Assignment 8

Q1. How-to-count-distance-to-the-previous-zero

For each value, count the difference back to the previous zero (or the start of the Series,

whichever is closer)

Answer:

import pandas as pd

import numpy as np

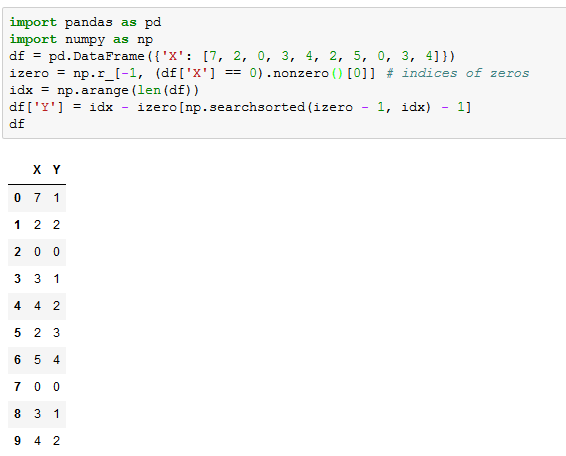
df=pd.DataFrame({'X':[7, 2, 0, 3, 4, 2, 5, 0, 3, 4]})

izero=np.r\_[-1,(df['X']==0).nonzero()[0]]

idx=np.arange(len(df))

df['Y']=idx-izero[np.searchsorted(izero-1,idx)-1]

df



Q2.

Create a DatetimeIndex that contains each business day of 2015 and use it to index a

Series of random numbers

Answer :

import pandas as pd

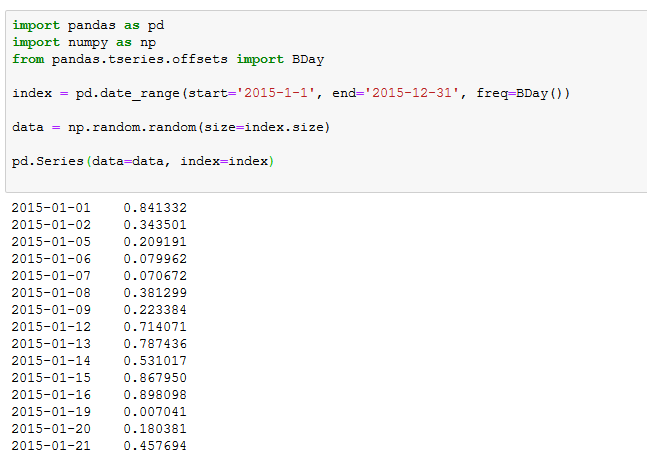
import numpy as np

from pandas.tseries.offsets import BDay

index = pd.date\_range(start='2015-1-1', end='2015-12-31', freq=BDay())

data = np.random.random(size=index.size)

pd.Series(data=data, index=index)



Q3 Find the sum of the values in s for every Wednesday.

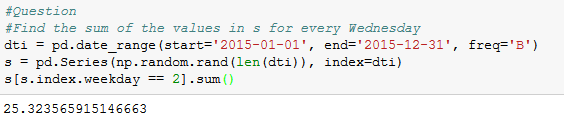
Answer:

import pandas as pd

dti = pd.date\_range(start='2015-01-01', end='2015-12-31', freq='B')

s = pd.Series(np.random.rand(len(dti)), index=dti)

s[s.index.weekday == 2].sum()



Q4 Average For each calendar month

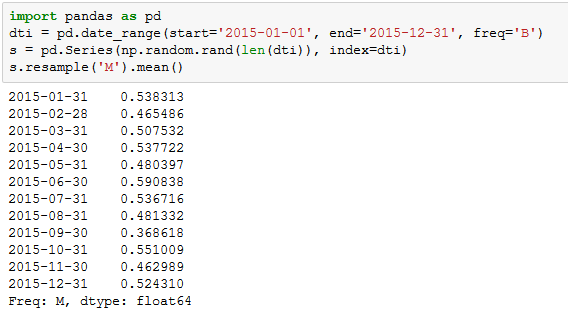
Answer :

import pandas as pd

dti = pd.date\_range(start='2015-01-01', end='2015-12-31', freq='B')

s = pd.Series(np.random.rand(len(dti)), index=dti)

s.resample('M').mean()



Q5 For each group of four consecutive calendar months in s, find the date on which the

highest value occurred.

Answer :

import pandas as pd

dti = pd.date\_range(start='2015-01-01', end='2015-12-31', freq='B')

s = pd.Series(np.random.rand(len(dti)), index=dti)

s.groupby(pd.TimeGrouper('4M')).idxmax()

