabelu upiši u novu datoteku Razred-Ocene.xlsx vodeći računa o tome da tabela sadrži slova specifična za srpski jezik.

# Primer

- a) Učitaj ovu tabelu u strukturu podataka DataFrame i prikaži prvih nekoliko redova da razumeš strukturu tabele. (Kolone označene sa "K" predstavljaju ocene iz kontrolnog zadatka, kolone označene sa "P" predstavljaju ocenu iz pismenog zadatka (ovo je latinično slovo P), kolone označene sa "U" predstavljaju ocenu iz usmene provere, a kolona "D" ocenu iz domaćih zadataka.)

Funkciji za učitavanje Eksel tabele pored imena datoteke treba dati i ime radnog lista sa koga se učitava tabela. Ukoliko se ne navede ime radnog lista funkcija će učitati tabelu iz prvog radnog lista na koji naiđe. Ovo obično koristimo samo u situacijama kada smo sigurni da Eksel radna sveska ima samo jedan radni list.

```
pip install openpyxl
```

```
import pandas as pd
ocene = pd.read_excel("Razred.xlsx")
print(ocene.head(10))
```

	Ime	D	K1	K2	P1	U1	K3	K4	K5	P2	U2	K6	P3	K7	U3
0	Učenik 1	5	1	3	3	4	2	3	3	1	3	2	1	3	4
1	Učenik 2	1	2	3	1	3	2	2	3	1	2	1	2	2	2
2	Učenik 3	5	4	5	4	5	4	5	4	4	3	5	4	5	5
3	Učenik 4	5	5	4	5	5	5	5	5	5	5	5	5	5	5
4	Učenik 5	1	3	3	3	5	5	5	5	5	5	5	5	3	4
5	Učenik 6	5	4	5	5	5	4	5	5	5	5	5	5	5	5
6	Učenik 7	5	4	5	4	5	4	5	5	5	5	5	4	5	5
7	Učenik 8	5	5	4	5	5	4	5	5	4	5	5	5	5	5
8	Učenik 9	5	5	5	5	5	5	5	5	5	5	5	5	5	5
9	Učenik 10	5	5	4	5	5	5	5	5	5	4	4	5	5	5

# Primer

b) Indeksiraj tabelu kolonom "Ime".

```
ocene.set_index("Ime", inplace = True)
```

	D	K1	K2	P1	U1	K3	K4	K5	P2	U2	K6	P3	K7	U3
Ime														
Učenik 1	5	1	3	3	4	2	3	3	1	3	2	1	3	4
Učenik 2	1	2	3	1	3	2	2	3	1	2	1	2	2	2
Učenik 3	5	4	5	4	5	4	5	4	4	3	5	4	5	5
Učenik 4	5	5	4	5	5	5	5	5	5	5	5	5	5	5
Učenik 5	1	3	3	3	5	5	5	5	5	5	5	5	3	4
Učenik 6	5	4	5	5	5	4	5	5	5	5	5	5	5	5
Učenik 7	5	4	5	4	5	4	5	5	5	5	5	4	5	5
Učenik 8	5	5	4	5	5	4	5	5	4	5	5	5	5	5
Učenik 9	5	5	5	5	5	5	5	5	5	5	5	5	5	5
Učenik 10	5	5	4	5	5	5	5	5	5	4	4	5	5	5

c) Izračunaj i ispiši prosečnu ocenu na svakom od tri pismena zadatka (kolone "P1", "P2" i "P3").

```
for i in ocene.index:
```

```
    print((ocene.loc[i, "P1"]+ocene.loc[i, "P2"]+ocene.loc[i, "P3"] )/ 3)
```

```
1.6666666666666667
1.3333333333333333
4.0
5.0
4.3333333333333333
5.0
4.3333333333333333
4.6666666666666667
5.0
5.0
4.6666666666666667
3.3333333333333335
1.0
4.6666666666666667
3.6666666666666665
5.0
4.6666666666666667
```

