

EDA Case Study - titanic data set

July 16, 2022

1 titanic :

<https://www.kaggle.com/c/titanic/data> Use this to acquire domain knowledge

```
[1]: import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
[2]: df = pd.read_csv("titanic.csv", index_col = 0)
df
```

```
[2]:
```

	Survived	Pclass	\
PassengerId			
1	0	3	
2	1	1	
3	1	3	
4	1	1	
5	0	3	
...	
887	0	2	
888	1	1	
889	0	3	
890	1	1	
891	0	3	

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

891 Dooley, Mr. Patrick male 32.0

PassengerId	SibSp	Parch	Ticket	Fare	Cabin	Embarked
1	1	0	A/5 21171	7.2500	NaN	S
2	1	0	PC 17599	71.2833	C85	C
3	0	0	STON/O2. 3101282	7.9250	NaN	S
4	1	0	113803	53.1000	C123	S
5	0	0	373450	8.0500	NaN	S
...
887	0	0	211536	13.0000	NaN	S
888	0	0	112053	30.0000	B42	S
889	1	2	W./C. 6607	23.4500	NaN	S
890	0	0	111369	30.0000	C148	C
891	0	0	370376	7.7500	NaN	Q

[891 rows x 11 columns]

[3]: df[0:10]

[3]:

PassengerId	Survived	Pclass	\
1	0	3	
2	1	1	
3	1	3	
4	1	1	
5	0	3	
6	0	3	
7	0	1	
8	0	3	
9	1	3	
10	1	2	

PassengerId	Name	Sex	Age	\
1	Braund, Mr. Owen Harris	male	22.0	
2	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	
3	Heikkinen, Miss. Laina	female	26.0	
4	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	
5	Allen, Mr. William Henry	male	35.0	
6	Moran, Mr. James	male	NaN	
7	McCarthy, Mr. Timothy J	male	54.0	
8	Palsson, Master. Gosta Leonard	male	2.0	
9	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27.0	
10	Nasser, Mrs. Nicholas (Adele Achem)	female	14.0	

SibSp	Parch	Ticket	Fare	Cabin	Embarked
-------	-------	--------	------	-------	----------

PassengerId						
1	1	0	A/5 21171	7.2500	NaN	S
2	1	0	PC 17599	71.2833	C85	C
3	0	0	STON/O2. 3101282	7.9250	NaN	S
4	1	0	113803	53.1000	C123	S
5	0	0	373450	8.0500	NaN	S
6	0	0	330877	8.4583	NaN	Q
7	0	0	17463	51.8625	E46	S
8	3	1	349909	21.0750	NaN	S
9	0	2	347742	11.1333	NaN	S
10	1	0	237736	30.0708	NaN	C

```
[5]: df.head(8)
```

```
[5]:
```

	Survived	Pclass	\
PassengerId			
1	0	3	
2	1	1	
3	1	3	
4	1	1	
5	0	3	
6	0	3	
7	0	1	
8	0	3	

	Name	Sex	Age	\
PassengerId				
1	Braund, Mr. Owen Harris	male	22.0	
2	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	
3	Heikkinen, Miss. Laina	female	26.0	
4	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	
5	Allen, Mr. William Henry	male	35.0	
6	Moran, Mr. James	male	NaN	
7	McCarthy, Mr. Timothy J	male	54.0	
8	Palsson, Master. Gosta Leonard	male	2.0	

	SibSp	Parch	Ticket	Fare	Cabin	Embarked
PassengerId						
1	1	0	A/5 21171	7.2500	NaN	S
2	1	0	PC 17599	71.2833	C85	C
3	0	0	STON/O2. 3101282	7.9250	NaN	S
4	1	0	113803	53.1000	C123	S
5	0	0	373450	8.0500	NaN	S
6	0	0	330877	8.4583	NaN	Q
7	0	0	17463	51.8625	E46	S
8	3	1	349909	21.0750	NaN	S

```
[7]: df.tail(12)
```

```
[7]:
```

	Survived	Pclass	Name \
PassengerId			
880	1	1	Potter, Mrs. Thomas Jr (Lily Alexenia Wilson)
881	1	2	Shelley, Mrs. William (Imanita Parrish Hall)
882	0	3	Markun, Mr. Johann
883	0	3	Dahlberg, Miss. Gerda Ulrika
884	0	2	Banfield, Mr. Frederick James
885	0	3	Sutehall, Mr. Henry Jr
886	0	3	Rice, Mrs. William (Margaret Norton)
887	0	2	Montvila, Rev. Juozas
888	1	1	Graham, Miss. Margaret Edith
889	0	3	Johnston, Miss. Catherine Helen "Carrie"
890	1	1	Behr, Mr. Karl Howell
891	0	3	Dooley, Mr. Patrick

	Sex	Age	SibSp	Parch		Ticket	Fare	Cabin \
PassengerId								
880	female	56.0	0	1		11767	83.1583	C50
881	female	25.0	0	1		230433	26.0000	NaN
882	male	33.0	0	0		349257	7.8958	NaN
883	female	22.0	0	0		7552	10.5167	NaN
884	male	28.0	0	0	C.A./SOTON	34068	10.5000	NaN
885	male	25.0	0	0	SOTON/OQ	392076	7.0500	NaN
886	female	39.0	0	5		382652	29.1250	NaN
887	male	27.0	0	0		211536	13.0000	NaN
888	female	19.0	0	0		112053	30.0000	B42
889	female	NaN	1	2	W./C.	6607	23.4500	NaN
890	male	26.0	0	0		111369	30.0000	C148
891	male	32.0	0	0		370376	7.7500	NaN

