PANDAS PART - 2 BY VIRAT TIWARI

October 11, 2023

1 PANDAS PART 2 - IT ALLOWS US TO ANALYZE THE DATA AND MANIPULATION OF THE DATA . IT ALSO HELP IN CLEANING THE MESSY DATA OR PRESENT THEM IN A STRUCTED WAY - VIRAT TIWARI

[1]:
```
import pandas as pd
```

[2]:
```
# df.read_csv ( ) function is used for reading the csv datasets

df=pd.read_csv("https://raw.githubusercontent.com/datasciencedojo/datasets/
↪master/titanic.csv")
```

[3]:
```
df.head()
```

[3]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th…	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

[4]:
```
# type ( ) function gives the type of df or dataframe

type(df)
```

[4]:
```
pandas.core.frame.DataFrame
```

[5]:
```
# df.dtypes ( ) function gives the data type of all types of data that present
  ↪in dataset

df.dtypes
```

[5]:
```
PassengerId      int64
Survived         int64
Pclass           int64
Name            object
Sex             object
Age            float64
SibSp            int64
Parch            int64
Ticket          object
Fare           float64
Cabin           object
Embarked        object
dtype: object
```

[7]:
```
# describe ( ) function gives the description of numerical data

df.describe()
```

[7]:

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000		

```
max      6.000000  512.329200
```

[8]:
```
# df.dtypes ( ) function gives the data type of all types of data that present␣
 ↪in dataset

df.dtypes
```

[8]:
```
PassengerId      int64
Survived         int64
Pclass           int64
Name            object
Sex             object
Age            float64
SibSp            int64
Parch            int64
Ticket          object
Fare           float64
Cabin           object
Embarked        object
dtype: object
```

[10]:
```
# df [" "  , " " , " "] - by using this format we can extract the more than 1␣
 ↪column from the dataset

df[["Name", "Sex","Ticket","Cabin"]]
```

[10]:
```
                                                 Name     Sex  \
0                             Braund, Mr. Owen Harris    male
1    Cumings, Mrs. John Bradley (Florence Briggs Th…  female
2                                Heikkinen, Miss. Laina  female
3         Futrelle, Mrs. Jacques Heath (Lily May Peel)  female
4                               Allen, Mr. William Henry    male
..                                                 …     …
886                              Montvila, Rev. Juozas    male
887                       Graham, Miss. Margaret Edith  female
888           Johnston, Miss. Catherine Helen "Carrie"  female
889                              Behr, Mr. Karl Howell    male
890                                Dooley, Mr. Patrick    male

               Ticket Cabin
0           A/5 21171   NaN
1            PC 17599   C85
2    STON/O2. 3101282   NaN
3              113803  C123
4              373450   NaN
..                ...   …
886            211536   NaN
```

```
887            112053   B42
888        W./C. 6607   NaN
889            111369  C148
890            370376   NaN

[891 rows x 4 columns]
```

[11]:
```
# This is how we specifically extract the objects(string) from the dtypes or datatypes

df.dtypes=="object"
```

[11]:
```
PassengerId    False
Survived       False
Pclass         False
Name            True
Sex             True
Age            False
SibSp          False
Parch          False
Ticket          True
Fare           False
Cabin           True
Embarked        True
dtype: bool
```

[13]:
```
df.dtypes[df.dtypes=="object"].index
```

[13]:
```
Index(['Name', 'Sex', 'Ticket', 'Cabin', 'Embarked'], dtype='object')
```

[15]:
```
type(df.dtypes[df.dtypes=="object"])
```

[15]:
```
pandas.core.series.Series
```

[17]:
```
df[df.dtypes[df.dtypes=="object"].index].describe()
```

[17]:
```
                           Name   Sex  Ticket    Cabin Embarked
count                       891   891     891      204      889
unique                      891     2     681      147        3
top     Braund, Mr. Owen Harris  male  347082  B96 B98        S
freq                          1   577       7        4      644
```

[18]:
```
df.dtypes
```

[18]:
```
PassengerId      int64
Survived         int64
Pclass           int64
```

```
Name            object
Sex             object
Age            float64
SibSp            int64
Parch            int64
Ticket          object
Fare           float64
Cabin           object
Embarked        object
dtype: object
```

[19]:
```
df[df.dtypes[df.dtypes=="float"].index]
```

[19]:
```
      Age     Fare
0    22.0   7.2500
1    38.0  71.2833
2    26.0   7.9250
3    35.0  53.1000
4    35.0   8.0500
..    …       …
886  27.0  13.0000
887  19.0  30.0000
888   NaN  23.4500
889  26.0  30.0000
890  32.0   7.7500

[891 rows x 2 columns]
```

[22]:
```
# df.dtypes ( ) function gives the data type of all types of data that present␣
 ↪in dataset

df.dtypes
```

[22]:
```
PassengerId      int64
Survived         int64
Pclass           int64
Name            object
Sex             object
Age            float64
SibSp            int64
Parch            int64
Ticket          object
Fare           float64
Cabin           object
Embarked        object
dtype: object
```

[23]:
```
df[df.dtypes[df.dtypes=="int64"].index]
```

[23]:
```
     PassengerId  Survived  Pclass  SibSp  Parch
0              1         0       3      1      0
1              2         1       1      1      0
2              3         1       3      0      0
3              4         1       1      1      0
4              5         0       3      0      0
..           …       …     …    …    …
886          887         0       2      0      0
887          888         1       1      0      0
888          889         0       3      1      2
889          890         1       1      0      0
890          891         0       3      0      0

[891 rows x 5 columns]
```

[24]:
```
# By using the df.columns ( ) function we can extract the all columns name␣
 ↪easily from the dataset

df.columns
```

[24]:
```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
      dtype='object')
```

[25]:
```
# By passing the name of column inside the df[ " "] function we get the values␣
 ↪of that particular column easily

df["Ticket"]
```

[25]:
```
0             A/5 21171
1              PC 17599
2      STON/O2. 3101282
3                113803
4                373450
             …
886              211536
887              112053
888          W./C. 6607
889              111369
890              370376
Name: Ticket, Length: 891, dtype: object
```

[26]:
```
# By passing the name of column inside the df[ " "] function we get the values␣
 ↪of that particular column easily in a TABLE FORMAT
```

```
df[["Ticket"]]
```

[26]:
```
               Ticket
0           A/5 21171
1            PC 17599
2    STON/O2. 3101282
3              113803
4              373450
..                ...
886            211536
887            112053
888        W./C. 6607
889            111369
890            370376

[891 rows x 1 columns]
```

[27]:
```
# By passing the name of column inside the df[ " "] function we get the values␣
 ↪of that particular column easily and we select the range of no that we want␣
 ↪so we define them it also , it same as SLICING OPERATION

df[["Ticket"]][4:11]
```

[27]:
```
     Ticket
4    373450
5    330877
6     17463
7    349909
8    347742
9    237736
10  PP 9549
```

[28]:
```
# By passing the name of column inside the df[ " "] function we get the values␣
 ↪of that particular column easily and we select the range of no that we want␣
 ↪so we define them it also, it same as SLICING OPERATION

df[["Ticket","Cabin"]][4:11]
```

[28]:
```
     Ticket Cabin
4    373450   NaN
5    330877   NaN
6     17463   E46
7    349909   NaN
8    347742   NaN
9    237736   NaN
10  PP 9549    G6
```

[29]:
```
# By passing the name of column inside the df[ " "] function we get the values␣
↪of that particular column easily and we select the range of no that we want␣
↪so we define them it also

df[["Ticket","Cabin"]][4:11:2]
```

[29]:
```
     Ticket Cabin
4    373450   NaN
6     17463   E46
8    347742   NaN
10  PP 9549    G6
```

[30]:
```
# It simply gives the import dataset

df
```

[30]:
```
     PassengerId  Survived  Pclass  \
0              1         0       3
1              2         1       1
2              3         1       3
3              4         1       1
4              5         0       3
..           …       …     …
886          887         0       2
887          888         1       1
888          889         0       3
889          890         1       1
890          891         0       3

                                                 Name     Sex   Age  SibSp  \
0                             Braund, Mr. Owen Harris    male  22.0      1
1    Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
2                              Heikkinen, Miss. Laina  female  26.0      0
3        Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                            Allen, Mr. William Henry    male  35.0      0
..                                                 …     …   …    …
886                            Montvila, Rev. Juozas    male  27.0      0
887                     Graham, Miss. Margaret Edith  female  19.0      0
888         Johnston, Miss. Catherine Helen "Carrie"  female   NaN      1
889                            Behr, Mr. Karl Howell    male  26.0      0
890                              Dooley, Mr. Patrick    male  32.0      0

     Parch            Ticket     Fare Cabin Embarked
0        0         A/5 21171   7.2500   NaN        S
1        0          PC 17599  71.2833   C85        C
2        0  STON/O2. 3101282   7.9250   NaN        S
3        0            113803  53.1000  C123        S
```

```
4        0            373450   8.0500   NaN        S
..     …                …       …   …      …
886      0            211536  13.0000   NaN        S
887      0            112053  30.0000   B42        S
888      2        W./C. 6607  23.4500   NaN        S
889      0            111369  30.0000  C148        C
890      0            370376   7.7500   NaN        Q

[891 rows x 12 columns]
```

[31]:
```
df["new_col"]=0
```

[32]:
```
df
```

[32]:
```
     PassengerId  Survived  Pclass  \
0              1         0       3
1              2         1       1
2              3         1       3
3              4         1       1
4              5         0       3
..           …       …     …
886          887         0       2
887          888         1       1
888          889         0       3
889          890         1       1
890          891         0       3

                                                 Name     Sex   Age  SibSp  \
0                             Braund, Mr. Owen Harris    male  22.0      1
1    Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
2                                Heikkinen, Miss. Laina  female  26.0      0
3          Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                               Allen, Mr. William Henry    male  35.0      0
..                                                   …     …   …    …
886                               Montvila, Rev. Juozas    male  27.0      0
887                        Graham, Miss. Margaret Edith  female  19.0      0
888            Johnston, Miss. Catherine Helen "Carrie"  female   NaN      1
889                               Behr, Mr. Karl Howell    male  26.0      0
890                                 Dooley, Mr. Patrick    male  32.0      0

     Parch            Ticket     Fare Cabin Embarked  new_col
0        0         A/5 21171   7.2500   NaN        S        0
1        0          PC 17599  71.2833   C85        C        0
2        0  STON/O2. 3101282   7.9250   NaN        S        0
3        0            113803  53.1000  C123        S        0
4        0            373450   8.0500   NaN        S        0
..     …               …       …   …      …       …
```

```
886      0            211536  13.0000   NaN        S        0
887      0            112053  30.0000   B42        S        0
888      2        W./C. 6607  23.4500   NaN        S        0
889      0            111369  30.0000  C148        C        0
890      0            370376   7.7500   NaN        Q        0

[891 rows x 13 columns]
```

[33]:
```
df["new_col1"]=df["PassengerId"]+df["Pclass"]
```

[34]:
```
df
```

[34]:
```
     PassengerId  Survived  Pclass  \
0              1         0       3
1              2         1       1
2              3         1       3
3              4         1       1
4              5         0       3
..           …       …     …
886          887         0       2
887          888         1       1
888          889         0       3
889          890         1       1
890          891         0       3

                                                 Name     Sex   Age  SibSp  \
0                             Braund, Mr. Owen Harris    male  22.0      1
1    Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
2                                Heikkinen, Miss. Laina  female  26.0      0
3          Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                               Allen, Mr. William Henry    male  35.0      0
..                                                   …     …   …    …
886                              Montvila, Rev. Juozas    male  27.0      0
887                       Graham, Miss. Margaret Edith  female  19.0      0
888            Johnston, Miss. Catherine Helen "Carrie"  female   NaN      1
889                              Behr, Mr. Karl Howell    male  26.0      0
890                                Dooley, Mr. Patrick    male  32.0      0

     Parch            Ticket     Fare Cabin Embarked  new_col  new_col1
0        0         A/5 21171   7.2500   NaN        S        0         4
1        0          PC 17599  71.2833   C85        C        0         3
2        0  STON/O2. 3101282   7.9250   NaN        S        0         6
3        0            113803  53.1000  C123        S        0         5
4        0            373450   8.0500   NaN        S        0         8
..     …               …      …   …      …       …        …
886      0            211536  13.0000   NaN        S        0       889
887      0            112053  30.0000   B42        S        0       889
```

```
888      2        W./C. 6607  23.4500   NaN        S        0       892
889      0            111369  30.0000  C148        C        0       891
890      0            370376   7.7500   NaN        Q        0       894

[891 rows x 14 columns]
```

[35]:
```
df.head()
```

[35]:
```
   PassengerId  Survived  Pclass  \
0            1         0       3
1            2         1       1
2            3         1       3
3            4         1       1
4            5         0       3

                                                Name     Sex   Age  SibSp  \
0                            Braund, Mr. Owen Harris    male  22.0      1
1  Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
2                             Heikkinen, Miss. Laina  female  26.0      0
3       Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                           Allen, Mr. William Henry    male  35.0      0

   Parch            Ticket     Fare Cabin Embarked  new_col  new_col1
0      0         A/5 21171   7.2500   NaN        S        0         4
1      0          PC 17599  71.2833   C85        C        0         3
2      0  STON/O2. 3101282   7.9250   NaN        S        0         6
3      0            113803  53.1000  C123        S        0         5
4      0            373450   8.0500   NaN        S        0         8
```

[36]:
```
# tail ( ) function gives the last 5 values or 5 set of values of the dataset

df.tail()
```