# Primer

- d) Dodaj tabeli novu kolonu "Prosek" i onda za svakog učenika izračunaj prosek ocena i upiši dobijenu vrednost u ovu kolonu tabele.

```
kolone_sa_ocenama = ocene.columns[1:]  
print(kolone_sa_ocenama)
```

```
ocene["Prosek"]=0.0  
for i in ocene.index:  
    sum=0.0  
    for j in kolone_sa_ocenama:  
        sum +=ocene.loc[i,j]  
    ocene.loc[i,"Prosek"] = sum/len(kolone_sa_ocenama)
```

	Ime	D	K1	K2	P1	U1	K3	K4	K5	P2	U2	K6	P3	K7	U3	Prosek
0	Učenik 1	5	1	3	3	4	2	3	3	1	3	2	1	3	4	2.714286
1	Učenik 2	1	2	3	1	3	2	2	3	1	2	1	2	2	2	1.928571
2	Učenik 3	5	4	5	4	5	4	5	4	4	3	5	4	5	5	4.428571
3	Učenik 4	5	5	4	5	5	5	5	5	5	5	5	5	5	5	4.928571
4	Učenik 5	1	3	3	3	5	5	5	5	5	5	5	5	3	4	4.071429
5	Učenik 6	5	4	5	5	5	4	5	5	5	5	5	5	5	5	4.857143

# Primer

- e) Dodaj tabeli novu kolonu "Ocena" i onda za svakog učenika izračunaj zaključnu ocenu na osnovu proseka, i upiši tu ocenu u ovu kolonu tabele.

```
def zaklj_ocena(prosek):  
    if prosek >= 4.50:  
        return 5  
    elif prosek >= 3.50:  
        return 4  
    elif prosek >= 2.50:  
        return 3  
    elif prosek >= 1.50:  
        return 2  
    else:  
        return 1
```

	Ime	D	K1	K2	P1	U1	K3	...	U2	K6	P3	K7	U3	Prosek	Ocena
0	Učenik 1	5	1	3	3	4	2	...	3	2	1	3	4	2.714286	3.0
1	Učenik 2	1	2	3	1	3	2	...	2	1	2	2	2	1.928571	2.0
2	Učenik 3	5	4	5	4	5	4	...	3	5	4	5	5	4.428571	4.0
3	Učenik 4	5	5	4	5	5	5	...	5	5	5	5	5	4.928571	5.0
4	Učenik 5	1	3	3	3	5	5	...	5	5	5	3	4	4.071429	4.0
5	Učenik 6	5	4	5	5	5	4	...	5	5	5	5	5	4.857143	5.0
-	...	-	-	-	-	-	-	...	-	-	-	-	-	-	-

```
for i in ocene.index:  
    ocene.loc[i,"Ocena"] = zaklj_ocena(ocene.loc[i,"Prosek"])
```

# Primer

- f) Dobijenu tabelu upiši u novu datoteku Razred-Ocene.xlsx vodeći računa o tome da tabela sadrži slova specifična za srpski jezik.

```
ocene.to_excel("Razred_ocene.xlsx", encoding="utf-8")
```