	Embarked
PassengerId	
880	C
881	S
882	S
883	S
884	S
885	S
886	Q
887	S
888	S
889	S
890	C
891	Q

```
[48]: df.sample(15)
```

```
[48]:
```

	Survived	Pclass	Name \
PassengerId			
575	0	3	Rush, Mr. Alfred George John
418	1	2	Silven, Miss. Lyyli Karoliina
622	1	1	Kimball, Mr. Edwin Nelson Jr
550	1	2	Davies, Master. John Morgan Jr
874	0	3	Vander Cruyssen, Mr. Victor
101	0	3	Petranec, Miss. Matilda
831	1	3	Yasbeck, Mrs. Antoni (Selini Alexander)
364	0	3	Asim, Mr. Adola
570	1	3	Jonsson, Mr. Carl
689	0	3	Fischer, Mr. Eberhard Thelander
639	0	3	Panula, Mrs. Juha (Maria Emilia Ojala)
727	1	2	Renouf, Mrs. Peter Henry (Lillian Jefferys)
859	1	3	Baclini, Mrs. Solomon (Latifa Qurban)
673	0	2	Mitchell, Mr. Henry Michael
566	0	3	Davies, Mr. Alfred J

	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin \
PassengerId							
575	male	16.0	0	0	A/4. 20589	8.0500	NaN
418	female	18.0	0	2	250652	13.0000	NaN
622	male	42.0	1	0	11753	52.5542	D19
550	male	8.0	1	1	C.A. 33112	36.7500	NaN
874	male	47.0	0	0	345765	9.0000	NaN
101	female	28.0	0	0	349245	7.8958	NaN
831	female	15.0	1	0	2659	14.4542	NaN
364	male	35.0	0	0	SOTON/O.Q. 3101310	7.0500	NaN
570	male	32.0	0	0	350417	7.8542	NaN
689	male	18.0	0	0	350036	7.7958	NaN
639	female	41.0	0	5	3101295	39.6875	NaN
727	female	30.0	3	0	31027	21.0000	NaN
859	female	24.0	0	3	2666	19.2583	NaN
673	male	70.0	0	0	C.A. 24580	10.5000	NaN
566	male	24.0	2	0	A/4 48871	24.1500	NaN

```
Embarked
```

PassengerId	Embarked
575	S
418	S
622	S
550	S
874	S
101	S
831	C

364	S
570	S
689	S
639	S
727	S
859	C
673	S
566	S

```
[50]: dfp1 = df[df['Pclass'] == 1]
dfp1
```

```
[50]:
```

PassengerId	Survived	Pclass	\
2	1	1	
4	1	1	
7	0	1	
12	1	1	
24	1	1	
...	
872	1	1	
873	0	1	
880	1	1	
888	1	1	
890	1	1	

PassengerId	Name	Sex	Age	\
2	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	
4	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	
7	McCarthy, Mr. Timothy J	male	54.0	
12	Bonnell, Miss. Elizabeth	female	58.0	
24	Sloper, Mr. William Thompson	male	28.0	
...	
872	Beckwith, Mrs. Richard Leonard (Sallie Monypeny)	female	47.0	
873	Carlsson, Mr. Frans Olof	male	33.0	
880	Potter, Mrs. Thomas Jr (Lily Alexenia Wilson)	female	56.0	
888	Graham, Miss. Margaret Edith	female	19.0	
890	Behr, Mr. Karl Howell	male	26.0	

PassengerId	SibSp	Parch	Ticket	Fare	Cabin	Embarked
2	1	0	PC 17599	71.2833	C85	C
4	1	0	113803	53.1000	C123	S
7	0	0	17463	51.8625	E46	S
12	0	0	113783	26.5500	C103	S
24	0	0	113788	35.5000	A6	S

...
872	1	1	11751	52.5542		D35	S	
873	0	0	695	5.0000	B51 B53 B55		S	
880	0	1	11767	83.1583		C50	C	
888	0	0	112053	30.0000		B42	S	
890	0	0	111369	30.0000		C148	C	

[216 rows x 11 columns]

```
[51]: dfp2 = df[df['Pclass'] == 2]
dfp2
```

```
[51]:
```

Survived	Pclass	Name \	
PassengerId			
10	1	2	Nasser, Mrs. Nicholas (Adele Achem)
16	1	2	Hewlett, Mrs. (Mary D Kingcome)
18	1	2	Williams, Mr. Charles Eugene
21	0	2	Fynney, Mr. Joseph J
22	1	2	Beesley, Mr. Lawrence
...
867	1	2	Duran y More, Miss. Asuncion
875	1	2	Abelson, Mrs. Samuel (Hannah Wozosky)
881	1	2	Shelley, Mrs. William (Imanita Parrish Hall)
884	0	2	Banfield, Mr. Frederick James
887	0	2	Montvila, Rev. Juozas

Sex	Age	SibSp	Parch	Ticket	Fare	Cabin \
PassengerId						
10	female	14.0	1	0	237736	30.0708 NaN
16	female	55.0	0	0	248706	16.0000 NaN
18	male	NaN	0	0	244373	13.0000 NaN
21	male	35.0	0	0	239865	26.0000 NaN
22	male	34.0	0	0	248698	13.0000 D56
...
867	female	27.0	1	0	SC/PARIS 2149	13.8583 NaN
875	female	28.0	1	0	P/PP 3381	24.0000 NaN
881	female	25.0	0	1	230433	26.0000 NaN
884	male	28.0	0	0	C.A./SOTON 34068	10.5000 NaN
887	male	27.0	0	0	211536	13.0000 NaN

```
Embarked
```

PassengerId	Embarked
10	C
16	S
18	S
21	S
22	S

```
...
867      C
875      C
881      S
884      S
887      S
```

[184 rows x 11 columns]

```
[52]: dfp3 = df[df['Pclass'] == 3]
dfp3
```

```
[52]:
```

Survived	Pclass	Name \
PassengerId		
1	0	3 Braund, Mr. Owen Harris
3	1	3 Heikkinen, Miss. Laina
5	0	3 Allen, Mr. William Henry
6	0	3 Moran, Mr. James
8	0	3 Palsson, Master. Gosta Leonard
...
883	0	3 Dahlberg, Miss. Gerda Ulrika
885	0	3 Sutehall, Mr. Henry Jr
886	0	3 Rice, Mrs. William (Margaret Norton)
889	0	3 Johnston, Miss. Catherine Helen "Carrie"
891	0	3 Dooley, Mr. Patrick

Sex	Age	SibSp	Parch	Ticket	Fare	Cabin \
PassengerId						
1	male	22.0	1	0	A/5 21171	7.2500 NaN
3	female	26.0	0	0	STON/O2. 3101282	7.9250 NaN
5	male	35.0	0	0	373450	8.0500 NaN
6	male	NaN	0	0	330877	8.4583 NaN
8	male	2.0	3	1	349909	21.0750 NaN
...
883	female	22.0	0	0	7552	10.5167 NaN
885	male	25.0	0	0	SOTON/OQ 392076	7.0500 NaN
886	female	39.0	0	5	382652	29.1250 NaN
889	female	NaN	1	2	W./C. 6607	23.4500 NaN
891	male	32.0	0	0	370376	7.7500 NaN

```
Embarked
PassengerId
1      S
3      S
5      S
6      Q
8      S
```



```
...
883      S
885      S
886      Q
889      S
891      Q
```

[491 rows x 11 columns]

```
[53]: dfp3[dfp3['Age'] < 40]
```

```
[53]:      Survived  Pclass  \
PassengerId
1           0         3
3           1         3
5           0         3
8           0         3
9           1         3
...
882          0         3
883          0         3
885          0         3
886          0         3
891          0         3
```

```

Name      Sex  Age  \
PassengerId
1      Braund, Mr. Owen Harris    male  22.0
3      Heikkinen, Miss. Laina    female  26.0
5      Allen, Mr. William Henry    male  35.0
8      Palsson, Master. Gosta Leonard    male   2.0
9      Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)    female  27.0
...
882      Markun, Mr. Johann    male  33.0
883      Dahlberg, Miss. Gerda Ulrika    female  22.0
885      Sutehall, Mr. Henry Jr    male  25.0
886      Rice, Mrs. William (Margaret Norton)    female  39.0
891      Dooley, Mr. Patrick    male  32.0
```

```

SibSp  Parch      Ticket    Fare Cabin Embarked
PassengerId
1      1      0      A/5 21171    7.2500   NaN      S
3      0      0  STON/O2. 3101282    7.9250   NaN      S
5      0      0      373450    8.0500   NaN      S
8      3      1      349909   21.0750   NaN      S
9      0      2      347742   11.1333   NaN      S
...      ...      ...      ...      ...      ...
```

882	0	0		349257	7.8958	NaN	S
883	0	0		7552	10.5167	NaN	S
885	0	0	SOTON/OQ	392076	7.0500	NaN	S
886	0	5		382652	29.1250	NaN	Q
891	0	0		370376	7.7500	NaN	Q

[310 rows x 11 columns]

```
[64]: len(df[(df['Sex'] == 'female') & (df['Pclass'] == 3)]) / len(df[df['Pclass'] == 3]) * 100
```

[64]: 29.327902240325866

```
[63]: len(df[(df['Sex'] == 'female') & (df['Pclass'] == 1)]) / len(df[df['Pclass'] == 1]) * 100
```

[63]: 43.51851851851852

```
[65]: len(df[(df['Sex'] == 'female') & (df['Pclass'] == 2)]) / len(df[df['Pclass'] == 2]) * 100
```

[65]: 41.30434782608695

```
[62]: len(df[(df['Sex'] == 'male')]) / len(df) * 100
```

[62]: 64.75869809203144

```
[ ]: df[(df['Sex'] == 'male') & (df['Pclass'] == 3)]
```

```
[75]: len(df[(df['Pclass'] == 3) & (df['Embarked'] == 'S')]) / len(df[df['Pclass'] == 3])
```

[75]: 0.7189409368635438

```
[79]: len(df[(df['Pclass'] == 2) & (df['Embarked'] == 'S')]) / len(df[df['Pclass'] == 2])
```

[79]: 0.8913043478260869

```
[82]: len(df[(df['Pclass'] == 1) & (df['Embarked'] == 'S')]) / len(df[df['Pclass'] == 1])
```

[82]: 0.5879629629629629

```
[81]: len(df[(df['Pclass'] == 1) & (df['Embarked'] == 'C')]) / len(df[df['Pclass'] == 1])
```

```
[81]: 0.39351851851851855
```

```
[86]: len(df[(df['Pclass'] == 1) & (df['Embarked'] == 'Q')])/len(df[df['Pclass'] == 1])
```

```
[86]: 0.009259259259259259
```

```
[77]: len(df[(df['Pclass'] == 3) & (df['Embarked'] == 'C')])/len(df[df['Pclass'] == 3])
```

```
[77]: 0.13441955193482688
```

```
[76]: len(df[(df['Pclass'] == 3) & (df['Embarked'] == 'Q')])/len(df[df['Pclass'] == 3])
```

```
[76]: 0.14663951120162932
```

```
[68]: len(df[df['Pclass']==2])/len(df)*100
```

```
[68]: 20.65095398428732
```

```
[69]: len(df[df['Pclass']==1])/len(df)*100
```

```
[69]: 24.242424242424242
```

```
[87]: df.index
```

```
[87]: Int64Index([ 1, 2, 3, 4, 5, 6, 7, 8, 9, 10,
...,
882, 883, 884, 885, 886, 887, 888, 889, 890, 891],
dtype='int64', name='PassengerId', length=891)
```

```
[89]: list(df.columns)
```

```
[89]: ['Survived',
      'Pclass',
      'Name',
      'Sex',
      'Age',
      'SibSp',
      'Parch',
      'Ticket',
      'Fare',
      'Cabin',
      'Embarked']
```

```
[74]: # access column Age
df["Age"].head(15)
```

```
[74]: PassengerId
1      22.0
2      38.0
3      26.0
4      35.0
5      35.0
6       NaN
7      54.0
8       2.0
9      27.0
10     14.0
11      4.0
12     58.0
13     20.0
14     39.0
15     14.0
Name: Age, dtype: float64
```

```
[92]: # access column Age, Fare, Survived
df[["Age", "Fare", 'Survived']]
```

```
[92]:
```

	Age	Fare	Survived
PassengerId			
1	22.0	7.2500	0
2	38.0	71.2833	1
3	26.0	7.9250	1
4	35.0	53.1000	1
5	35.0	8.0500	0
...
887	27.0	13.0000	0
888	19.0	30.0000	1
889	NaN	23.4500	0
890	26.0	30.0000	1
891	32.0	7.7500	0

[891 rows x 3 columns]

```
[77]: df.Embarked.unique()      # to see the unique values in a particular column
```

```
[77]: array(['S', 'C', 'Q', nan], dtype=object)
```

```
[93]: df.SibSp.unique()      # to see the unique values in a particular column
```

```
[93]: array([1, 0, 3, 4, 2, 5, 8], dtype=int64)
```

```
[94]: df.Parch.unique()      # to see the unique values in a particular column
```

```
[94]: array([0, 1, 2, 5, 3, 4, 6], dtype=int64)
```

```
[95]: df.Fare.unique()      # to see the unique values in a particular column
```

```
[95]: array([ 7.25 , 71.2833,  7.925 , 53.1   ,  8.05  ,  8.4583,
51.8625, 21.075 , 11.1333, 30.0708, 16.7   , 26.55  ,
31.275 ,  7.8542, 16.    , 29.125 , 13.    , 18.    ,
 7.225 , 26.    ,  8.0292, 35.5   , 31.3875, 263.    ,
 7.8792,  7.8958, 27.7208, 146.5208,  7.75  , 10.5   ,
82.1708, 52.    ,  7.2292, 11.2417,  9.475 , 21.    ,
41.5792, 15.5   , 21.6792, 17.8   , 39.6875,  7.8   ,
76.7292, 61.9792, 27.75  , 46.9   , 80.    , 83.475 ,
27.9   , 15.2458,  8.1583,  8.6625, 73.5   , 14.4542,
56.4958,  7.65  , 29.    , 12.475 ,  9.    ,  9.5   ,
 7.7875, 47.1   , 15.85  , 34.375 , 61.175 , 20.575 ,
34.6542, 63.3583, 23.    , 77.2875,  8.6542,  7.775 ,
24.15  ,  9.825 , 14.4583, 247.5208,  7.1417, 22.3583,
 6.975 ,  7.05  , 14.5   , 15.0458, 26.2833,  9.2167,
79.2   ,  6.75  , 11.5   , 36.75  ,  7.7958, 12.525 ,
66.6   ,  7.3125, 61.3792,  7.7333, 69.55  , 16.1   ,
15.75  , 20.525 , 55.    , 25.925 , 33.5   , 30.6958,
25.4667, 28.7125,  0.    , 15.05  , 39.    , 22.025 ,
50.    ,  8.4042,  6.4958, 10.4625, 18.7875, 31.    ,
113.275 , 27.    , 76.2917, 90.    ,  9.35  , 13.5   ,
 7.55  , 26.25  , 12.275 ,  7.125 , 52.5542, 20.2125,
86.5   , 512.3292, 79.65  , 153.4625, 135.6333, 19.5   ,
29.7   , 77.9583, 20.25  , 78.85  , 91.0792, 12.875 ,
 8.85  , 151.55  , 30.5   , 23.25  , 12.35  , 110.8833,
108.9  , 24.    , 56.9292, 83.1583, 262.375 , 14.    ,
164.8667, 134.5  ,  6.2375, 57.9792, 28.5   , 133.65  ,
15.9   ,  9.225 , 35.    , 75.25  , 69.3   , 55.4417,
211.5  ,  4.0125, 227.525 , 15.7417,  7.7292, 12.    ,
120.   , 12.65  , 18.75  ,  6.8583, 32.5   ,  7.875 ,
14.4   , 55.9   ,  8.1125, 81.8583, 19.2583, 19.9667,
89.1042, 38.5   ,  7.725 , 13.7917,  9.8375,  7.0458,
 7.5208, 12.2875,  9.5875, 49.5042, 78.2667, 15.1   ,
 7.6292, 22.525 , 26.2875, 59.4   ,  7.4958, 34.0208,
93.5   , 221.7792, 106.425 , 49.5   , 71.    , 13.8625,
 7.8292, 39.6   , 17.4   , 51.4792, 26.3875, 30.    ,
40.125 ,  8.7125, 15.    , 33.    , 42.4   , 15.55  ,
65.    , 32.3208,  7.0542,  8.4333, 25.5875,  9.8417,
 8.1375, 10.1708, 211.3375, 57.    , 13.4167,  7.7417,
 9.4833,  7.7375,  8.3625, 23.45  , 25.9292,  8.6833,
 8.5167,  7.8875, 37.0042,  6.45  ,  6.95  ,  8.3   ,
 6.4375, 39.4   , 14.1083, 13.8583, 50.4958,  5.    ,
```

```
9.8458, 10.5167])
```

```
[96]: df[df['Embarked'].isnull()]
```

```
[96]:
```

	Survived	Pclass	Name \
PassengerId			
62	1	1	Icard, Miss. Amelie
830	1	1	Stone, Mrs. George Nelson (Martha Evelyn)

	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
PassengerId								
62	female	38.0	0	0	113572	80.0	B28	NaN
830	female	62.0	0	0	113572	80.0	B28	NaN

```
[98]: df[df['Ticket'] == '113572']
```

```
[98]:
```

	Survived	Pclass	Name \
PassengerId			
62	1	1	Icard, Miss. Amelie
830	1	1	Stone, Mrs. George Nelson (Martha Evelyn)

	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
PassengerId								
62	female	38.0	0	0	113572	80.0	B28	NaN
830	female	62.0	0	0	113572	80.0	B28	NaN

```
[99]: df[df['Cabin'] == 'B28']
```

```
[99]:
```

	Survived	Pclass	Name \
PassengerId			
62	1	1	Icard, Miss. Amelie
830	1	1	Stone, Mrs. George Nelson (Martha Evelyn)

	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
PassengerId								
62	female	38.0	0	0	113572	80.0	B28	NaN
830	female	62.0	0	0	113572	80.0	B28	NaN

```
[101]: df.Embarked
```

```
[101]: PassengerId
1      S
2      C
3      S
4      S
5      S
..
```

```

887    S
888    S
889    S
890    C
891    Q
Name: Embarked, Length: 891, dtype: object

```

```
[52]: df.Parch.unique()
```

```
[52]: array([0, 1, 2, 5, 3, 4, 6], dtype=int64)
```

```
[47]: df['SibSp'].unique()      #df.SibSp.unique()
```

```
[47]: array([1, 0, 3, 4, 2, 5, 8], dtype=int64)
```

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

```
[21]:
```

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

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

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

```
[69]: df[3:17]
```

```
[69]:
```

	Survived	Pclass	\
PassengerId			
4	1	1	
5	0	3	

6	0	3
7	0	1
8	0	3
9	1	3
10	1	2
11	1	3
12	1	1
13	0	3
14	0	3
15	0	3
16	1	2
17	0	3

PassengerId	Name	Sex	Age \
4	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0
5	Allen, Mr. William Henry	male	35.0
6	Moran, Mr. James	male	NaN
7	McCarthy, Mr. Timothy J	male	54.0
8	Palsson, Master. Gosta Leonard	male	2.0
9	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27.0
10	Nasser, Mrs. Nicholas (Adele Achem)	female	14.0
11	Sandstrom, Miss. Marguerite Rut	female	4.0
12	Bonnell, Miss. Elizabeth	female	58.0
13	Saunderscock, Mr. William Henry	male	20.0
14	Andersson, Mr. Anders Johan	male	39.0
15	Vestrom, Miss. Hulda Amanda Adolfina	female	14.0
16	Hewlett, Mrs. (Mary D Kingcome)	female	55.0
17	Rice, Master. Eugene	male	2.0

PassengerId	SibSp	Parch	Ticket	Fare	Cabin	Embarked
4	1	0	113803	53.1000	C123	S
5	0	0	373450	8.0500	NaN	S
6	0	0	330877	8.4583	NaN	Q
7	0	0	17463	51.8625	E46	S
8	3	1	349909	21.0750	NaN	S
9	0	2	347742	11.1333	NaN	S
10	1	0	237736	30.0708	NaN	C
11	1	1	PP 9549	16.7000	G6	S
12	0	0	113783	26.5500	C103	S
13	0	0	A/5. 2151	8.0500	NaN	S
14	1	5	347082	31.2750	NaN	S
15	0	0	350406	7.8542	NaN	S
16	0	0	248706	16.0000	NaN	S
17	4	1	382652	29.1250	NaN	Q


```
[95]: df.loc[4:16, "Fare"].head(10)      # df['Fare'].loc[4:16].head(10)
```

```
[95]: PassengerId
4      53.1000
5       8.0500
6       8.4583
7      51.8625
8      21.0750
9      11.1333
10     30.0708
11     16.7000
12     26.5500
13      8.0500
Name: Fare, dtype: float64
```

```
[22]: df.loc[:, "Fare"].head(2)          # df['Fare'].head(2)
```

```
[22]: PassengerId
1       7.2500
2      71.2833
Name: Fare, dtype: float64
```

```
[24]: df.loc[10:20, ["Fare", "Name"]]    # df[["Fare", "Name"]].loc[10:20]
```

```
[24]:
```

	Fare	Name
PassengerId		
10	30.0708	Nasser, Mrs. Nicholas (Adele Achem)
11	16.7000	Sandstrom, Miss. Marguerite Rut
12	26.5500	Bonnell, Miss. Elizabeth
13	8.0500	Saundercock, Mr. William Henry
14	31.2750	Andersson, Mr. Anders Johan
15	7.8542	Vestrom, Miss. Hulda Amanda Adolfina
16	16.0000	Hewlett, Mrs. (Mary D Kingcome)
17	29.1250	Rice, Master. Eugene
18	13.0000	Williams, Mr. Charles Eugene
19	18.0000	Vander Planke, Mrs. Julius (Emelia Maria Vande...
20	7.2250	Masselmani, Mrs. Fatima

```
[29]: df.loc[3 : 5, ["Fare", "Name"]].head(2)
```

```
[29]:
```

	Fare	Name
PassengerId		
3	7.925	Heikkinen, Miss. Laina
4	53.100	Futrelle, Mrs. Jacques Heath (Lily May Peel)

```
[109]: df
```

```
[109]:
```

	Survived	Pclass	\
PassengerId			
1	0	3	
2	1	1	
3	1	3	
4	1	1	
5	0	3	
...	
887	0	2	
888	1	1	
889	0	3	
890	1	1	
891	0	3	

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

	SibSp	Parch	Ticket	Fare	Cabin	Embarked
PassengerId						
1	1	0	A/5 21171	7.2500	NaN	S
2	1	0	PC 17599	71.2833	C85	C
3	0	0	STON/O2. 3101282	7.9250	NaN	S
4	1	0	113803	53.1000	C123	S
5	0	0	373450	8.0500	NaN	S
...
887	0	0	211536	13.0000	NaN	S
888	0	0	112053	30.0000	B42	S
889	1	2	W./C. 6607	23.4500	NaN	S
890	0	0	111369	30.0000	C148	C
891	0	0	370376	7.7500	NaN	Q

[891 rows x 11 columns]

```
[103]: df1 = df[(df['Survived'] == 1) & (df['Parch'] == 1) & (df['Pclass'] == 1)]
df1
```

```
[103]:
```

PassengerId	Survived	Pclass	\
2	1	1	
4	1	1	
12	1	1	
24	1	1	
32	1	1	
...	
863	1	1	
872	1	1	
880	1	1	
888	1	1	
890	1	1	

PassengerId	Name	Sex	Age	\
2	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	
4	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	
12	Bonnell, Miss. Elizabeth	female	58.0	
24	Sloper, Mr. William Thompson	male	28.0	
32	Spencer, Mrs. William Augustus (Marie Eugenie)	female	NaN	
...	
863	Swift, Mrs. Frederick Joel (Margaret Welles Ba...	female	48.0	
872	Beckwith, Mrs. Richard Leonard (Sallie Monypeny)	female	47.0	
880	Potter, Mrs. Thomas Jr (Lily Alexenia Wilson)	female	56.0	
888	Graham, Miss. Margaret Edith	female	19.0	
890	Behr, Mr. Karl Howell	male	26.0	

PassengerId	SibSp	Parch	Ticket	Fare	Cabin	Embarked
2	1	0	PC 17599	71.2833	C85	C
4	1	0	113803	53.1000	C123	S
12	0	0	113783	26.5500	C103	S
24	0	0	113788	35.5000	A6	S
32	1	0	PC 17569	146.5208	B78	C
...
863	0	0	17466	25.9292	D17	S
872	1	1	11751	52.5542	D35	S
880	0	1	11767	83.1583	C50	C
888	0	0	112053	30.0000	B42	S
890	0	0	111369	30.0000	C148	C

[136 rows x 11 columns]

```
[108]: pd.set_option('display.max_rows', 10)
```

```
[132]: df['SibSp'].unique()
```

```
[132]: array([1, 0, 3, 4, 2, 5, 8], dtype=int64)
```

```
[136]: len(df.columns)
```

```
[136]: 11
```

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

```
[106]:
```

	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

```
[104]: df['Age'].mean()
```

```
[104]: 29.69911764705882
```

```
[105]: len(df[df['Age'] <= df['Age'].mean()])/len(df)
```

```
[105]: 0.43097643097643096
```

```
[3]: # query for people of age 30 - 40
df[(df.Age > 30) & (df.Age < 40)] # Apply sample() to analyse random data
↳ records
```

```
[3]:
```

PassengerId	Survived	Pclass	\
2	1	1	
4	1	1	
5	0	3	
14	0	3	
19	0	3	
...	
868	0	1	
873	0	1	
882	0	3	
886	0	3	
891	0	3	

PassengerId	Name	Sex	Age	\
2	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	

4	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0
5	Allen, Mr. William Henry	male	35.0
14	Andersson, Mr. Anders Johan	male	39.0
19	Vander Planke, Mrs. Julius (Emelia Maria Vande...	female	31.0
...
868	Roebeling, Mr. Washington Augustus II	male	31.0
873	Carlsson, Mr. Frans Olof	male	33.0
882	Markun, Mr. Johann	male	33.0
886	Rice, Mrs. William (Margaret Norton)	female	39.0
891	Dooley, Mr. Patrick	male	32.0

PassengerId	SibSp	Parch	Ticket	Fare	Cabin	Embarked
2	1	0	PC 17599	71.2833	C85	C
4	1	0	113803	53.1000	C123	S
5	0	0	373450	8.0500	NaN	S
14	1	5	347082	31.2750	NaN	S
19	1	0	345763	18.0000	NaN	S
...
868	0	0	PC 17590	50.4958	A24	S
873	0	0	695	5.0000	B51 B53 B55	S
882	0	0	349257	7.8958	NaN	S
886	0	5	382652	29.1250	NaN	Q
891	0	0	370376	7.7500	NaN	Q

[142 rows x 11 columns]

```
[9]: # query for people of age 30 - 40
df[(df.Age > 30) & (df.Age < 40)]['Age'].mean() # Apply sample() to analyse
↳ random data records
```

[9]: 34.598591549295776

```
[107]: df[(df.Sex == 'male') & (df.Survived == 1)]
```

```
[107]:
```

PassengerId	Survived	Pclass	Name	Sex	Age	\
18	1	2	Williams, Mr. Charles Eugene	male	NaN	
22	1	2	Beesley, Mr. Lawrence	male	34.0	
24	1	1	Sloper, Mr. William Thompson	male	28.0	
37	1	3	Mamee, Mr. Hanna	male	NaN	
56	1	1	Woolner, Mr. Hugh	male	NaN	
...	
839	1	3	Chip, Mr. Chang	male	32.0	
840	1	1	Marechal, Mr. Pierre	male	NaN	
858	1	1	Daly, Mr. Peter Denis	male	51.0	
870	1	3	Johnson, Master. Harold Theodor	male	4.0	

890	1	1	Behr, Mr. Karl Howell	male	26.0
-----	---	---	-----------------------	------	------

PassengerId	SibSp	Parch	Ticket	Fare	Cabin	Embarked
18	0	0	244373	13.0000	NaN	S
22	0	0	248698	13.0000	D56	S
24	0	0	113788	35.5000	A6	S
37	0	0	2677	7.2292	NaN	C
56	0	0	19947	35.5000	C52	S
...
839	0	0	1601	56.4958	NaN	S
840	0	0	11774	29.7000	C47	C
858	0	0	113055	26.5500	E17	S
870	1	1	347742	11.1333	NaN	S
890	0	0	111369	30.0000	C148	C

[109 rows x 11 columns]

```
[11]: df[(df.Sex == 'female') & (df.Survived == 1)]
```

```
[11]:
```

PassengerId	Survived	Pclass	\
2	1	1	
3	1	3	
4	1	1	
9	1	3	
10	1	2	
...	
875	1	2	
876	1	3	
880	1	1	
881	1	2	
888	1	1	

PassengerId	Name	Sex	Age	\
2	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	
3	Heikkinen, Miss. Laina	female	26.0	
4	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	
9	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27.0	
10	Nasser, Mrs. Nicholas (Adele Achem)	female	14.0	
...	
875	Abelson, Mrs. Samuel (Hannah Wizosky)	female	28.0	
876	Najib, Miss. Adele Kiamie "Jane"	female	15.0	
880	Potter, Mrs. Thomas Jr (Lily Alexenia Wilson)	female	56.0	
881	Shelley, Mrs. William (Imanita Parrish Hall)	female	25.0	
888	Graham, Miss. Margaret Edith	female	19.0	

PassengerId	SibSp	Parch	Ticket	Fare	Cabin	Embarked
2	1	0	PC 17599	71.2833	C85	C
3	0	0	STON/O2. 3101282	7.9250	NaN	S
4	1	0	113803	53.1000	C123	S
9	0	2	347742	11.1333	NaN	S
10	1	0	237736	30.0708	NaN	C
...
875	1	0	P/PP 3381	24.0000	NaN	C
876	0	0	2667	7.2250	NaN	C
880	0	1	11767	83.1583	C50	C
881	0	1	230433	26.0000	NaN	S
888	0	0	112053	30.0000	B42	S

[233 rows x 11 columns]

```
[111]: df[(df.Sex == 'male')].count()
```

```
[111]: Survived    577
Pclass          577
Name            577
Sex             577
Age            453
SibSp           577
Parch           577
Ticket          577
Fare            577
Cabin           107
Embarked        577
dtype: int64
```

```
[162]: df[(df.Sex == 'female')].count()
```

```
[162]: Survived    314
Pclass          314
Name            314
Sex             314
Age            261
SibSp           314
Parch           314
Ticket          314
Fare            314
Cabin            97
Embarked        312
dtype: int64
```

```
[164]: df[(df.Sex == 'male') & (df.Survived == 1)].count()
```

```
[164]: Survived      109  
      Pclass       109  
      Name         109  
      Sex          109  
      Age          93  
      SibSp        109  
      Parch        109  
      Ticket       109  
      Fare         109  
      Cabin        45  
      Embarked     109  
      dtype: int64
```

```
[163]: df[(df.Sex == 'female') & (df.Survived == 1)].count()
```

```
[163]: Survived      233  
      Pclass       233  
      Name         233  
      Sex          233  
      Age         197  
      SibSp        233  
      Parch        233  
      Ticket       233  
      Fare         233  
      Cabin        91  
      Embarked     231  
      dtype: int64
```

```
[12]: len(df[(df.Sex == 'male') & (df.Survived == 1)]) / len(df[(df.Sex == 'male')])
```

```
[12]: 0.18890814558058924
```

```
[13]: len(df[(df.Sex == 'female') & (df.Survived == 1)]) / len(df[(df.Sex == 'female')])
```

```
[13]: 0.7420382165605095
```

```
[115]: # dim of df ??  
      df.shape
```

```
[115]: (891, 11)
```

```
[116]: df.Fare.mean()
```

```
[116]: 32.204207968574636
```



```
[117]: df[(df.Fare < df.Fare.mean())] # df[df.Fare == df.Fare.median()]
```

```
[117]:
```

	Survived	Pclass	Name \
PassengerId			
1	0	3	Braund, Mr. Owen Harris
3	1	3	Heikkinen, Miss. Laina
5	0	3	Allen, Mr. William Henry
6	0	3	Moran, Mr. James
8	0	3	Palsson, Master. Gosta Leonard
...
887	0	2	Montvila, Rev. Juozas
888	1	1	Graham, Miss. Margaret Edith
889	0	3	Johnston, Miss. Catherine Helen "Carrie"
890	1	1	Behr, Mr. Karl Howell
891	0	3	Dooley, Mr. Patrick

	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin \
PassengerId							
1	male	22.0	1	0	A/5 21171	7.2500	NaN
3	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN
5	male	35.0	0	0	373450	8.0500	NaN
6	male	NaN	0	0	330877	8.4583	NaN
8	male	2.0	3	1	349909	21.0750	NaN
...
887	male	27.0	0	0	211536	13.0000	NaN
888	female	19.0	0	0	112053	30.0000	B42
889	female	NaN	1	2	W./C. 6607	23.4500	NaN
890	male	26.0	0	0	111369	30.0000	C148
891	male	32.0	0	0	370376	7.7500	NaN

	Embarked
PassengerId	
1	S
3	S
5	S
6	Q
8	S
...	...
887	S
888	S
889	S
890	C
891	Q

[680 rows x 11 columns]

```
[118]: df[(df.Fare < df.Fare.mean()) & (df.Survived == 1)] # df[df.Fare_
      ↪ == df.Fare.median()]
```

```
[118]: Survived Pclass \
```

```
PassengerId
```

```
3          1          3
9          1          3
10         1          2
11         1          3
12         1          1
...
875        1          2
876        1          3
881        1          2
888        1          1
890        1          1
```

```
Name Sex Age \
```

```
PassengerId
```

```
3          Heikkinen, Miss. Laina female 26.0
9      Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg) female 27.0
10          Nasser, Mrs. Nicholas (Adele Achem) female 14.0
11          Sandstrom, Miss. Marguerite Rut female 4.0
12          Bonnell, Miss. Elizabeth female 58.0
...
875          Abelson, Mrs. Samuel (Hannah Witosky) female 28.0
876          Najib, Miss. Adele Kiamie "Jane" female 15.0
881      Shelley, Mrs. William (Imanita Parrish Hall) female 25.0
888          Graham, Miss. Margaret Edith female 19.0
890          Behr, Mr. Karl Howell male 26.0
```

```
SibSp Parch Ticket Fare Cabin Embarked
```

```
PassengerId
```

```
3          0          0 STON/O2. 3101282 7.9250 NaN S
9          0          2          347742 11.1333 NaN S
10         1          0          237736 30.0708 NaN C
11         1          1          PP 9549 16.7000 G6 S
12         0          0          113783 26.5500 C103 S
...
875        1          0          P/PP 3381 24.0000 NaN C
876        0          0          2667 7.2250 NaN C
881        0          1          230433 26.0000 NaN S
888        0          0          112053 30.0000 B42 S
890        0          0          111369 30.0000 C148 C
```

```
[216 rows x 11 columns]
```

```
[119]: len(df[(df.Fare < df.Fare.mean()) & (df.Survived == 1)]) / len(df[(df.Fare < df.
      ↪ Fare.mean())])
```

```
[119]: 0.3176470588235294
```

```
[120]: len(df[(df.Fare >= df.Fare.mean()) & (df.Survived == 1)]) / len(df[(df.Fare >= df.
      ↪ Fare.mean())])
```

```
[120]: 0.5971563981042654
```

```
[20]: r = df[df.Fare < 32]                                # df[df.Fare == df.Fare.median()]
      r
```

```
[20]:
```

	Survived	Pclass	Name \
PassengerId			
1	0	3	Braund, Mr. Owen Harris
3	1	3	Heikkinen, Miss. Laina
5	0	3	Allen, Mr. William Henry
6	0	3	Moran, Mr. James
8	0	3	Palsson, Master. Gosta Leonard
...
887	0	2	Montvila, Rev. Juozas
888	1	1	Graham, Miss. Margaret Edith
889	0	3	Johnston, Miss. Catherine Helen "Carrie"
890	1	1	Behr, Mr. Karl Howell
891	0	3	Dooley, Mr. Patrick

	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin \
PassengerId							
1	male	22.0	1	0	A/5 21171	7.2500	NaN
3	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN
5	male	35.0	0	0	373450	8.0500	NaN
6	male	NaN	0	0	330877	8.4583	NaN
8	male	2.0	3	1	349909	21.0750	NaN
...
887	male	27.0	0	0	211536	13.0000	NaN
888	female	19.0	0	0	112053	30.0000	B42
889	female	NaN	1	2	W./C. 6607	23.4500	NaN
890	male	26.0	0	0	111369	30.0000	C148
891	male	32.0	0	0	370376	7.7500	NaN

```
Embarked
```

PassengerId	
1	S
3	S
5	S
6	Q

```

8          S
...
887        S
888        S
889        S
890        C
891        Q

```

[680 rows x 11 columns]

```
[92]: r[r['Survived'] == 1]
```

```
[92]:
      Survived  Pclass  \
PassengerId
3             1        3
9             1        3
10            1        2
11            1        3
12            1        1
...
875           1        2
876           1        3
881           1        2
888           1        1
890           1        1

```

```

                                     Name      Sex  Age  \
PassengerId
3                                Heikkinen, