

PostiLetto

```
[79]: # import libraries
import pandas as pd
import matplotlib.pyplot as plt

# read csv
csv = pd.read_csv('postiletto.csv', sep = ';')
csv_norm = pd.read_csv('popolazione_regioni_CSV_UTF8.csv', sep = ';')

# inspect data and their types
print(csv.head())
print(csv.dtypes)

# convert data to numbers where possible
csv = csv._convert(numeric = True)
csv_norm = csv_norm._convert(numeric = True)

# check data types again
print(csv.dtypes)
```

	Anno	Codice Regione	Descrizione Regione	Codice Azienda	Tipo Azienda	\
0	2010	10	PIEMONTE	201	1	
1	2010	10	PIEMONTE	201	1	
2	2010	10	PIEMONTE	201	1	
3	2010	10	PIEMONTE	201	1	
4	2010	10	PIEMONTE	201	1	

	Codice struttura	Denominazione struttura	\
0	10007	OSPEDALE MARTINI	
1	10010	OSPEDALE OFTALMICO	
2	10012	OSPEDALE EVANGELICO VALDESE	
3	10012	OSPEDALE EVANGELICO VALDESE	
4	10612	CASA DI CURA KOELLIKER OSPEDALINO	

	Indirizzo	Codice Comune	\
0	VIA TOFANE	1272	
1	VIA JUVARRA FILIPPO	1272	
2	VIA PELLICO SILVIO	1272	
3	VIA PELLICO SILVIO	1272	

4 CORSO FERRARIS GALILEO

1272

	Comune	Sigla	provincia	\
0	TORINO		TO	
1	TORINO		TO	
2	TORINO		TO	
3	TORINO		TO	
4	TORINO		TO	

	Codice tipo struttura	Descrizione tipo struttura	Tipo di Disciplina	\
0	1.0	Ospedale a gestione diretta	ACUTI	
1	1.0	Ospedale a gestione diretta	ACUTI	
2	1.0	Ospedale a gestione diretta	ACUTI	
3	1.0	Ospedale a gestione diretta	RIABILITAZIONE	
4	5.1	Casa di cura privata accreditata	ACUTI	

	Posti letto degenza ordinaria	Posti letto degenza a pagamento	\
0	202	0	
1	39	0	
2	51	0	
3	0	0	
4	84	0	

	Posti letto Day Hospital	Posti letto Day Surgery	Totale posti letto
0	21	1	224
1	4	20	63
2	20	36	107
3	1	0	1
4	0	21	105

Anno	int64
Codice Regione	int64
Descrizione Regione	object
Codice Azienda	int64
Tipo Azienda	int64
Codice struttura	int64
Denominazione struttura	object
Indirizzo	object
Codice Comune	int64
Comune	object
Sigla provincia	object
Codice tipo struttura	float64
Descrizione tipo struttura	object
Tipo di Disciplina	object
Posti letto degenza ordinaria	object
Posti letto degenza a pagamento	object
Posti letto Day Hospital	object
Posti letto Day Surgery	object
Totale posti letto	object

```

dtype: object
Anno                                int64
Codice Regione                     int64
Descrizione Regione                object
Codice Azienda                     int64
Tipo Azienda                       int64
Codice struttura                   int64
Denominazione struttura            object
Indirizzo                         object
Codice Comune                     int64
Comune                            object
Sigla provincia                   object
Codice tipo struttura              float64
Descrizione tipo struttura         object
Tipo di Disciplina                 object
Posti letto degenza ordinaria      float64
Posti letto degenza a pagamento    float64
Posti letto Day Hospital           float64
Posti letto Day Surgery            float64
Totale posti letto                 float64
dtype: object

```

```

[78]: anni = list(range(2010,2020))

for anno in anni:
    # select column with beds number and only for year 2014
    csvANNO = csv[ csv['Anno']==anno ]
    bedsANNO = csvANNO['Totale posti letto']

    csv_normANNO = csv_norm[ csv_norm['ANNO']==anno ]

    # descriptive stats
    #print(bedsANNO.describe())

    # plot distribution
    histogram = bedsANNO.hist(bins=50)

    # add details
    histogram.set_title('Distribution of beds per hospital - {}'.format(anno))
    histogram.set_xlabel('number of beds')
    histogram.set_ylabel('count')

    # show the plot
    plt.show()

    # sort dataframe by beds count

```

```

csvANNO = csvANNO.sort_values('Totale posti letto', ascending = False)
csv_normANNO = csv_normANNO.sort_values('POPOLAZIONE', ascending = False)

# show hospitals' name and beds count
#print(csvANNO[['Denominazione struttura', 'Totale posti letto']])

# group beds counts by region
bedsByRegion = csvANNO[['Descrizione Regione', 'Totale posti letto']].
↳groupby('Descrizione Regione')

# sum beds in each region and sort
summedAndSortedBedsByRegion = bedsByRegion.sum().sort_values('Totale posti_
↳letto')

# horizontal bar plot
summedAndSortedBedsByRegion.plot.barh()

# show the plot
plt.title("Totale posti letto - {}".format(anno))
plt.show()

TotalePostiLetto = []
for i in summedAndSortedBedsByRegion['Totale posti letto']:
    TotalePostiLetto.append(i)

PopxReg = []
for i in csv_normANNO['POPOLAZIONE']:
    PopxReg.append(i)

ValsNorm = []
for i in range(len(PopxReg)):
    ValNorm = TotalePostiLetto[i]/PopxReg[i]
    ValsNorm.append(ValNorm)

# normalization for population
summedAndSortedBedsByRegion['Totale posti letto'] = ValsNorm
summedAndSortedBedsByRegion.rename(columns={'Totale posti letto': 'Totale_
↳posti letto normalizzati rispetto al numero di abitanti'}, inplace=True)
summedAndSortedBedsByRegion.plot.barh()
plt.title("Totale posti letto normalizzati rispetto al numero di abitanti -_
↳{}".format(anno))
plt.show()

# Medical centers geografic distributions
MedCentersByRegion = csvANNO[['Descrizione Regione', 'Denominazione_
↳struttura']]

```

```

conteggio = MedCentersByRegion.groupby('Descrizione_
↪ Regione')['Denominazione struttura'].count()
conteggio.plot.barh()
plt.title("Distribuzione dei centri medici per regione - {}".format(anno))
plt.show()

```









