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	Ime	D	K1	K2	P1	U1	K3	K4	K5	P2	U2	K6	P3	K7	U3	Prosek	Ocena
2	Učenik 1	5	1	3	3	4	2	3	3	1	3	2	1	3	4	2.538462	3
3	Učenik 2	1	2	3	1	3	2	2	3	1	2	1	2	2	2	2	2
4	Učenik 3	5	4	5	4	5	4	5	4	4	3	5	4	5	5	4.384615	4
5	Učenik 4	5	5	4	5	5	5	5	5	5	5	5	5	5	5	4.923077	5
6	Učenik 5	1	3	3	3	5	5	5	5	5	5	5	5	3	4	4.307692	4
7	Učenik 6	5	4	5	5	5	4	5	5	5	5	5	5	5	5	4.846154	5
8	Učenik 7	5	4	5	4	5	4	5	5	5	5	5	4	5	5	4.692308	5
9	Učenik 8	5	5	4	5	5	4	5	5	4	5	5	5	5	5	4.769231	5
10	Učenik 9	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
11	Učenik 10	5	5	4	5	5	5	5	5	5	4	4	5	5	5	4.769231	5
12	Učenik 11	5	5	5	4	5	5	4	4	5	5	4	5	5	5	4.692308	5
13	Učenik 12	5	4	5	4	5	4	2	3	4	4	3	2	2	4	3.538462	4
14	Učenik 13	5	3	1	1	3	2	3	1	1	3	2	1	2	5	2.153846	2
15	Učenik 14	5	4	5	4	5	5	5	4	5	5	5	5	5	5	4.769231	5
16	Učenik 15	5	5	3	4	5	4	3	4	2	5	5	5	5	3	4.076923	4
17	Učenik 16	5	4	5	5	4	5	5	5	5	5	5	5	2	5	4.615385	5
18	Učenik 17	5	3	5	5	5	3	5	5	4	5	3	5	5	5	4.461538	4

Ukoliko je indeksna kolona ostala podrazumena 1,2... a nećemo da bude deo izlaznog fajla

```
ocene.to_excel("Razred_ocene.xlsx", encoding="utf-8", index=false)
```

# Preuzimanje podataka iz udaljenih resursa

- Moguće je preuzeti i podatke sa udaljenih resursa bez potrebe da se oni prvo prebace na lokalnu mašinu. Da bismo pristupili podatku koji se nalazi na nekoj drugoj mašini potrebno je da obe mašine imaju pristup Internetu i da znamo tačnu lokaciju podatka na udaljenoj mašini. Tačna lokacija bilo kog resursa na Internetu je opisana njegovim URL-om (od engl. Universal Resource Locator, što znači "Univerzalni lokator resursa").
- Na adresi
- [https://raw.githubusercontent.com/cs109/2014\\_data/master/countries.csv](https://raw.githubusercontent.com/cs109/2014_data/master/countries.csv)

se nalazi javno dostupan spisak svih država na svetu. Ovu tabelu možemo lako učitati naredbom `read_csv`:

```
drzave = pd.read_csv("https://raw.githubusercontent.com/cs109/2014_data/master/countries.csv")
print(drzave.head(10))
```

	Country	Region
0	Algeria	AFRICA
1	Angola	AFRICA
2	Benin	AFRICA
3	Botswana	AFRICA
4	Burkina	AFRICA
5	Burundi	AFRICA
6	Cameroon	AFRICA
7	Cape Verde	AFRICA
8	Central African Republic	AFRICA
9	Chad	AFRICA



# Preuzimanje podataka iz udaljenih resursa

- Pomoću naredbe `read_html` može se pročitati i tabela direktno iz HTML koda neke veb stranice. Recimo, sledeća naredba čita spisak svih federalnih jedinica Sjedinjenih Američkih Država sa odgovarajuće stranice Vikipedije:

```
pip install lxml
```

```
US = pd.read_html("https://simple.wikipedia.org/wiki/List_of_U.S._states", header=[0,1])[0]
print(US.head())
```

Na veb strani koju čitamo može biti više tabela i zato funkcija `read_html` vraća listu tabela. Tabela koju želimo da vidimo je prva na navedenoj strani i zato iza naredbe sledi konstrukt `[0]`. Argument `header=[0,1]` znači da prve dve vrste treba uzeti za zaglavlje tabele.

```
Flag, name and postal abbreviation [1] ... Number of Reps.
Flag, name and postal abbreviation [1] ... Number of Reps.
0                                     Alabama ...          7
1                                     Alaska ...          1
2                                     Arizona ...          9
3                                     Arkansas ...         4
4                                     California ...         52
```