Exploratory Data Analysis

EDA and it's 10 important steps

In [ ]:
```
# Import Dataset
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt

#Load dataset

df = sns.load_dataset('titanic')
df1 = sns.load_dataset('tips')
```

In [ ]:
```
#step-1 Data shape

df.shape
rows, cols = df.shape
print("Number of Rows: ", rows) #instances
print("Number of Cols: ", cols ) #series
```

```
Number of Rows:  891
Number of Cols:  15
```

In [ ]:
```
# Step-2 Data structure
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
 #   Column       Non-Null Count  Dtype   
---  ------       --------------  -----   
 0   survived     891 non-null    int64   
 1   pclass       891 non-null    int64   
 2   sex          891 non-null    object  
 3   age          714 non-null    float64 
 4   sibsp        891 non-null    int64   
 5   parch        891 non-null    int64   
 6   fare         891 non-null    float64 
 7   embarked     889 non-null    object  
 8   class        891 non-null    category
 9   who          891 non-null    object  
 10  adult_male   891 non-null    bool    
 11  deck         203 non-null    category
 12  embark_town  889 non-null    object  
 13  alive        891 non-null    object  
 14  alone        891 non-null    bool    
dtypes: bool(2), category(2), float64(2), int64(4), object(5)
memory usage: 80.7+ KB
```

In [ ]:
```
# Step-3 Find missing values
df.isnull().sum()
```

Out[ ]:
```
survived         0
pclass           0
sex              0
age            177
sibsp            0
parch            0
fare             0
embarked         2
class            0
who              0
adult_male       0
deck           688
embark_town      2
alive            0
alone            0
dtype: int64
```

In [ ]:
```
# percnet calculation of missing value:
df.isnull().sum() / df.shape[0] *100
```

Out[ ]:
```
survived        0.000000
pclass          0.000000
sex             0.000000
age            19.865320
sibsp           0.000000
parch           0.000000
fare            0.000000
embarked        0.224467
class           0.000000
who             0.000000
adult_male      0.000000
deck           77.216611
embark_town     0.224467
alive           0.000000
alone           0.000000
dtype: float64
```

In [ ]:
```
# df1.isnull().sum() / df1.shape[0] *100
```

In [ ]:
```
# Step-4 Spilt  variables for new coulumn needed / Feature engineering

city = pd.DataFrame(np.array([["Lahore, Pakistan",67, 100], ["Beijing, China", 5, 6],
                              columns=['address', 'males', 'females'])
city
```

Out[ ]:

	address	males	females
0	Lahore, Pakistan	67	100
1	Beijing, China	5	6
2	berlin, Germany	8	9

In [ ]:
```
city [['city', 'country']] = df1['address'].str.split(',', expand=True)
city
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
File ~\AppData\Local\Programs\Python\Python310\lib\site-packages\pandas\core\indexes
\base.py:3621, in Index.get_loc(self, key, method, tolerance)
   <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site
-packages/pandas/core/indexes/base.py?line=3619'>3620</a> try:
-> <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site
-packages/pandas/core/indexes/base.py?line=3620'>3621</a>     return self._engine.get
_loc(casted_key)
   <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site
-packages/pandas/core/indexes/base.py?line=3621'>3622</a> except KeyError as err:

File ~\AppData\Local\Programs\Python\Python310\lib\site-packages\pandas\_libs\index.p
yx:136, in pandas._libs.index.IndexEngine.get_loc()

File ~\AppData\Local\Programs\Python\Python310\lib\site-packages\pandas\_libs\index.p
yx:163, in pandas._libs.index.IndexEngine.get_loc()

File pandas\_libs\hashtable_class_helper.pxi:5198, in pandas._libs.hashtable.PyObject
HashTable.get_item()

File pandas\_libs\hashtable_class_helper.pxi:5206, in pandas._libs.hashtable.PyObject
HashTable.get_item()

KeyError: 'address'

The above exception was the direct cause of the following exception:

KeyError                                  Traceback (most recent call last)
d:\Machine_learning_ka_Chilla\07__10_eda_steps.ipynb Cell 10' in <cell line: 1>()
----> <a href='vscode-notebook-cell:/d%3A/Machine_learning_ka_Chilla/07__10_eda_step
s.ipynb#ch0000009?line=0'>1</a> city [['city', 'country']] = df1['address'].str.split
(',', expand=True)
      <a href='vscode-notebook-cell:/d%3A/Machine_learning_ka_Chilla/07__10_eda_step
s.ipynb#ch0000009?line=1'>2</a> city

File ~\AppData\Local\Programs\Python\Python310\lib\site-packages\pandas\core\frame.p
y:3505, in DataFrame.__getitem__(self, key)
   <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site
-packages/pandas/core/frame.py?line=3502'>3503</a> if self.columns.nlevels > 1:
   <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site
-packages/pandas/core/frame.py?line=3503'>3504</a>     return self._getitem_multileve
l(key)
-> <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site
-packages/pandas/core/frame.py?line=3504'>3505</a> indexer = self.columns.get_loc(ke
y)
   <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site
-packages/pandas/core/frame.py?line=3505'>3506</a> if is_integer(indexer):
   <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site
-packages/pandas/core/frame.py?line=3506'>3507</a>     indexer = [indexer]

File ~\AppData\Local\Programs\Python\Python310\lib\site-packages\pandas\core\indexes
\base.py:3623, in Index.get_loc(self, key, method, tolerance)
   <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site
-packages/pandas/core/indexes/base.py?line=3620'>3621</a>     return self._engine.get
_loc(casted_key)
   <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site
-packages/pandas/core/indexes/base.py?line=3621'>3622</a> except KeyError as err:
-> <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site
-packages/pandas/core/indexes/base.py?line=3622'>3623</a>     raise KeyError(key) fro
```

```
m err
   <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site-packages/pandas/core/indexes/base.py?line=3623'>3624</a> except TypeError:
   <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site-packages/pandas/core/indexes/base.py?line=3624'>3625</a>     # If we have a listlike key, _check_indexing_error will raise
   <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site-packages/pandas/core/indexes/base.py?line=3625'>3626</a>     #  InvalidIndexError. Otherwise we fall through and re-raise
   <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site-packages/pandas/core/indexes/base.py?line=3626'>3627</a>     #  the TypeError.
   <a href='file:///c%3A/Users/Faiza/AppData/Local/Programs/Python/Python310/lib/site-packages/pandas/core/indexes/base.py?line=3627'>3628</a>     self._check_indexing_error(key)

KeyError: 'address'
```

In [ ]:
```
#type casting/ conersion of dtype
city.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3 entries, 0 to 2
Data columns (total 5 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   address  3 non-null      object
 1   males    3 non-null      object
 2   females  3 non-null      object
 3   city     3 non-null      object
 4   country  3 non-null      object
dtypes: object(5)
memory usage: 248.0+ bytes
```

In [ ]:
```
# to convert into an int
city[['males', 'females']] = df1[['males', 'females']].astype('int64')
city.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3 entries, 0 to 2
Data columns (total 5 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   address  3 non-null      object
 1   males    3 non-null      int64 
 2   females  3 non-null      int64 
 3   city     3 non-null      object
 4   country  3 non-null      object
dtypes: int64(2), object(3)
memory usage: 248.0+ bytes
```

In [ ]:
```
# to conver into str
city[["city", "country"]] = city[["city", "country"]].astype('str') # bool float
city.info()
# why info dtype is not chnaging
```

In [ ]:
```
# STep 6 summary statistics
df.describe()
```

Out[ ]:

	survived	pclass	age	sibsp	parch	fare
count	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

In [ ]:
```
# Step-7 value count for a specific column
df['class'].value_counts()
```

Out[ ]:
```
24.00    30
22.00    27
18.00    26
19.00    25
28.00    25
         ..
36.50     1
55.50     1
0.92      1
23.50     1
74.00     1
Name: age, Length: 88, dtype: int64
```

In [ ]:
```
# finding unique values in a colu,n/seris
df['class'].unique()
```

Out[ ]:
```
['Third', 'First', 'Second']
Categories (3, object): ['First', 'Second', 'Third']
```

In [ ]:
```
# Step- 8 Deal with duplicates
df[df.embark_town == 'Queenstown']
```

Out[ ]:

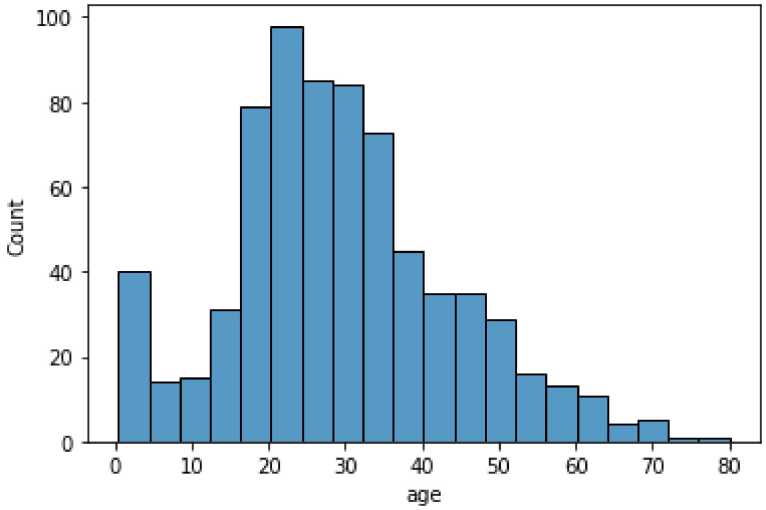
	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	de
5	0	3	male	NaN	0	0	8.4583	Q	Third	man	True	Na
16	0	3	male	2.0	4	1	29.1250	Q	Third	child	False	Na
22	1	3	female	15.0	0	0	8.0292	Q	Third	child	False	Na
28	1	3	female	NaN	0	0	7.8792	Q	Third	woman	False	Na
32	1	3	female	NaN	0	0	7.7500	Q	Third	woman	False	Na
...	...	...	...	...	...	...	...	...	...	...	...	
790	0	3	male	NaN	0	0	7.7500	Q	Third	man	True	Na
825	0	3	male	NaN	0	0	6.9500	Q	Third	man	True	Na
828	1	3	male	NaN	0	0	7.7500	Q	Third	man	True	Na
885	0	3	female	39.0	0	5	29.1250	Q	Third	woman	False	Na
890	0	3	male	32.0	0	0	7.7500	Q	Third	man	True	Na

77 rows × 15 columns

In [ ]:
```
# Check the normality / Standard normal distribution

sns.histplot(df['age'], data = df) #how to make histplot for two cariable for a contir
```

Out[ ]:
```
<AxesSubplot:xlabel='age', ylabel='Count'>
```



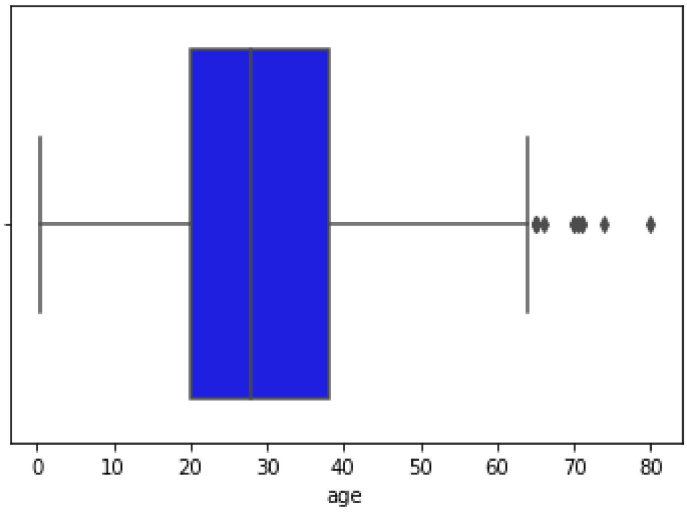
In [ ]:
```
#measure its skewness and kurtosis
df['age'].agg(['skew', 'kurtosis']).transpose()
```

Out[ ]:
```
skew        0.389108
kurtosis    0.178274
Name: age, dtype: float64
```

In [ ]:
```
sns.boxplot(df['age'], color = 'blue')
```

```
C:\Users\Faiza\AppData\Local\Programs\Python\Python310\lib\site-packages\seaborn\_decorators.py:36: FutureWarning: Pass the following variable as a keyword arg: x. From version 0.12, the only valid positional argument will be `data`, and passing other arguments without an explicit keyword will result in an error or misinterpretation.
  warnings.warn(
```

Out[ ]:
```
<AxesSubplot:xlabel='age'>
```



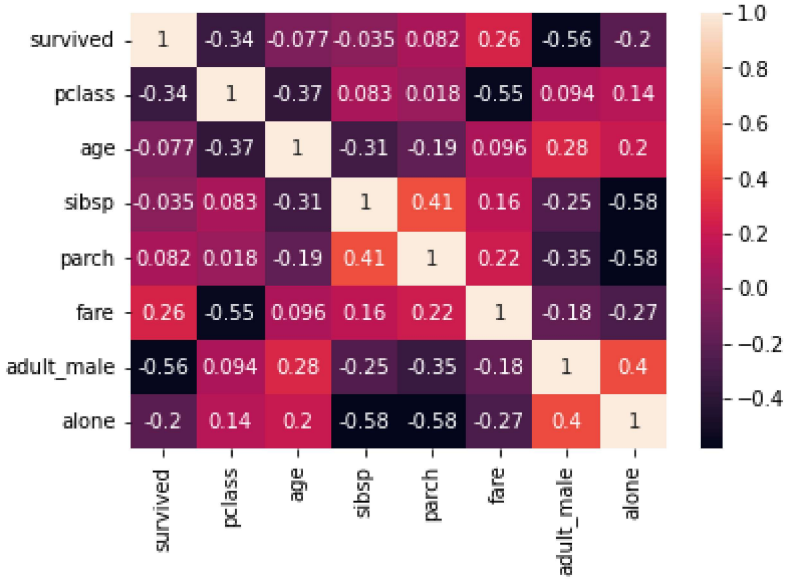
In [ ]:
```
# Step-10 Correlation
corr = df.corr(method="pearson") # you can use spearman if you want
corr
# this will display a correlation matrix
```

Out[ ]:

	survived	pclass	age	sibsp	parch	fare	adult_male	alone
survived	1.000000	-0.338481	-0.077221	-0.035322	0.081629	0.257307	-0.557080	-0.203367
pclass	-0.338481	1.000000	-0.369226	0.083081	0.018443	-0.549500	0.094035	0.135207
age	-0.077221	-0.369226	1.000000	-0.308247	-0.189119	0.096067	0.280328	0.198270
sibsp	-0.035322	0.083081	-0.308247	1.000000	0.414838	0.159651	-0.253586	-0.584471
parch	0.081629	0.018443	-0.189119	0.414838	1.000000	0.216225	-0.349943	-0.583398
fare	0.257307	-0.549500	0.096067	0.159651	0.216225	1.000000	-0.182024	-0.271832
adult_male	-0.557080	0.094035	0.280328	-0.253586	-0.349943	-0.182024	1.000000	0.404744
alone	-0.203367	0.135207	0.198270	-0.584471	-0.583398	-0.271832	0.404744	1.000000

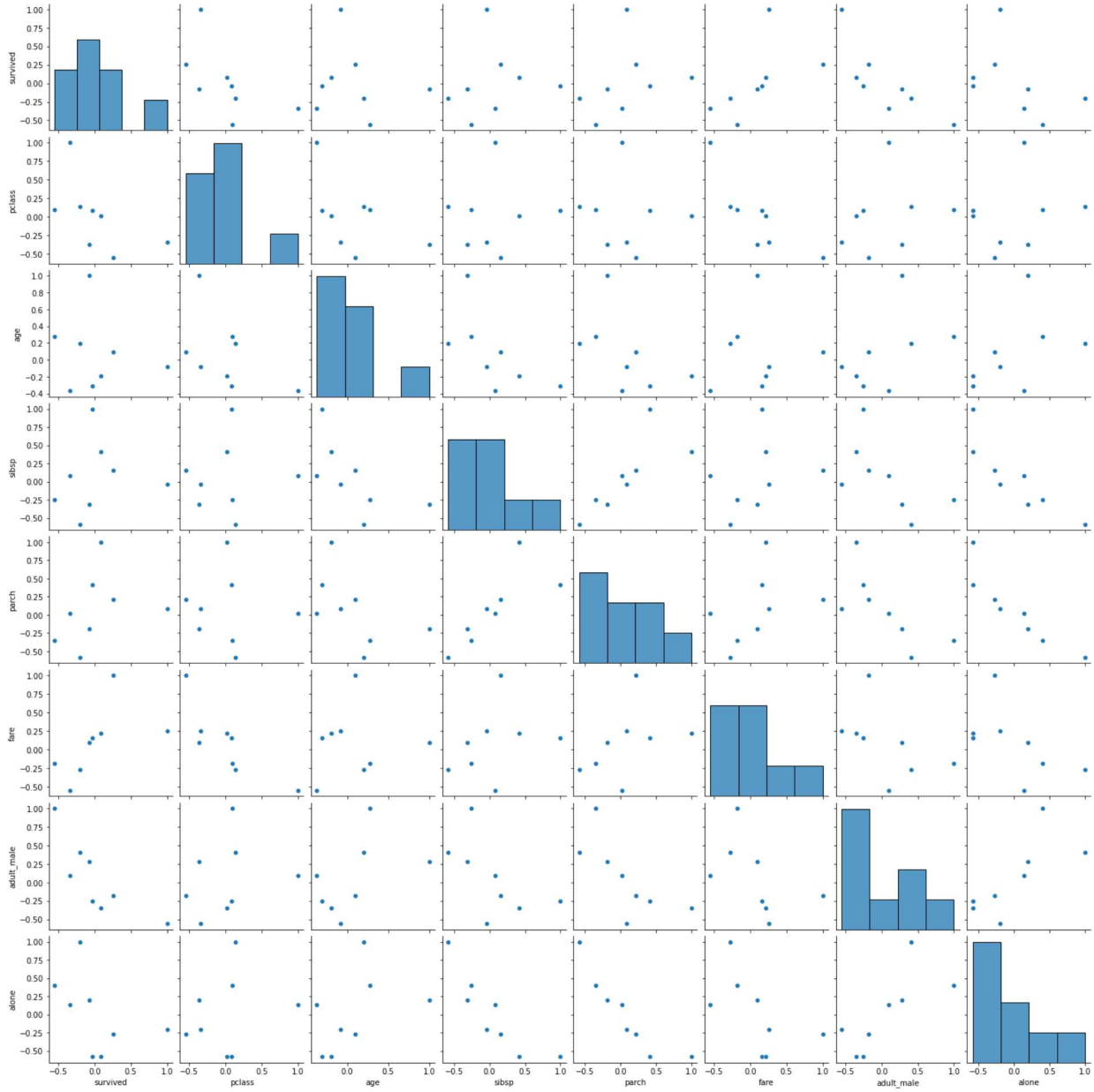
In [ ]:
```
sns.heatmap(corr, annot=True)
# this will show the numbers with colors
```

Out[ ]:
```
<AxesSubplot:>
```



In [ ]:
```
sns.pairplot(corr)
```

Out[ ]:
```
<seaborn.axisgrid.PairGrid at 0x1d9f808a7d0>
```



In [ ]:
```
corr.style.background_gradient(cmap='coolwarm')
```

Out[ ]:

	survived	pclass	age	sibsp	parch	fare	adult_male	alone
survived	1.000000	-0.338481	-0.077221	-0.035322	0.081629	0.257307	-0.557080	-0.203367
pclass	-0.338481	1.000000	-0.369226	0.083081	0.018443	-0.549500	0.094035	0.135207
age	-0.077221	-0.369226	1.000000	-0.308247	-0.189119	0.096067	0.280328	0.198270
sibsp	-0.035322	0.083081	-0.308247	1.000000	0.414838	0.159651	-0.253586	-0.584471
parch	0.081629	0.018443	-0.189119	0.414838	1.000000	0.216225	-0.349943	-0.583398
fare	0.257307	-0.549500	0.096067	0.159651	0.216225	1.000000	-0.182024	-0.271832
adult_male	-0.557080	0.094035	0.280328	-0.253586	-0.349943	-0.182024	1.000000	0.404744
alone	-0.203367	0.135207	0.198270	-0.584471	-0.583398	-0.271832	0.404744	1.000000