DSBDA Practical No.01

May 19, 2023

[65]:
```
#1. Import all the required Python Librarie
```

[3]:
```
import pandas as pd
```

[4]:
```
import numpy as np
```

[5]:
```
import matplotlib.pyplot as plt
```

[68]:
```
%matplotlib inline
#so that we can view the graphs inside the notebook
```

[7]:
```
s1 = pd.Series(range(1,10,1))
```

[8]:
```
s1
```

[8]:
```
0    1
1    2
2    3
3    4
4    5
5    6
6    7
7    8
8    9
dtype: int64
```

[9]:
```
s3 = pd.Series({1:21, 2:13,3:45})
```

[10]:
```
s3
```

[10]:
```
1    21
2    13
3    45
dtype: int64
```

[11]:
```
s2 = pd.Series([1, 2, 3, 4], index=['p', 'q', 'r','s'], name='one')
```

[12]:
```
s2
```

[12]:
```
p    1
q    2
r    3
s    4
Name: one, dtype: int64
```

[13]:
```
df1 = pd.DataFrame(s2)
```

[14]:
```
df1
```

[14]:
```
   one
p    1
q    2
r    3
s    4
```

[70]:
```
#Load the Dataset into pandas data frame
```

[15]:
```
df2 = pd.read_csv("/Users/janhvikarki/Desktop/Dataset/employees.csv")
```

[16]:
```
df2.head(10)
```

[16]:
```
  First Name  Gender  Start Date Last Login Time    Salary  Bonus %  \
0    Douglas    Male    8/6/1993        12:42 PM   97308.0    6.945
1     Thomas    Male   3/31/1996         6:53 AM   61933.0    4.170
2      Maria  Female   4/23/1993        11:17 AM  130590.0   11.858
3      Jerry    Male    3/4/2005         1:00 PM  138705.0    9.340
4      Larry    Male   1/24/1998         4:47 PM  101004.0    1.389
5     Dennis    Male   4/18/1987         1:35 AM  115163.0   10.125
6       Ruby  Female   8/17/1987         4:20 PM   65476.0   10.012
7        NaN  Female   7/20/2015        10:43 AM   45906.0   11.598
8     Angela  Female  11/22/2005         6:29 AM   95570.0   18.523
9    Frances  Female    8/8/2002         6:51 AM  139852.0    7.524

  Senior Management                  Team
0              True             Marketing
1              True                   NaN
2             False               Finance
3              True               Finance
4              True       Client Services
5             False                 Legal
6              True               Product
7               NaN               Finance
8              True           Engineering
9              True  Business Development
```

[17]:

```
df2.tail(3)
```

[17]:

```
    First Name Gender Start Date Last Login Time    Salary  Bonus %  \
997    Russell   Male  5/20/2013        12:39 PM   96914.0    1.421
998      Larry   Male  4/20/2013         4:45 PM   60500.0   11.985
999     Albert   Male  5/15/2012         6:24 PM  129949.0   10.169

    Senior Management                  Team
997             False               Product
998             False  Business Development
999              True                 Sales
```

[18]:

```
df2.to_json('data1.json')
```

[21]:

```
len(df2['Team'])
```

[21]:

```
1000
```

[22]:

```
df2['Team'].count()
```

[22]:

```
957
```

[24]:

```
df2['Salary'].mean()
```

[24]:

```
90579.97213622292
```

[25]:

```
df2['Salary'].sum()
```

[25]:

```
87771993.0
```

[26]:

```
df2['Salary'].median()
```

[26]:

```
90370.0
```

[27]:

```
df2['Salary'].std()
```

[27]:

```
32916.214577497005
```

[28]:

```
df2['Salary'].min()
```

[28]:

```
35013.0
```

[29]:

```
df2['Salary'].describe()
```

[29]:

```
count      969.000000
mean     90579.972136
std      32916.214577
```

```
min       35013.000000
25%       62666.000000
50%       90370.000000
75%      118733.000000
max      149908.000000
Name: Salary, dtype: float64
```

[30]:

```
df2['Salary'].cumsum()
```

[30]:

```
0         97308.0
1        159241.0
2        289831.0
3        428536.0
4        529540.0
          …
995    87442238.0
996    87484630.0
997    87581544.0
998    87642044.0
999    87771993.0
Name: Salary, Length: 1000, dtype: float64
```

[64]:

```
# When you give the whole dataframe, then all numerical columns will be analysis
df2.mean()
```

```
/var/folders/cs/hplqvnxd09bg_bgmf6zh8t3m0000gn/T/ipykernel_9509/3587575296.py:1:
FutureWarning: The default value of numeric_only in DataFrame.mean is
deprecated. In a future version, it will default to False. In addition,
specifying 'numeric_only=None' is deprecated. Select only valid columns or
specify the value of numeric_only to silence this warning.
  df2.mean()
```

[64]:

```
Salary               90579.942000
Bonus %                 10.207555
Senior Management        0.501608
dtype: float64
```

[32]:

```
df2.describe()
```

[32]:

```
              Salary      Bonus %
count     969.000000  1000.000000
mean    90579.972136    10.207555
std     32916.214577     5.528481
min     35013.000000     1.015000
25%     62666.000000     5.401750
50%     90370.000000     9.838500
75%    118733.000000    14.838000
```

```
max    149908.000000    19.944000
```

[33]:

```
# DATA PREPROCESSING
```

[41]:

```
#importing pandas as pd
import pandas as pd

#making data frame from csv file
df2 = pd.read_csv("/Users/shreyaspeherkar/Desktop/Dataset/employees.csv")

df2.head(10)
```

[41]:

```
  First Name  Gender  Start Date Last Login Time    Salary  Bonus %  \
0    Douglas    Male    8/6/1993        12:42 PM   97308.0    6.945
1     Thomas    Male   3/31/1996         6:53 AM   61933.0    4.170
2      Maria  Female   4/23/1993        11:17 AM  130590.0   11.858
3      Jerry    Male    3/4/2005         1:00 PM  138705.0    9.340
4      Larry    Male   1/24/1998         4:47 PM  101004.0    1.389
5     Dennis    Male   4/18/1987         1:35 AM  115163.0   10.125
6       Ruby  Female   8/17/1987         4:20 PM   65476.0   10.012
7        NaN  Female   7/20/2015        10:43 AM   45906.0   11.598
8     Angela  Female  11/22/2005         6:29 AM   95570.0   18.523
9    Frances  Female    8/8/2002         6:51 AM  139852.0    7.524

  Senior Management                  Team
0              True             Marketing
1              True                   NaN
2             False               Finance
3              True               Finance
4              True       Client Services
5             False                 Legal
6              True               Product
7               NaN               Finance
8              True           Engineering
9              True  Business Development
```

[42]:

```
df2.describe()
```

[42]:

```
              Salary      Bonus %
count     969.000000  1000.000000
mean    90579.972136    10.207555
std     32916.214577     5.528481
min     35013.000000     1.015000
25%     62666.000000     5.401750
50%     90370.000000     9.838500
75%    118733.000000    14.838000
max    149908.000000    19.944000
```

[43]:

```
df2.isnull()
```

[43]:

```
     First Name  Gender  Start Date  Last Login Time  Salary  Bonus %  \
0         False   False       False            False   False    False
1         False   False       False            False   False    False
2         False   False       False            False   False    False
3         False   False       False            False   False    False
4         False   False       False            False   False    False
..          …     …         …              …     …      …
995       False    True       False            False   False    False
996       False   False       False            False   False    False
997       False   False       False            False   False    False
998       False   False       False            False   False    False
999       False   False       False            False   False    False

     Senior Management  Team
0                False  False
1                False   True
2                False  False
3                False  False
4                False  False
..                 …    …
995              False  False
996              False  False
997              False  False
998              False  False
999              False  False

[1000 rows x 8 columns]
```

[44]:

```
df2.notnull()
```

[44]:

```
     First Name  Gender  Start Date  Last Login Time  Salary  Bonus %  \
0          True    True        True             True    True     True
1          True    True        True             True    True     True
2          True    True        True             True    True     True
3          True    True        True             True    True     True
4          True    True        True             True    True     True
..          …     …         …              …     …      …
995        True   False        True             True    True     True
996        True    True        True             True    True     True
997        True    True        True             True    True     True
998        True    True        True             True    True     True
999        True    True        True             True    True     True

     Senior Management  Team
0                 True   True
```

```
1                  True  False
2                  True   True
3                  True   True
4                  True   True
..                  …     …
995                True   True
996                True   True
997                True   True
998                True   True
999                True   True

[1000 rows x 8 columns]
```

[45]:
```
df2.isnull().sum()
```

[45]:
```
First Name             67
Gender                145
Start Date              0
Last Login Time         0
Salary                 31
Bonus %                 0
Senior Management      67
Team                   43
dtype: int64
```

[47]:
```
#Filling a null values using fillna()
```

[48]:
```
df2["Gender"].fillna("No Gender", inplace = True)
```

[49]:
```
df2.isnull().sum()
```

[49]:
```
First Name            67
Gender                 0
Start Date             0
Last Login Time        0
Salary                31
Bonus %                0
Senior Management     67
Team                  43
dtype: int64
```

[50]:
```
# will replace  Nan value in dataframe with value -99
```

[51]:
```
import numpy as np
df2.replace(to_replace = np.nan, value = -99)
```

[51]:

```
    First Name     Gender  Start Date Last Login Time    Salary  Bonus %  \
0      Douglas       Male    8/6/1993        12:42 PM   97308.0    6.945
1       Thomas       Male   3/31/1996         6:53 AM   61933.0    4.170
2        Maria     Female   4/23/1993        11:17 AM  130590.0   11.858
3        Jerry       Male    3/4/2005         1:00 PM  138705.0    9.340
4        Larry       Male   1/24/1998         4:47 PM  101004.0    1.389
..          …        …         …             …       …      …
995      Henry  No Gender  11/23/2014         6:09 AM  132483.0   16.655
996    Phillip       Male   1/31/1984         6:30 AM   42392.0   19.675
997    Russell       Male   5/20/2013        12:39 PM   96914.0    1.421
998      Larry       Male   4/20/2013         4:45 PM   60500.0   11.985
999     Albert       Male   5/15/2012         6:24 PM  129949.0   10.169

    Senior Management                 Team
0                True            Marketing
1                True                  -99
2               False              Finance
3                True              Finance
4                True      Client Services
..                 …                   …
995             False         Distribution
996             False              Finance
997             False              Product
998             False  Business Development
999              True                Sales

[1000 rows x 8 columns]
```

[52]:

```
# filling a missing value with previous ones
df2.fillna(method ='pad')
```

[52]:

```
    First Name     Gender  Start Date Last Login Time    Salary  Bonus %  \
0      Douglas       Male    8/6/1993        12:42 PM   97308.0    6.945
1       Thomas       Male   3/31/1996         6:53 AM   61933.0    4.170
2        Maria     Female   4/23/1993        11:17 AM  130590.0   11.858
3        Jerry       Male    3/4/2005         1:00 PM  138705.0    9.340
4        Larry       Male   1/24/1998         4:47 PM  101004.0    1.389
..          …        …         …             …       …      …
995      Henry  No Gender  11/23/2014         6:09 AM  132483.0   16.655
996    Phillip       Male   1/31/1984         6:30 AM   42392.0   19.675
997    Russell       Male   5/20/2013        12:39 PM   96914.0    1.421
998      Larry       Male   4/20/2013         4:45 PM   60500.0   11.985
999     Albert       Male   5/15/2012         6:24 PM  129949.0   10.169

     Senior Management                 Team
0                 True            Marketing
1                 True            Marketing
```

```
2                False               Finance
3                 True               Finance
4                 True       Client Services
..                  …                   …
995              False          Distribution
996              False               Finance
997              False               Product
998              False  Business Development
999               True                 Sales

[1000 rows x 8 columns]
```

[53]:

```
df2['Salary'].fillna(int(df2['Salary'].mean()), inplace=True)
```

[54]:

```
#Dropping missing values using dropna()
```

[55]:

```
df2.dropna(axis=1)
```

[55]:

```
        Gender  Start Date Last Login Time    Salary  Bonus %
0         Male    8/6/1993        12:42 PM   97308.0    6.945
1         Male   3/31/1996         6:53 AM   61933.0    4.170
2       Female   4/23/1993        11:17 AM  130590.0   11.858
3         Male    3/4/2005         1:00 PM  138705.0    9.340
4         Male   1/24/1998         4:47 PM  101004.0    1.389
..         …          …               …        …       …
995  No Gender  11/23/2014         6:09 AM  132483.0   16.655
996       Male   1/31/1984         6:30 AM   42392.0   19.675
997       Male   5/20/2013        12:39 PM   96914.0    1.421
998       Male   4/20/2013         4:45 PM   60500.0   11.985
999       Male   5/15/2012         6:24 PM  129949.0   10.169

[1000 rows x 5 columns]
```

[56]:

```
# importing pandas as pd
import pandas as pd
# Creating the dataframe
df = pd.DataFrame({"A":[12, 4, 5, None, 1],
                   "B":[None, 2, 54, 3, None],
                   "C":[20, 16, None, 3, 8],
                   "D":[14, 3, None, None, 6]})
# Print the dataframe
df
```

[56]:

```
      A     B     C     D
0  12.0   NaN  20.0  14.0
1   4.0   2.0  16.0   3.0
2   5.0  54.0   NaN   NaN
```

```
3   NaN   3.0   3.0   NaN
4   1.0   NaN   8.0   6.0
```

```
[58]: df.interpolate(method = 'linear', limit_direction ='forward')
```

```
[58]:       A     B     C     D
      0  12.0   NaN  20.0  14.0
      1   4.0   2.0  16.0   3.0
      2   5.0  54.0   9.5   4.0
      3   3.0   3.0   3.0   5.0
      4   1.0   3.0   8.0   6.0
```

```
[59]: #Data Formatting and Data Normalization
```

```
[60]: #remove white space everywhere
      text="today is Monday"
      #df['Col Name'] = df['Col Name'].str.replace(‘ ‘, ‘’)
      text.replace(' ','')
```

```
[60]: 'todayisMonday'
```

```
[61]: text=' Today'
      text.lstrip()
```

```
[61]: 'Today'
```

```
[62]: text='Today '
      text.rstrip()
```

```
[62]: 'Today'
```

```
[63]: text=' Today '
      text.strip()
```

```
[63]: 'Today'
```