[36]:
```
     PassengerId  Survived  Pclass                                      Name  \
886          887         0       2                     Montvila, Rev. Juozas
887          888         1       1              Graham, Miss. Margaret Edith
888          889         0       3  Johnston, Miss. Catherine Helen "Carrie"
889          890         1       1                     Behr, Mr. Karl Howell
890          891         0       3                       Dooley, Mr. Patrick

        Sex   Age  SibSp  Parch      Ticket   Fare Cabin Embarked  new_col  \
886    male  27.0      0      0      211536  13.00   NaN        S        0
887  female  19.0      0      0      112053  30.00   B42        S        0
888  female   NaN      1      2  W./C. 6607  23.45   NaN        S        0
889    male  26.0      0      0      111369  30.00  C148        C        0
890    male  32.0      0      0      370376   7.75   NaN        Q        0
```

```
     new_col1
886       889
887       889
888       892
889       891
890       894
```

[37]:
```
# By passing the name of column inside the df[ " "] function we get the values␣
↪of that particular column easily

df["Pclass"]
```

[37]:
```
0      3
1      1
2      3
3      1
4      3
      ..
886    2
887    1
888    3
889    1
890    3
Name: Pclass, Length: 891, dtype: int64
```

[38]:
```
# pd.categorical ( ) function gives categorical or string as well as object␣
↪values

pd.Categorical(df["Pclass"])
```

[38]:
```
[3, 1, 3, 1, 3, …, 2, 1, 3, 1, 3]
Length: 891
Categories (3, int64): [1, 2, 3]
```

[40]:
```
# pd.categorical ( ) function gives categorical or string values

pd.Categorical(df["Survived"])
```

[40]:
```
[0, 1, 1, 1, 0, …, 0, 1, 0, 1, 0]
Length: 891
Categories (2, int64): [0, 1]
```

[42]:
```
# by using " unique ( ) " function we get the uinque values from the particular␣
↪column

df["Cabin"].unique()
```

[42]:
```
array([nan, 'C85', 'C123', 'E46', 'G6', 'C103', 'D56', 'A6',
       'C23 C25 C27', 'B78', 'D33', 'B30', 'C52', 'B28', 'C83', 'F33',
       'F G73', 'E31', 'A5', 'D10 D12', 'D26', 'C110', 'B58 B60', 'E101',
       'F E69', 'D47', 'B86', 'F2', 'C2', 'E33', 'B19', 'A7', 'C49', 'F4',
       'A32', 'B4', 'B80', 'A31', 'D36', 'D15', 'C93', 'C78', 'D35',
       'C87', 'B77', 'E67', 'B94', 'C125', 'C99', 'C118', 'D7', 'A19',
       'B49', 'D', 'C22 C26', 'C106', 'C65', 'E36', 'C54',
       'B57 B59 B63 B66', 'C7', 'E34', 'C32', 'B18', 'C124', 'C91', 'E40',
       'T', 'C128', 'D37', 'B35', 'E50', 'C82', 'B96 B98', 'E10', 'E44',
       'A34', 'C104', 'C111', 'C92', 'E38', 'D21', 'E12', 'E63', 'A14',
       'B37', 'C30', 'D20', 'B79', 'E25', 'D46', 'B73', 'C95', 'B38',
       'B39', 'B22', 'C86', 'C70', 'A16', 'C101', 'C68', 'A10', 'E68',
       'B41', 'A20', 'D19', 'D50', 'D9', 'A23', 'B50', 'A26', 'D48',
       'E58', 'C126', 'B71', 'B51 B53 B55', 'D49', 'B5', 'B20', 'F G63',
       'C62 C64', 'E24', 'C90', 'C45', 'E8', 'B101', 'D45', 'C46', 'D30',
       'E121', 'D11', 'E77', 'F38', 'B3', 'D6', 'B82 B84', 'D17', 'A36',
       'B102', 'B69', 'E49', 'C47', 'D28', 'E17', 'A24', 'C50', 'B42',
       'C148'], dtype=object)
```

[43]:
```
# This " head ( ) " function gives the 5 set of intial data from the dataset

df.head()
```

[43]:

	PassengerId	Survived	Pclass	\
0	1	0	3	
1	2	1	1	
2	3	1	3	
3	4	1	1	
4	5	0	3	

	Name	Sex	Age	SibSp	\
0	Braund, Mr. Owen Harris	male	22.0	1	
1	Cumings, Mrs. John Bradley (Florence Briggs Th…	female	38.0	1	
2	Heikkinen, Miss. Laina	female	26.0	0	
3	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	
4	Allen, Mr. William Henry	male	35.0	0	

	Parch	Ticket	Fare	Cabin	Embarked	new_col	new_col1
0	0	A/5 21171	7.2500	NaN	S	0	4
1	0	PC 17599	71.2833	C85	C	0	3
2	0	STON/O2. 3101282	7.9250	NaN	S	0	6
3	0	113803	53.1000	C123	S	0	5
4	0	373450	8.0500	NaN	S	0	8

[47]:
```
# This is how we get the age more than 20 by using this simple operation

df[df["Age"]>20]
```

[47]:
```
     PassengerId  Survived  Pclass  \
0              1         0       3
1              2         1       1
2              3         1       3
3              4         1       1
4              5         0       3
..           …       …     …
884          885         0       3
885          886         0       3
886          887         0       2
889          890         1       1
890          891         0       3

                                                 Name     Sex   Age  SibSp  \
0                             Braund, Mr. Owen Harris    male  22.0      1
1    Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
2                             Heikkinen, Miss. Laina  female  26.0      0
3         Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
4                           Allen, Mr. William Henry    male  35.0      0
..                                               …     …   …    …
884                           Sutehall, Mr. Henry Jr    male  25.0      0
885             Rice, Mrs. William (Margaret Norton)  female  39.0      0
886                           Montvila, Rev. Juozas    male  27.0      0
889                           Behr, Mr. Karl Howell    male  26.0      0
890                             Dooley, Mr. Patrick    male  32.0      0

     Parch            Ticket     Fare Cabin Embarked  new_col  new_col1
0        0         A/5 21171   7.2500   NaN        S        0         4
1        0          PC 17599  71.2833   C85        C        0         3
2        0  STON/O2. 3101282   7.9250   NaN        S        0         6
3        0            113803  53.1000  C123        S        0         5
4        0            373450   8.0500   NaN        S        0         8
..     …               …      …   …      …       …        …
884      0   SOTON/OQ 392076   7.0500   NaN        S        0       888
885      5            382652  29.1250   NaN        Q        0       889
886      0            211536  13.0000   NaN        S        0       889
889      0            111369  30.0000  C148        C        0       891
890      0            370376   7.7500   NaN        Q        0       894

[535 rows x 14 columns]
```

[48]:
```
len(df[df["Age"]>20])
```

[48]:
```
535
```

[55]:
```
# This is how we extract the fare values from the fare column more than 32 by␣
 ↪performing the simple operation
```

```
df[df["Fare"]>32.204208]
```

[55]:
```
     PassengerId  Survived  Pclass  \
1              2         1       1
3              4         1       1
6              7         0       1
23            24         1       1
27            28         0       1
..           …       …     …
856          857         1       1
863          864         0       3
867          868         0       1
871          872         1       1
879          880         1       1

                                                  Name     Sex   Age  SibSp  \
1    Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
3         Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
6                              McCarthy, Mr. Timothy J    male  54.0      0
23                          Sloper, Mr. William Thompson    male  28.0      0
27                       Fortune, Mr. Charles Alexander    male  19.0      3
..                                                  …     …   …    …
856          Wick, Mrs. George Dennick (Mary Hitchcock)  female  45.0      1
863                  Sage, Miss. Dorothy Edith "Dolly"  female   NaN      8
867                Roebling, Mr. Washington Augustus II    male  31.0      0
871   Beckwith, Mrs. Richard Leonard (Sallie Monypeny)  female  47.0      1
879      Potter, Mrs. Thomas Jr (Lily Alexenia Wilson)  female  56.0      0

     Parch    Ticket      Fare        Cabin Embarked  new_col  new_col1
1        0  PC 17599   71.2833          C85        C        0         3
3        0    113803   53.1000         C123        S        0         5
6        0     17463   51.8625          E46        S        0         8
23       0    113788   35.5000           A6        S        0        25
27       2     19950  263.0000  C23 C25 C27        S        0        29
..     …       …       …          …      …       …        …
856      1     36928  164.8667          NaN        S        0       858
863      2  CA. 2343   69.5500          NaN        S        0       867
867      0  PC 17590   50.4958          A24        S        0       869
871      1     11751   52.5542          D35        S        0       873
879      1     11767   83.1583          C50        C        0       881

[211 rows x 14 columns]
```

[56]:
```
# This  is how we extract the Fare values from the Fare column by performing␣
 ↪the simple operation
```

```
df[df["Fare"]==0]
```

[56]:

	PassengerId	Survived	Pclass	Name	Sex	\
179	180	0	3	Leonard, Mr. Lionel	male	
263	264	0	1	Harrison, Mr. William	male	
271	272	1	3	Tornquist, Mr. William Henry	male	
277	278	0	2	Parkes, Mr. Francis "Frank"	male	
302	303	0	3	Johnson, Mr. William Cahoone Jr	male	
413	414	0	2	Cunningham, Mr. Alfred Fleming	male	
466	467	0	2	Campbell, Mr. William	male	
481	482	0	2	Frost, Mr. Anthony Wood "Archie"	male	
597	598	0	3	Johnson, Mr. Alfred	male	
633	634	0	1	Parr, Mr. William Henry Marsh	male	
674	675	0	2	Watson, Mr. Ennis Hastings	male	
732	733	0	2	Knight, Mr. Robert J	male	
806	807	0	1	Andrews, Mr. Thomas Jr	male	
815	816	0	1	Fry, Mr. Richard	male	
822	823	0	1	Reuchlin, Jonkheer. John George	male	

	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	new_col	new_col1
179	36.0	0	0	LINE	0.0	NaN	S	0	183
263	40.0	0	0	112059	0.0	B94	S	0	265
271	25.0	0	0	LINE	0.0	NaN	S	0	275
277	NaN	0	0	239853	0.0	NaN	S	0	280
302	19.0	0	0	LINE	0.0	NaN	S	0	306
413	NaN	0	0	239853	0.0	NaN	S	0	416
466	NaN	0	0	239853	0.0	NaN	S	0	469
481	NaN	0	0	239854	0.0	NaN	S	0	484
597	49.0	0	0	LINE	0.0	NaN	S	0	601
633	NaN	0	0	112052	0.0	NaN	S	0	635
674	NaN	0	0	239856	0.0	NaN	S	0	677
732	NaN	0	0	239855	0.0	NaN	S	0	735
806	39.0	0	0	112050	0.0	A36	S	0	808
815	NaN	0	0	112058	0.0	B102	S	0	817
822	38.0	0	0	19972	0.0	NaN	S	0	824

[57]:
```
# This  is how we extract the males from the sex column by performing the␣
 ↪simple operation

df[df["Sex"]=="male"]
```

[57]:

	PassengerId	Survived	Pclass	Name	Sex	\
0	1	0	3	Braund, Mr. Owen Harris	male	
4	5	0	3	Allen, Mr. William Henry	male	
5	6	0	3	Moran, Mr. James	male	
6	7	0	1	McCarthy, Mr. Timothy J	male	
7	8	0	3	Palsson, Master. Gosta Leonard	male	

```
..          …       …     …                                …   …
883          884         0       2   Banfield, Mr. Frederick James  male
884          885         0       3          Sutehall, Mr. Henry Jr  male
886          887         0       2           Montvila, Rev. Juozas  male
889          890         1       1           Behr, Mr. Karl Howell  male
890          891         0       3             Dooley, Mr. Patrick  male

      Age  SibSp  Parch            Ticket     Fare Cabin Embarked  new_col  \
0    22.0      1      0         A/5 21171   7.2500   NaN        S        0
4    35.0      0      0            373450   8.0500   NaN        S        0
5     NaN      0      0            330877   8.4583   NaN        Q        0
6    54.0      0      0             17463  51.8625   E46        S        0
7     2.0      3      1            349909  21.0750   NaN        S        0
..    …    …    …                …       …   …       …       …
883  28.0      0      0  C.A./SOTON 34068  10.5000   NaN        S        0
884  25.0      0      0   SOTON/OQ 392076   7.0500   NaN        S        0
886  27.0      0      0            211536  13.0000   NaN        S        0
889  26.0      0      0            111369  30.0000  C148        C        0
890  32.0      0      0            370376   7.7500   NaN        Q        0

     new_col1
0           4
4           8
5           9
6           8
7          11
..        …
883      886
884      888
886      889
889      891
890      894

[577 rows x 14 columns]
```

[58]:
```
# By using this operation we extract the Pclass or Passenger class that is␣
 ↪equal to 1

df[df["Pclass"]==1]
```

[58]:
```
     PassengerId  Survived  Pclass  \
1              2         1       1
3              4         1       1
6              7         0       1
11            12         1       1
23            24         1       1
..           …       …     …
```

```
871          872         1       1
872          873         0       1
879          880         1       1
887          888         1       1
889          890         1       1

                                                  Name     Sex   Age  SibSp  \
1    Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
3         Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
6                              McCarthy, Mr. Timothy J    male  54.0      0
11                            Bonnell, Miss. Elizabeth  female  58.0      0
23                          Sloper, Mr. William Thompson    male  28.0      0
..                                                   …     …   …    …
871   Beckwith, Mrs. Richard Leonard (Sallie Monypeny)  female  47.0      1
872                            Carlsson, Mr. Frans Olof    male  33.0      0
879      Potter, Mrs. Thomas Jr (Lily Alexenia Wilson)  female  56.0      0
887                        Graham, Miss. Margaret Edith  female  19.0      0
889                               Behr, Mr. Karl Howell    male  26.0      0

     Parch    Ticket     Fare        Cabin Embarked  new_col  new_col1
1        0  PC 17599  71.2833          C85        C        0         3
3        0    113803  53.1000         C123        S        0         5
6        0     17463  51.8625          E46        S        0         8
11       0    113783  26.5500         C103        S        0        13
23       0    113788  35.5000           A6        S        0        25
..     …       …       …           …       …      …       …
871      1     11751  52.5542          D35        S        0       873
872      0       695   5.0000  B51 B53 B55        S        0       874
879      1     11767  83.1583          C50        C        0       881
887      0    112053  30.0000          B42        S        0       889
889      0    111369  30.0000         C148        C        0       891

[216 rows x 14 columns]
```

[60]:
```
# This is how we simply extract the females from the sex column easily

df["Sex"]=="female"
```

[60]:
```
0      False
1       True
2       True
3       True
4      False
       …
886    False
887     True
888     True
```

```
889    False
890    False
Name: Sex, Length: 891, dtype: bool
```

[62]:
```
# This is how we extract the fair more than 32

df["Fare"]>32
```

[62]:
```
0      False
1       True
2      False
3       True
4      False
       …
886    False
887    False
888    False
889    False
890    False
Name: Fare, Length: 891, dtype: bool
```

[64]:
```
# This is how we extract the values from two diffrent columns like we extract␣
 ↪female from sex column and 32 from fair column at the same time

df[(df["Sex"]=="female") & (df["Fare"]>32)]
```

[64]:
```
     PassengerId  Survived  Pclass  \
1              2         1       1
3              4         1       1
31            32         1       1
43            44         1       2
52            53         1       1
..           …       …     …
853          854         1       1
856          857         1       1
863          864         0       3
871          872         1       1
879          880         1       1

                                                  Name     Sex   Age  SibSp  \
1    Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
3         Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
31      Spencer, Mrs. William Augustus (Marie Eugenie)  female   NaN      1
43            Laroche, Miss. Simonne Marie Anne Andree  female   3.0      1
52            Harper, Mrs. Henry Sleeper (Myna Haxtun)  female  49.0      1
..                                                 …     …   …    …
853                         Lines, Miss. Mary Conover  female  16.0      0
```

```
856                        Wick, Mrs. George Dennick (Mary Hitchcock)  female  45.0      1
863                                Sage, Miss. Dorothy Edith "Dolly"  female   NaN      8
871                 Beckwith, Mrs. Richard Leonard (Sallie Monypeny)  female  47.0      1
879                    Potter, Mrs. Thomas Jr (Lily Alexenia Wilson)  female  56.0      0

     Parch         Ticket      Fare Cabin Embarked  new_col  new_col1
1        0       PC 17599   71.2833   C85        C        0         3
3        0         113803   53.1000  C123        S        0         5
31       0       PC 17569  146.5208   B78        C        0        33
43       2  SC/Paris 2123   41.5792   NaN        C        0        46
52       0       PC 17572   76.7292   D33        C        0        54
..     …            …       …   …      …       …        …
853      1       PC 17592   39.4000   D28        S        0       855
856      1          36928  164.8667   NaN        S        0       858
863      2       CA. 2343   69.5500   NaN        S        0       867
871      1          11751   52.5542   D35        S        0       873
879      1          11767   83.1583   C50        C        0       881

[104 rows x 14 columns]
```

[65]:
```
# This is how we extract the values from two diffrent columns like we extract␣
 ↪male from sex column and 32 from fair column at the same time

df[(df["Sex"]=="female") | (df["Fare"]>32)]
```

[65]:
```
     PassengerId  Survived  Pclass  \
1              2         1       1
2              3         1       3
3              4         1       1
6              7         0       1
8              9         1       3
..           …       …     …
880          881         1       2
882          883         0       3
885          886         0       3
887          888         1       1
888          889         0       3

                                                  Name     Sex   Age  SibSp  \
1    Cumings, Mrs. John Bradley (Florence Briggs Th…  female  38.0      1
2                                Heikkinen, Miss. Laina  female  26.0      0
3         Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  35.0      1
6                               McCarthy, Mr. Timothy J    male  54.0      0
8    Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)  female  27.0      0
..                                                 …     …   …    …
880       Shelley, Mrs. William (Imanita Parrish Hall)  female  25.0      0
882                         Dahlberg, Miss. Gerda Ulrika  female  22.0      0
```

```
885                Rice, Mrs. William (Margaret Norton)  female  39.0      0
887                       Graham, Miss. Margaret Edith  female  19.0      0
888             Johnston, Miss. Catherine Helen "Carrie"  female   NaN      1

     Parch            Ticket     Fare Cabin Embarked  new_col  new_col1
1        0         PC 17599  71.2833   C85        C        0         3
2        0  STON/O2. 3101282   7.9250   NaN        S        0         6
3        0            113803  53.1000  C123        S        0         5
6        0             17463  51.8625   E46        S        0         8
8        2            347742  11.1333   NaN        S        0        12
..     …               …       …   …      …       …        …
880      1            230433  26.0000   NaN        S        0       883
882      0              7552  10.5167   NaN        S        0       886
885      5            382652  29.1250   NaN        Q        0       889
887      0            112053  30.0000   B42        S        0       889
888      2        W./C. 6607  23.4500   NaN        S        0       892

[421 rows x 14 columns]
```

[66]:
```
# This is how we extract the two columns with given conditions like we want␣
 ↪male and fare more than 32

df[(df["Sex"]=="male") & (df["Fare"]>32)]
```

[66]:
```
     PassengerId  Survived  Pclass                                  Name  \
6              7         0       1               McCarthy, Mr. Timothy J
23            24         1       1          Sloper, Mr. William Thompson
27            28         0       1       Fortune, Mr. Charles Alexander
34            35         0       1              Meyer, Mr. Edgar Joseph
35            36         0       1       Holverson, Mr. Alexander Oskar
..           …       …     …                                       …
827          828         1       2                Mallet, Master. Andre
838          839         1       3                     Chip, Mr. Chang
846          847         0       3             Sage, Mr. Douglas Bullen
848          849         0       2                    Harper, Rev. John
867          868         0       1  Roebling, Mr. Washington Augustus II

      Sex   Age  SibSp  Parch           Ticket      Fare        Cabin  \
6    male  54.0      0      0            17463   51.8625          E46
23   male  28.0      0      0           113788   35.5000           A6
27   male  19.0      3      2            19950  263.0000  C23 C25 C27
34   male  28.0      1      0         PC 17604   82.1708          NaN
35   male  42.0      1      0           113789   52.0000          NaN
..    …    …     …    …              …       …         …
827  male   1.0      0      2  S.C./PARIS 2079   37.0042          NaN
838  male  32.0      0      0             1601   56.4958          NaN
846  male   NaN      8      2         CA. 2343   69.5500          NaN
```

```
848  male  28.0      0      1           248727   33.0000          NaN
867  male  31.0      0      0        PC 17590   50.4958          A24

    Embarked  new_col  new_col1
6          S        0         8
23         S        0        25
27         S        0        29
34         C        0        36
35         S        0        37
..       …      …       …
827        C        0       830
838        S        0       842
846        S        0       850
848        S        0       851
867        S        0       869

[107 rows x 14 columns]
```

[67]:
```
df["Fare"]
```

[67]:
```
0       7.2500
1      71.2833
2       7.9250
3      53.1000
4       8.0500
        …
886    13.0000
887    30.0000
888    23.4500
889    30.0000
890     7.7500
Name: Fare, Length: 891, dtype: float64
```

[68]:
```
# By using this " max ( ) " function we can extract the maximum value of price␣
 ↪or fare etc

max(df["Fare"])
```

[68]:
```
512.3292
```

[72]:
```
# This is how we get the name of person who pay the maximum amount of money as␣
 ↪a fair or ticket

df[df["Fare"]==max(df["Fare"])]["Name"]
```

[72]:
```
258                      Ward, Miss. Anna
679    Cardeza, Mr. Thomas Drake Martinez
```

```
737                Lesurer, Mr. Gustave J
Name: Name, dtype: object
```

[69]:
```
min(df["Fare"])
```

[69]:
```
0.0
```

[73]:
```
# head ( ) function gives the first five columns of data

df.head()
```

[73]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	new_col	new_col1
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S	0	4
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th…	female	38.0	1	0	PC 17599	71.2833	C85	C	0	3
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S	0	6
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S	0	5
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S	0	8

[74]:
```
# This is how we extract the row

df[0:2]
```

[74]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	new_col	new_col1
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S	0	4
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th…	female	38.0	1	0	PC 17599	71.2833	C85	C	0	3

[76]:
```
# loc = loc is nothing but a location in which we choose a no from which
  ↪location to which location no we want a data
df.loc[0:2]
```

[76]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	new_col	new_col1
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S	0	4
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th…	female	38.0	1	0	PC 17599	71.2833	C85	C	0	3
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S	0	6

THANK YOU SO MUCH !!

YOURS VIRAT TIWARI :)