SHETH L.U.J. & SIR M.V. COLLEGE

Rajanish bhardwaj | T073

Data Wrangling

3.1 Creating a Data Frame

```
import pandas as pd
import numpy as np

df = pd.read_csv("/content/Titanic-Dataset.csv")
print("Loaded shape:", df.shape)
df.head(2)
```

```
Loaded shape: (891, 12)
```

	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	female	38.0	1	0	PC 17599	71.2833	C85	C

3.2 Describing the Data

```
print("Dimensions:", df.shape)
df.describe(include='all')
```

```
Dimensions: (891, 12)
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
count	891.000000	891.000000	891.000000	891	891	714.000000	891.000000	891.000000	891	891.000000	204	889
unique	NaN	NaN	NaN	891	2	NaN	NaN	NaN	681	NaN	147	3
top	NaN	NaN	NaN	Dooley, Mr. Patrick	male	NaN	NaN	NaN	347082	NaN	G6	S
freq	NaN	NaN	NaN	1	577	NaN	NaN	NaN	7	NaN	4	644
mean	446.000000	0.383838	2.308642	NaN	NaN	29.699118	0.523008	0.381594	NaN	32.204208	NaN	NaN
std	257.353842	0.486592	0.836071	NaN	NaN	14.526497	1.102743	0.806057	NaN	49.693429	NaN	NaN
min	1.000000	0.000000	1.000000	NaN	NaN	0.420000	0.000000	0.000000	NaN	0.000000	NaN	NaN
25%	223.500000	0.000000	2.000000	NaN	NaN	20.125000	0.000000	0.000000	NaN	7.910400	NaN	NaN
50%	446.000000	0.000000	3.000000	NaN	NaN	28.000000	0.000000	0.000000	NaN	14.454200	NaN	NaN
75%	668.500000	1.000000	3.000000	NaN	NaN	38.000000	1.000000	0.000000	NaN	31.000000	NaN	NaN
max	891.000000	1.000000	3.000000	NaN	NaN	80.000000	8.000000	6.000000	NaN	512.329200	NaN	NaN

3.3 Navigating DataFrames

```
print(df.iloc[0])
print(df.iloc[1:4])
print(df.iloc[:4])
```

```
PassengerId                        1
Survived                           0
Pclass                             3
Name           Braund, Mr. Owen Harris
Sex                             male
Age                             22.0
SibSp                              1
Parch                              0
Ticket                     A/5 21171
Fare                            7.25
Cabin                            NaN
Embarked                           S
Name: 0, dtype: object
   PassengerId  Survived  Pclass  \
1            2         1       1
2            3         1       3
3            4         1       1

                                                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

   Parch            Ticket     Fare Cabin Embarked
1      0          PC 17599  71.2833   C85        C
2      0  STON/O2. 3101282   7.9250   NaN        S
```

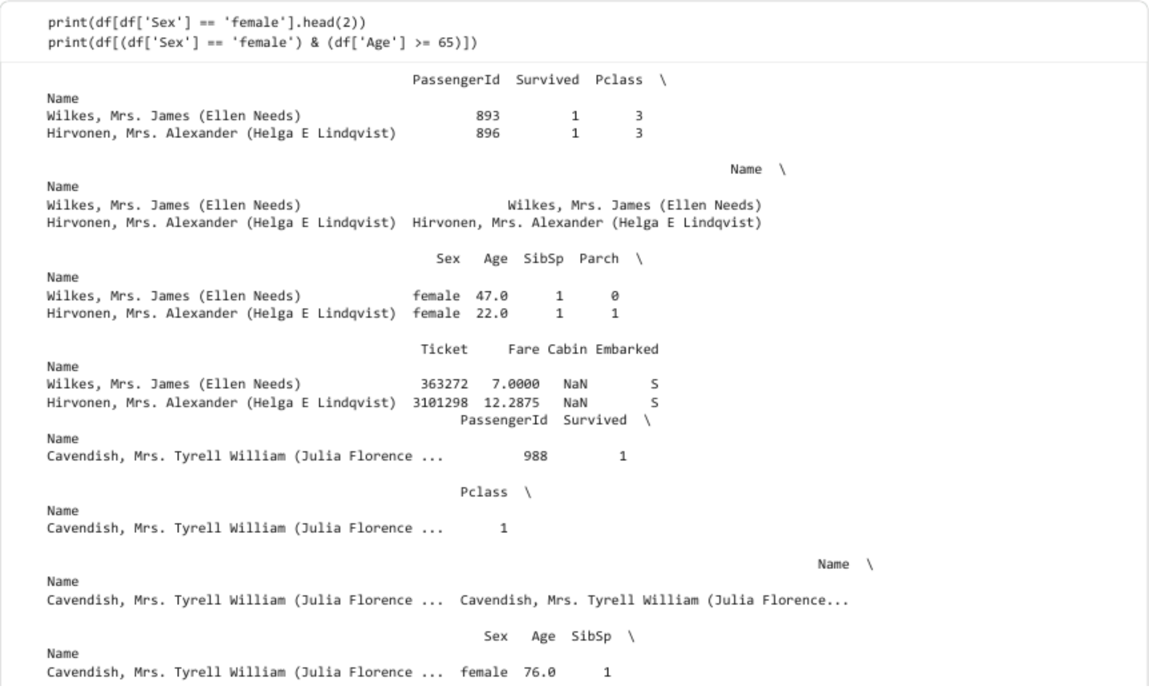
```
3      0            113803  53.1000  C123        S
   PassengerId  Survived  Pclass  \
0            1         0       3   
1            2         1       1   
2            3         1       3   
3            4         1       1   

                                                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   

   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  
```

```
df_by_name = df.set_index("Name")
df_by_name.loc[df_by_name.index[0]]
```

	Braund, Mr. Owen Harris
PassengerId	1
Survived	0
Pclass	3
Sex	male
Age	22.0
SibSp	1
Parch	0
Ticket	A/5 21171
Fare	7.25
Cabin	NaN
Embarked	S

dtype: object

3.4 Selecting Rows Based on Conditionals



```
print(df[df['Sex'] == 'female'].head(2))
print(df[(df['Sex'] == 'female') & (df['Age'] >= 65)])
```

```
                                              PassengerId  Survived  Pclass  \
Name                                                                          
Wilkes, Mrs. James (Ellen Needs)                      893         1       3   
Hirvonen, Mrs. Alexander (Helga E Lindqvist)          896         1       3   

                                                                                      Name  \
Name                                                                                         
Wilkes, Mrs. James (Ellen Needs)                          Wilkes, Mrs. James (Ellen Needs)   
Hirvonen, Mrs. Alexander (Helga E Lindqvist)  Hirvonen, Mrs. Alexander (Helga E Lindqvist)   

                                                 Sex   Age  SibSp  Parch  \
Name                                                                       
Wilkes, Mrs. James (Ellen Needs)              female  47.0      1      0   
Hirvonen, Mrs. Alexander (Helga E Lindqvist)  female  22.0      1      1   

                                               Ticket     Fare Cabin Embarked  
Name                                                                           
Wilkes, Mrs. James (Ellen Needs)               363272   7.0000   NaN        S  
Hirvonen, Mrs. Alexander (Helga E Lindqvist)  3101298  12.2875   NaN        S  
                                                    PassengerId  Survived  \
Name                                                                        
Cavendish, Mrs. Tyrell William (Julia Florence ...          988         1   

                                                    Pclass  \
Name                                                         
Cavendish, Mrs. Tyrell William (Julia Florence ...       1   

                                                                                                  Name  \
Name                                                                                                     
Cavendish, Mrs. Tyrell William (Julia Florence ...  Cavendish, Mrs. Tyrell William (Julia Florence...   

                                                       Sex   Age  SibSp  \
Name                                                                      
Cavendish, Mrs. Tyrell William (Julia Florence ...  female  76.0      1   
```

3.5 Replacing Values

```
df["Sex"].replace("female", "Woman").head(5)

df["Sex"].replace(["female", "male"], ["Woman", "Man"]) .head(8)

df.replace(1, "One").head(2)
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	One	0	3	Braund, Mr. Owen Harris	male	22.0	One	0	A/5 21171	7.2500	NaN	S

3.6 Renaming Columns

```
df = df.rename(columns={"Pclass": "Passenger Class", "Sex": "Gender"})
df.head(2)
```

	PassengerId	Survived	Passenger Class	Name	Gender	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

3.7 Finding the Min, Max, Sum, Average, and Count

```
print("Max Age:", df["Age"].max())
print("Min Age:", df["Age"].min())
print("Mean Age:", df["Age"].mean())
print("Sum of Age:", df["Age"].sum())
print("Count of Age:", df["Age"].count())

```

```
Max Age: 80.0
Min Age: 0.42
Mean Age: 29.69911764705882
Sum of Age: 21205.17
Count of Age: 714
```

```
df.var(numeric_only=True)
df.std(numeric_only=True)
df.kurt(numeric_only=True)
df.skew(numeric_only=True)
```

	0
PassengerId	0.000000
Survived	0.478523
Passenger Class	-0.630548
Age	0.389108
SibSp	3.695352
Parch	2.749117
Fare	4.787317

dtype: float64

3.8 Finding Unique Values

```
df["Gender"]. unique()
df["Gender"].value_counts()
```

	count
Gender	
male	577
female	314

dtype: int64

3.9 Handling Missing Values

```
df[df["Age"].isnull()].head(5)
```

	PassengerId	Survived	Passenger Class	Name	Gender	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
5	6	0	3	Moran, Mr. James	male	NaN	0	0	330877	8.4583	NaN	Q
17	18	1	2	Williams, Mr. Charles Eugene	male	NaN	0	0	244373	13.0000	NaN	S
19	20	1	3	Masselmani, Mrs. Fatima	female	NaN	0	0	2649	7.2250	NaN	C
26	27	0	3	Emir, Mr. Farred Chehab	male	NaN	0	0	2631	7.2250	NaN	C
28	29	1	3	O'Dwyer, Miss. Ellen "Nellie"	female	NaN	0	0	330959	7.8792	NaN	Q

3.10 Deleting a Column and 3.11 Deleting a Row

```
df.drop("Age", axis=1).head(2)

df[df["Gender"] != "Man"].head(2)

df[df["Name"] != "Allison, Miss Helen Loraine"]. shape
```

```
(891, 12)
```

3.13 Grouping Rows by Values

```
df.groupby("Gender").mean(numeric_only=True)

df.groupby("Survived")["Name"].count()

df.groupby(["Gender", "Survived"]) ["Age"].mean()
```

		Age
Gender	**Survived**	
female	**0**	25.046875
	1	28.847716
male	**0**	31.618056
	1	27.276022

dtype: float64

3.15 Looping Over a Column

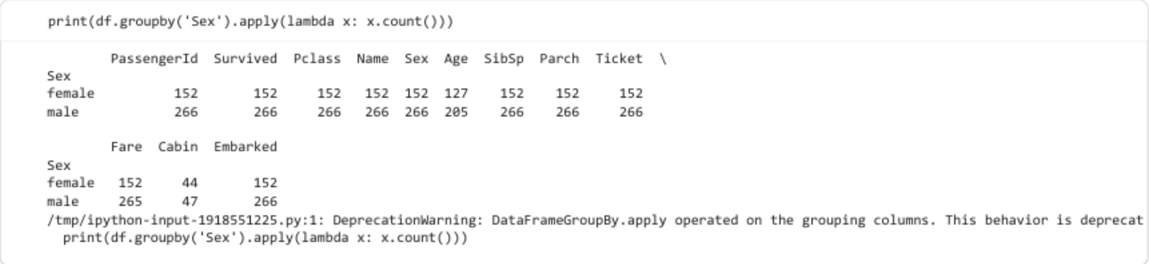
```
for name in df["Name"] [0:2]:
    print(name.upper())

print([name.upper() for name in df["Name"] [0:2]])

df["Name"].apply(lambda x: x.upper()).head(2)
```

```
BRAUND, MR. OWEN HARRIS
CUMINGS, MRS. JOHN BRADLEY (FLORENCE BRIGGS THAYER)
['BRAUND, MR. OWEN HARRIS', 'CUMINGS, MRS. JOHN BRADLEY (FLORENCE BRIGGS THAYER)']
```

	Name
0	BRAUND, MR. OWEN HARRIS
1	CUMINGS, MRS. JOHN BRADLEY (FLORENCE BRIGGS TH...

dtype: object

3.16 Applying a Function to Groups



```
print(df.groupby('Sex').apply(lambda x: x.count()))
```

```
        PassengerId  Survived  Pclass  Name  Sex  Age  SibSp  Parch  Ticket  \
Sex
female          152       152     152   152  152  127    152    152     152
male            266       266     266   266  266  205    266    266     266

        Fare  Cabin  Embarked
Sex
female   152     44       152
male     265     47       266
/tmp/ipython-input-1918551225.py:1: DeprecationWarning: DataFrameGroupBy.apply operated on the grouping columns. This behavior is deprecat
  print(df.groupby('Sex').apply(lambda x: x.count()))
```