pandas_intro

August 23, 2017

0.1 Pandas

Pandas je modul namenjen brzom i efikasnom radu sa podacima.

In [8]:
```
import pandas as pd
```

Za predstavljanje podataka se koriste dve strukture: Series i DataFrame

Series su strukture koje odgovaraju jednodimenzionim nizovima (čiji elementi mogu biti raznorodni) i koje imaju pridruženi niz obeležja tzv. indeks.

DataFrame su strukture koje odgovaraju tabelarnim podacima (eng. spreadsheet-like data structures) i koje imaju indekse i vrsta i kolona.

NaN označava nedostajuću vrednost. Saisnull i notnull se može proveriti da li je neka vrednost nedostajuća ili ne.

In [2]:
```
# stampa
#\?pd.Series
```

In [9]:
```
# kreiranje serije
points = pd.Series([(1, 1) , (2, 2), (3, 3)])
print(type(points[0]))
print(points)
import numpy as np
points1 = np.array([[1, 1] , (2, 2), (3, 3)])
print(type(points1[1]))
```

```
<class 'tuple'>
0    (1, 1)
1    (2, 2)
2    (3, 3)
dtype: object
<class 'numpy.ndarray'>
```

In [10]:
```
# ocitavanje elemenata serije
#print(points.values)
print(points.keys)
```

```
In [44]:    # ocitavanje indeksa kolona
            users.columns

Out[44]: Index(['age', 'location', 'name'], dtype='object')

In [45]:    # ocitavanje indeksa vrsta
            users.index

Out[45]: RangeIndex(start=0, stop=4, step=1)

In [46]:    # ispis vrednosti
            users.values

Out[46]: array([[24, 'Berlin', 'Anna'],
                [30, 'New York', 'John'],
                [33, 'Belgrade', 'Peter'],
                [21, 'London', 'Linda']], dtype=object)

In [47]:    # pristup pojedinacnim kolonama
            print(users['age'])
            print(users.age)

0    24
1    30
2    33
3    21
Name: age, dtype: int64
0    24
1    30
2    33
3    21
Name: age, dtype: int64


In [50]:    # pristup pojedinacnim vrstama
            users.ix[3]

Out[50]: age          21
         location     London
         name         Linda
         Name: 3, dtype: object

In [51]:    # dozvoljeno je i zadavanje opsega ako su u pitanju numericki indeksi
            users.ix[1:5]

Out[51]:    age  location  name
         1   30  New York  John
         2   33  Belgrade  Peter
         3   21  London    Linda
```

```
In [52]:    # pristup pojedinacnim elementima
            users.ix[1]['name']

Out[52]: 'John'

In [53]: type(users)

Out[53]: pandas.core.frame.DataFrame

In [54]: type(users['age'])

Out[54]: pandas.core.series.Series

In [55]: type(users.ix[1]['age'])

Out[55]: numpy.int64
```

Mogu se koristiti različiti metodi filtriranja podataka:

```
In [63]:    # izdvajanje korisnika koji imaju vise od 25 godina
            print(users[users.age\>25])
            print("    \n ")
            print(users.age\>25)

   age   location     name
1   30   New York     John
2   33   Belgrade   Peter


0     False
1      True
2      True
3     False
Name: age, dtype: bool
```

Postoji mogucnost promene indeksa bilo za serije bilo za tabelarne podatke. Ukoliko se dodaju nove vrste/kolone, njihove vrednosti su NaN (konstanta np.nan)

```
In [68]: users = users.reindex(columns = ['age', 'location', 'name', 'email'])
            print(users)

            users = users.reindex(index=[0, 1, 'x', 'y', 'z', 'w'])
            print(users)

   age   location     name   email
0   24     Berlin     Anna     NaN
1   30   New York     John     NaN
2   33   Belgrade   Peter     NaN
3   21     London   Linda     NaN
```

```
      age    location    name    email
0    24.0       Berlin    Anna       NaN
1    30.0    New  York     John       NaN
x     NaN             NaN      NaN       NaN
y     NaN             NaN      NaN       NaN
z     NaN             NaN      NaN       NaN
w      NaN             NaN      NaN       NaN
```

Provera nedostajucih vrednosti je moguca kroz isnull ili notnull

```
In [80]: pd.isnull(users['email'])
```

```
Out[80]: 0       True
         1       True
         2       True
         3       True
         Name: email, dtype: bool
```

Primer korišćenja raspoloživih podataka

```
In [15]:    # ucitavanje podataka iz CSV fajla
            # podaci su preuzeti sa https://www.kaggle.com/the-guardian/extinct-languages
            # i predstavljaju informacije o jezicima koji su ugrozeni ili u izumiranju.
            # Kategorije koje se razmatraju su:
            # vulnerable - jezik se uci u nekom uzem kontekstu
            # definitely endangered - jezik se ne uci kao maternji jezik
            # severely endangered - jezik govore pripadnici starijih generacija, dok ga njihovi potomci n
            # critically endangered - jezik govore samo pripadnici starijih generacija i to delimicno ili ja
            # extinct - jezici ciji govornici vise ne postoje

            languages = pd.read_csv('data/languages.csv')
```

```
In [70]:    # dodatne informacije o read_csv funkciji
            # stampa \?pd.read_csv

            # na raspolaganju su i funkcije za analiziranje Excel formata, JSON formata, SQL sadrzaja
```

```
In [71]:    # rezultat citanje je DataFrame struktura
            type(languages)
```

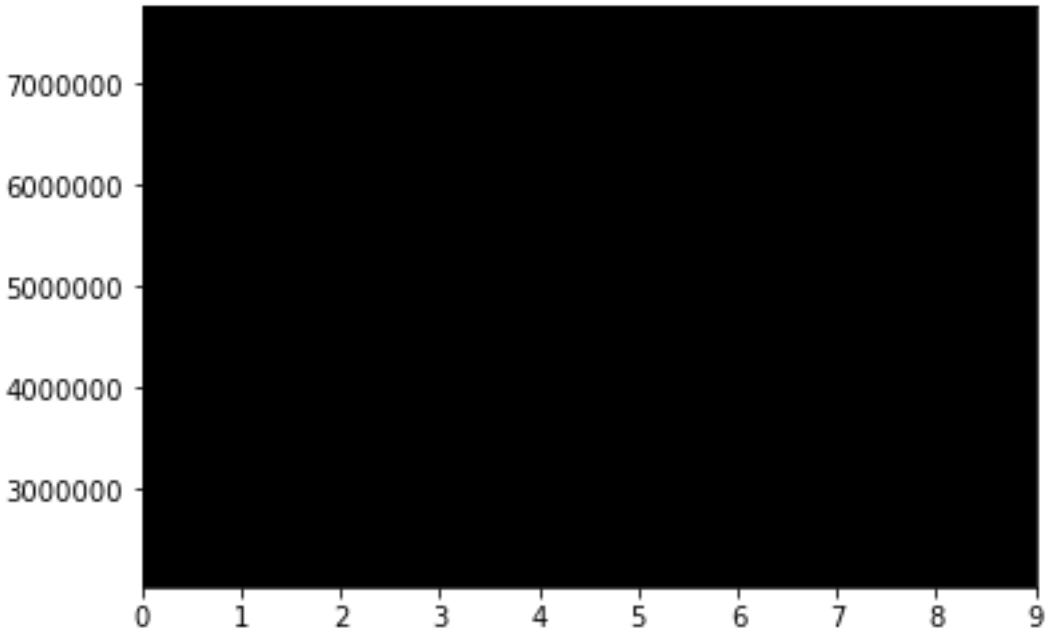
```
Out[71]: pandas.core.frame.DataFrame
```

```
In [72]:    # ispis broja vrsti ucitanog skupa podataka
            len(languages)
```

```
Out[72]: 2722
```

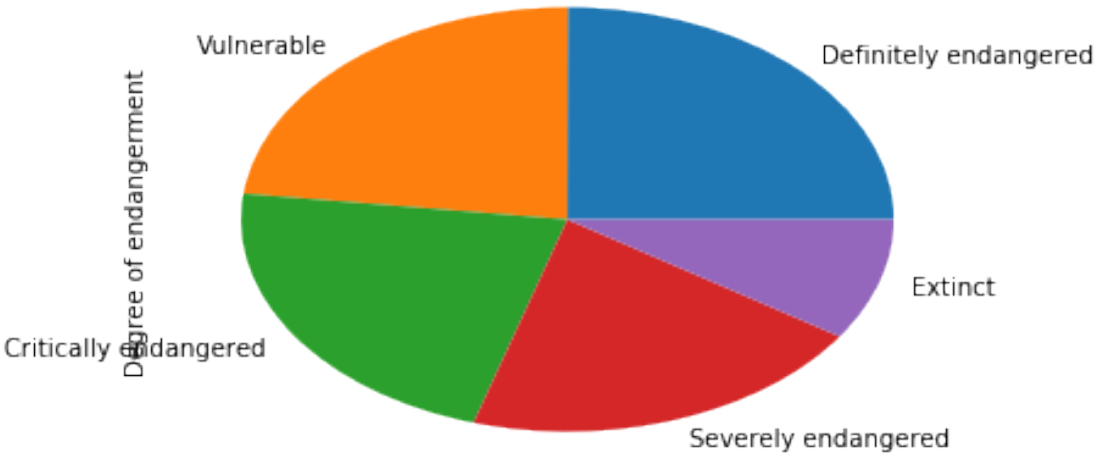
```
In [1]:    # ispis prvih 5 redova
           # stampa languages.head(5)
```

```
In [2]:   # ispis poslednjih 5 redova
          # stampa languages.tail(5)

In [3]:   # ispis zeljenog opsega redova
          # stampa languages[144:156]

In [76]:  # izdvajanje kolone sa zadatim imenom
          languages['Name in English'][:14:2]

Out[76]: 0                    South Italian
         2                        Low Saxon
         4                          Lombard
         6                  Yiddish (Israel)
         8           Limburgian-Ripuarian
         10                         Kumaoni
         12                 Emilian-Romagnol
         Name: Name in English, dtype: object

In [4]:   # izdvajanje veceg broja kolona istovremeno
          # stampa languages[['Name in English', 'Countries']]

In [5]:   # izdvajanje veceg broja kolona i vrsti istovremeno
          # stampa languages[['Name in English', 'Countries']][:10]

In [79]:  # grafik broja govornika za prvih 10 jezika
          import matplotlib
          \%matplotlib inline

          languages[['Name in English', 'Number of speakers']][:10].plot()

Out[79]: <matplotlib.axes._subplots.AxesSubplot at 0xc78b21f4e0>
```



In [84]:
```
# iscrtavamo grafik na osnovu izracunatih podataka
#languages['Degree of endangerment'].value_counts().plot()
languages['Degree of endangerment'].value_counts().plot(kind='pie')
```

Out[84]: <matplotlib.axes._subplots.AxesSubplot at 0xc78a9f7518>



In [85]:
```
# izdvajamo informacije samo o jezicima koji su kriticno ugrozeni
critically_endangered=languages[languages['Degree of endangerment'] == 'Critically endangere
```

In [6]:
```
# i ispisujemo informacije o prvih 5 najugrozenijih jezika
# stampa critically_endangered[:5]
```

In [16]:
```
# postoji i mogucnost vezivanja uslova: koriste se & |
ce = languages['Degree of endangerment'] == 'Critically endangered'
#print(ce)
vu = languages['Degree of endangerment'] == 'Vulnerable'
endangered_languages = languages[ce | vu]
#stampa endangered_languages[100:110]
```

In [91]:
```
# generisane podatke je moguce izvesti i sacuvati
endangered_languages.to_csv('./endangered.csv')
```

In [92]:
```
# stampa \!less endangered.csv
```

'less' is not recognized as an internal or external command,
operable program or batch file.

** Dodatni materijali: **

- https://github.com/jvns/pandas-cookbook
- knjiga : Python for Data Analysis, Wes McKinney

In [ ]: