# lab5-yash-22000290

March 1, 2024

[2]:
```
import pandas as pd
```

[3]:
```
Subject=["DataScience","Php",".net"]
Marks =[80,90,100]
pd.Series(Marks,Subject)
```

[3]:
```
DataScience     80
Php             90
.net           100
dtype: int64
```

[4]:
```
a=pd.Series([33,34,35],index=["DSA","Php",".net"])
a
```

[4]:
```
DSA     33
Php     34
.net    35
dtype: int64
```

[5]:
```
dict={
    "name":"Matin",
    "cursoe":"BCA",
    "aderes":"Vadodara"
}
s=pd.Series(dict)
s
```

[5]:
```
name         Matin
cursoe         BCA
aderes    Vadodara
dtype: object
```

[6]:
```
s.keys()
```

[6]:
```
Index(['name', 'cursoe', 'aderes'], dtype='object')
```

[7]:
```
s.values
```

[7]:
```
array(['Matin', 'BCA', 'Vadodara'], dtype=object)
```

[8]:
```
import numpy as np
df=pd.DataFrame(np.random.
 ↪randint(1,100,(4,3)),index=["Yash","Matin","Kuldeep","jay"],columns=["DSA","Php",".
 ↪net"])
df
```

[8]:

| | DSA | Php | .net |
|---|---|---|---|
| Yash | 92 | 99 | 49 |
| Matin | 83 | 80 | 61 |
| Kuldeep | 48 | 62 | 91 |
| jay | 37 | 28 | 10 |

[9]:
```
df["DSA"]
```

[9]:
```
Yash       92
Matin      83
Kuldeep    48
jay        37
Name: DSA, dtype: int32
```

[10]:
```
df[["Php","DSA"]]
```

[10]:

| | Php | DSA |
|---|---|---|
| Yash | 99 | 92 |
| Matin | 80 | 83 |
| Kuldeep | 62 | 48 |
| jay | 28 | 37 |

[11]:
```
df['Total']=df["DSA"]+df["Php"]+df[".net"]
df
```

[11]:

| | DSA | Php | .net | Total |
|---|---|---|---|---|
| Yash | 92 | 99 | 49 | 240 |
| Matin | 83 | 80 | 61 | 224 |
| Kuldeep | 48 | 62 | 91 | 201 |
| jay | 37 | 28 | 10 | 75 |

[12]:
```
df["perc"]=df["Total"]/3
df
```

[12]:

| | DSA | Php | .net | Total | perc |
|---|---|---|---|---|---|
| Yash | 92 | 99 | 49 | 240 | 80.000000 |
| Matin | 83 | 80 | 61 | 224 | 74.666667 |
| Kuldeep | 48 | 62 | 91 | 201 | 67.000000 |
| jay | 37 | 28 | 10 | 75 | 25.000000 |

[13]:
```
df.loc[["Yash","Matin"]]
```

[13]:

| | DSA | Php | .net | Total | perc |
|---|---|---|---|---|---|
| Yash | 92 | 99 | 49 | 240 | 80.000000 |
| Matin | 83 | 80 | 61 | 224 | 74.666667 |

[14]:
```
df.head(2)
```

[14]:

| | DSA | Php | .net | Total | perc |
|---|---|---|---|---|---|
| Yash | 92 | 99 | 49 | 240 | 80.000000 |
| Matin | 83 | 80 | 61 | 224 | 74.666667 |

[15]:
```
df.tail(2)
```

[15]:

| | DSA | Php | .net | Total | perc |
|---|---|---|---|---|---|
| Kuldeep | 48 | 62 | 91 | 201 | 67.0 |
| jay | 37 | 28 | 10 | 75 | 25.0 |

[16]:
```
df.loc["Matin":]
```

[16]:

| | DSA | Php | .net | Total | perc |
|---|---|---|---|---|---|
| Matin | 83 | 80 | 61 | 224 | 74.666667 |
| Kuldeep | 48 | 62 | 91 | 201 | 67.000000 |
| jay | 37 | 28 | 10 | 75 | 25.000000 |

[17]:
```
df.loc["Yash":"Kuldeep"]
```

[17]:

| | DSA | Php | .net | Total | perc |
|---|---|---|---|---|---|
| Yash | 92 | 99 | 49 | 240 | 80.000000 |
| Matin | 83 | 80 | 61 | 224 | 74.666667 |
| Kuldeep | 48 | 62 | 91 | 201 | 67.000000 |

[18]:
```
df.iloc[2]
```

[18]:
```
DSA       48.0
Php       62.0
.net      91.0
Total    201.0
perc      67.0
Name: Kuldeep, dtype: float64
```

[19]:
```
df.drop("perc",axis=1)
```

[19]:

| | DSA | Php | .net | Total |
|---|---|---|---|---|
| Yash | 92 | 99 | 49 | 240 |
| Matin | 83 | 80 | 61 | 224 |
| Kuldeep | 48 | 62 | 91 | 201 |

```
jay       37   28    10     75
```

[20]:
```
df.drop("perc",axis=1,inplace=True)
```

[21]:
```
df
```

[21]:
```
          DSA  Php  .net  Total
Yash       92   99    49    240
Matin      83   80    61    224
Kuldeep    48   62    91    201
jay        37   28    10     75
```

[22]:
```
df.drop("jay",axis=0)
```

[22]:
```
          DSA  Php  .net  Total
Yash       92   99    49    240
Matin      83   80    61    224
Kuldeep    48   62    91    201
```

[23]:
```
#lab 5
```

[24]:
```
ProductId=[10,20,30,40]
ProductName=["Sampho","Soap","Brush","HandWash"]
ProductPrice=[50,60,40,56]
```

[25]:
```
a=pd.
  ↪DataFrame(data=[ProductId,ProductName,ProductPrice],index=["ProductId","ProductName","Produ
  ↪transpose()
```

[26]:
```
a
```

[26]:
```
   ProductId ProductName ProductPrice
0         10      Sampho           50
1         20        Soap           60
2         30       Brush           40
3         40    HandWash           56
```

[27]:
```
a["ProductId"]
```

[27]:
```
0    10
1    20
2    30
3    40
Name: ProductId, dtype: object
```

[28]:
```
a["ProductName"]
```

[28]:
```
0      Sampho
1        Soap
2       Brush
3    HandWash
Name: ProductName, dtype: object
```

[29]:
```
a["ProductPrice"]
```

[29]:
```
0    50
1    60
2    40
3    56
Name: ProductPrice, dtype: object
```

[30]:
```
a.loc[0]
```

[30]:
```
ProductId           10
ProductName     Sampho
ProductPrice        50
Name: 0, dtype: object
```

[31]:
```
a.loc[1]
```

[31]:
```
ProductId         20
ProductName     Soap
ProductPrice      60
Name: 1, dtype: object
```

[32]:
```
a.loc[2]
```

[32]:
```
ProductId          30
ProductName     Brush
ProductPrice       40
Name: 2, dtype: object
```

[33]:
```
a.loc[3]
```

[33]:
```
ProductId             40
ProductName     HandWash
ProductPrice          56
Name: 3, dtype: object
```

[34]:
```
a.loc[1:4,["ProductName"]]
```

[34]:
```
  ProductName
1        Soap
2       Brush
```

```
3    HandWash
```

[35]:
```
a.iloc[1,[2]]
```

[35]:
```
ProductPrice    60
Name: 1, dtype: object
```

[36]:
```
a.iloc[2:,0:2]
```

[36]:
```
  ProductId ProductName
2        30       Brush
3        40    HandWash
```

[37]:
```
df
```

[37]:
```
          DSA  Php  .net  Total
Yash       92   99    49    240
Matin      83   80    61    224
Kuldeep    48   62    91    201
jay        37   28    10     75
```

[38]:
```
df["DSA"]<40
```

[38]:
```
Yash       False
Matin      False
Kuldeep    False
jay         True
Name: DSA, dtype: bool
```

[39]:
```
df.loc[df["DSA"]<40,'DSA']=41
df
```

[39]:
```
          DSA  Php  .net  Total
Yash       92   99    49    240
Matin      83   80    61    224
Kuldeep    48   62    91    201
jay        41   28    10     75
```

[40]:
```
df.add(2)
```

[40]:
```
          DSA  Php  .net  Total
Yash       94  101    51    242
Matin      85   82    63    226
Kuldeep    50   64    93    203
jay        43   30    12     77
```

[41]:
```
df[(df["DSA"]>50) &  (df[".net"]>60)]
```

[41]:
```
        DSA  Php  .net  Total
Matin    83   80    61    224
```

[42]:
```
df[df.index.str.startswith("M")]
```

[42]:
```
        DSA  Php  .net  Total
Matin    83   80    61    224
```

[43]:
```
df[df.index.str.endswith("h")]
```

[43]:
```
       DSA  Php  .net  Total
Yash    92   99    49    240
```

[44]:
```
df["perc"]=df["Total"]/3
df
```

[44]:
```
          DSA  Php  .net  Total       perc
Yash       92   99    49    240  80.000000
Matin      83   80    61    224  74.666667
Kuldeep    48   62    91    201  67.000000
jay        41   28    10     75  25.000000
```

[45]:
```
df["status"]=np.where(df["perc"]>=50,"Pass","fail")
df
```

[45]:
```
          DSA  Php  .net  Total       perc status
Yash       92   99    49    240  80.000000   Pass
Matin      83   80    61    224  74.666667   Pass
Kuldeep    48   62    91    201  67.000000   Pass
jay        41   28    10     75  25.000000   fail
```

[50]:
```
df['Grade']=np.where(df['perc']>=85,'A',np.where(df['perc']>=60,'B',np.
↪where(df['perc']>50,'c','d')))
df
```

[50]:
```
          DSA  Php  .net  Total       perc status Grade
Yash       92   99    49    240  80.000000   Pass     B
Matin      83   80    61    224  74.666667   Pass     B
Kuldeep    48   62    91    201  67.000000   Pass     B
jay        41   28    10     75  25.000000   fail     d
```

[52]:
```
df.describe()
```

[52]:
```
              DSA        Php       .net       Total       perc
count   4.000000   4.000000   4.000000    4.000000   4.000000
mean   66.000000  67.250000  52.750000  185.000000  61.666667
std    25.258662  30.214511  33.529837   75.059976  25.019992
```

```
min    41.000000  28.000000  10.000000   75.000000  25.000000
25%    46.250000  53.500000  39.250000  169.500000  56.500000
50%    65.500000  71.000000  55.000000  212.500000  70.833333
75%    85.250000  84.750000  68.500000  228.000000  76.000000
max    92.000000  99.000000  91.000000  240.000000  80.000000
```

[56]:
```
result = df.to_json(orient="table")
result
```

[56]:
```
'{"schema":{"fields":[{"name":"index","type":"string"},{"name":"DSA","type":"int
eger"},{"name":"Php","type":"integer"},{"name":".net","type":"integer"},{"name":
"Total","type":"integer"},{"name":"perc","type":"number"},{"name":"status","type
":"string"},{"name":"Grade","type":"string"}],"primaryKey":["index"],"pandas_ver
sion":"1.4.0"},"data":[{"index":"Yash","DSA":92,"Php":99,".net":49,"Total":240,"
perc":80.0,"status":"Pass","Grade":"B"},{"index":"Matin","DSA":83,"Php":80,".net
":61,"Total":224,"perc":74.6666666667,"status":"Pass","Grade":"B"},{"index":"Kul
deep","DSA":48,"Php":62,".net":91,"Total":201,"perc":67.0,"status":"Pass","Grade
":"B"},{"index":"jay","DSA":41,"Php":28,".net":10,"Total":75,"perc":25.0,"status
":"fail","Grade":"d"}]}'
```

[ ]:
```

```

[60]:
```
df.to_excel("output.xlsx")
```

[62]:
```
df.to_html()
```

[62]:
```
'<table border="1" class="dataframe">\n  <thead>\n    <tr style="text-align:
right;">\n      <th></th>\n      <th>DSA</th>\n      <th>Php</th>\n
<th>.net</th>\n      <th>Total</th>\n      <th>perc</th>\n
<th>status</th>\n      <th>Grade</th>\n    </tr>\n  </thead>\n  <tbody>\n
<tr>\n      <th>Yash</th>\n      <td>92</td>\n      <td>99</td>\n
<td>49</td>\n      <td>240</td>\n      <td>80.000000</td>\n      <td>Pass</td>\n
<td>B</td>\n    </tr>\n    <tr>\n      <th>Matin</th>\n      <td>83</td>\n
<td>80</td>\n      <td>61</td>\n      <td>224</td>\n      <td>74.666667</td>\n
<td>Pass</td>\n      <td>B</td>\n    </tr>\n    <tr>\n      <th>Kuldeep</th>\n
<td>48</td>\n      <td>62</td>\n      <td>91</td>\n      <td>201</td>\n
<td>67.000000</td>\n      <td>Pass</td>\n      <td>B</td>\n    </tr>\n    <tr>\n
<th>jay</th>\n      <td>41</td>\n      <td>28</td>\n      <td>10</td>\n
<td>75</td>\n      <td>25.000000</td>\n      <td>fail</td>\n      <td>d</td>\n
</tr>\n  </tbody>\n</table>'
```

[63]:
```
df.dtypes
```

[63]:
```
DSA         int32
Php         int32
.net        int32
Total       int32
```

```
perc      float64
status     object
Grade      object
dtype: object
```

[65]:
```
df.to_dict(orient='list')
```

[65]:
```
{'DSA': [92, 83, 48, 41],
 'Php': [99, 80, 62, 28],
 '.net': [49, 61, 91, 10],
 'Total': [240, 224, 201, 75],
 'perc': [80.0, 74.66666666666667, 67.0, 25.0],
 'status': ['Pass', 'Pass', 'Pass', 'fail'],
 'Grade': ['B', 'B', 'B', 'd']}
```

[75]:
```
a['ProductPrice']=a['ProductPrice']+500
newDataFrame=pd.DataFrame(a)
newDataFrame
```

[75]:

| | ProductId | ProductName | ProductPrice |
|---|---|---|---|
| 0 | 10 | Sampho | 2050 |
| 1 | 20 | Soap | 2060 |
| 2 | 30 | Brush | 2040 |
| 3 | 40 | HandWash | 2056 |

[ ]:
```
#task 2 fro here
```

[76]:
```
sal = pd.read_csv('Salaries.csv')
sal
```

```
C:\Users\Lenovo\AppData\Local\Temp\ipykernel_17636\1798377461.py:1:
DtypeWarning: Columns (3,4,5,6,12) have mixed types. Specify dtype option on
import or set low_memory=False.
  sal = pd.read_csv('Salaries.csv')
```

[76]:

| | Id | EmployeeName | \ |
|---|---|---|---|
| 0 | 1 | NATHANIEL FORD | |
| 1 | 2 | GARY JIMENEZ | |
| 2 | 3 | ALBERT PARDINI | |
| 3 | 4 | CHRISTOPHER CHONG | |
| 4 | 5 | PATRICK GARDNER | |
| … | … | … | |
| 148649 | 148650 | Roy I Tillery | |
| 148650 | 148651 | Not provided | |
| 148651 | 148652 | Not provided | |
| 148652 | 148653 | Not provided | |
| 148653 | 148654 | Joe Lopez | |

```
                                                 JobTitle       BasePay  \
0       GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY     167411.18
1                     CAPTAIN III (POLICE DEPARTMENT)     155966.02
2                     CAPTAIN III (POLICE DEPARTMENT)     212739.13
3                 WIRE ROPE CABLE MAINTENANCE MECHANIC       77916.0
4         DEPUTY CHIEF OF DEPARTMENT,(FIRE DEPARTMENT)      134401.6
…                                                   …           …
148649                                      Custodian          0.00
148650                                  Not provided  Not Provided
148651                                  Not provided  Not Provided
148652                                  Not provided  Not Provided
148653                       Counselor, Log Cabin Ranch          0.00

         OvertimePay      OtherPay      Benefits   TotalPay  TotalPayBenefits  \
0                0.0     400184.25           NaN  567595.43         567595.43
1          245131.88     137811.38           NaN  538909.28         538909.28
2          106088.18       16452.6           NaN  335279.91         335279.91
3           56120.71      198306.9           NaN  332343.61         332343.61
4             9737.0     182234.59           NaN  326373.19         326373.19
…                …             …             …          …                 …
148649          0.00          0.00          0.00       0.00              0.00
148650  Not Provided  Not Provided  Not Provided       0.00              0.00
148651  Not Provided  Not Provided  Not Provided       0.00              0.00
148652  Not Provided  Not Provided  Not Provided       0.00              0.00
148653          0.00       -618.13          0.00    -618.13           -618.13

         Year  Notes         Agency Status
0       2011    NaN  San Francisco    NaN
1       2011    NaN  San Francisco    NaN
2       2011    NaN  San Francisco    NaN
3       2011    NaN  San Francisco    NaN
4       2011    NaN  San Francisco    NaN
…       …    …            …    …
148649  2014    NaN  San Francisco     PT
148650  2014    NaN  San Francisco    NaN
148651  2014    NaN  San Francisco    NaN
148652  2014    NaN  San Francisco    NaN
148653  2014    NaN  San Francisco     PT

[148654 rows x 13 columns]
```

[77]:
```
sal.head()
```

[77]:
```
    Id       EmployeeName                                       JobTitle  \
0   1     NATHANIEL FORD  GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY
1   2       GARY JIMENEZ                 CAPTAIN III (POLICE DEPARTMENT)
```

```
2   3     ALBERT PARDINI                CAPTAIN III (POLICE DEPARTMENT)
3   4  CHRISTOPHER CHONG           WIRE ROPE CABLE MAINTENANCE MECHANIC
4   5    PATRICK GARDNER    DEPUTY CHIEF OF DEPARTMENT,(FIRE DEPARTMENT)

     BasePay OvertimePay   OtherPay Benefits   TotalPay  TotalPayBenefits  \
0  167411.18         0.0  400184.25      NaN  567595.43         567595.43
1  155966.02   245131.88  137811.38      NaN  538909.28         538909.28
2  212739.13   106088.18    16452.6      NaN  335279.91         335279.91
3    77916.0    56120.71   198306.9      NaN  332343.61         332343.61
4   134401.6      9737.0  182234.59      NaN  326373.19         326373.19

   Year  Notes         Agency Status
0  2011    NaN  San Francisco    NaN
1  2011    NaN  San Francisco    NaN
2  2011    NaN  San Francisco    NaN
3  2011    NaN  San Francisco    NaN
4  2011    NaN  San Francisco    NaN
```

[78]:
```
sal.info
```

[78]:
```
<bound method DataFrame.info of            Id       EmployeeName  \
0            1     NATHANIEL FORD
1            2       GARY JIMENEZ
2            3     ALBERT PARDINI
3            4  CHRISTOPHER CHONG
4            5    PATRICK GARDNER
…          …                …
148649  148650      Roy I Tillery
148650  148651       Not provided
148651  148652       Not provided
148652  148653       Not provided
148653  148654          Joe Lopez

                                          JobTitle       BasePay  \
0       GENERAL MANAGER-METROPOLITAN TRANSIT AUTHORITY     167411.18
1                    CAPTAIN III (POLICE DEPARTMENT)     155966.02
2                    CAPTAIN III (POLICE DEPARTMENT)     212739.13
3                WIRE ROPE CABLE MAINTENANCE MECHANIC       77916.0
4         DEPUTY CHIEF OF DEPARTMENT,(FIRE DEPARTMENT)      134401.6
…                                                   …           …
148649                                      Custodian          0.00
148650                                   Not provided  Not Provided
148651                                   Not provided  Not Provided
148652                                   Not provided  Not Provided
148653                     Counselor, Log Cabin Ranch          0.00

         OvertimePay      OtherPay      Benefits   TotalPay  TotalPayBenefits  \
```

```
0                0.0     400184.25           NaN  567595.43         567595.43
1           245131.88     137811.38           NaN  538909.28         538909.28
2           106088.18       16452.6           NaN  335279.91         335279.91
3            56120.71      198306.9           NaN  332343.61         332343.61
4              9737.0     182234.59           NaN  326373.19         326373.19
…                …           …           …        …                 …
148649          0.00          0.00          0.00       0.00              0.00
148650  Not Provided  Not Provided  Not Provided       0.00              0.00
148651  Not Provided  Not Provided  Not Provided       0.00              0.00
148652  Not Provided  Not Provided  Not Provided       0.00              0.00
148653          0.00       -618.13          0.00    -618.13           -618.13

        Year  Notes         Agency Status
0       2011    NaN  San Francisco    NaN
1       2011    NaN  San Francisco    NaN
2       2011    NaN  San Francisco    NaN
3       2011    NaN  San Francisco    NaN
4       2011    NaN  San Francisco    NaN
…       …    …            …    …
148649  2014    NaN  San Francisco     PT
148650  2014    NaN  San Francisco    NaN
148651  2014    NaN  San Francisco    NaN
148652  2014    NaN  San Francisco    NaN
148653  2014    NaN  San Francisco     PT

[148654 rows x 13 columns]>
```

[87]:
```
sal=sal.replace('Not provided',np.nan)
sal=sal.replace('Not Provided',np.nan)
sal['BasePay'].astype('float')
sal['BasePay']=sal['BasePay'].astype('float')
sal['BasePay'].mean()
```

[87]:
```
66325.44884050643
```

[80]:
```
sal['JobTitle'].nunique()
```

[80]:
```
2159
```

[81]:
```
sal['JobTitle'].value_counts().head(5)
```

[81]:
```
Transit Operator                7036
Special Nurse                   4389
Registered Nurse                3736
Public Svc Aide-Public Works    2518
Police Officer 3                2421
Name: JobTitle, dtype: int64
```

[82]:
```
sum(sal[sal['Year']==2013]['JobTitle'].value_counts() == 1) # pretty tricky way␣
↪to do this...
```

[82]:
```
202
```

[84]:
```
def chief_string(title):
    if 'chief' in title.lower():
        return True
    else:
        return False
sum(sal['JobTitle'].apply(lambda x: chief_string(x)))
```

[84]:
```
627
```

[85]:
```
sal['title_len'] = sal['JobTitle'].apply(len)
```

[86]:
```
sal[['title_len','TotalPayBenefits']].corr() # No correlation.
```

[86]:

| | title_len | TotalPayBenefits |
|---|---|---|
| title_len | 1.000000 | -0.036878 |
| TotalPayBenefits | -0.036878 | 1.000000 |

[ ]:
```

```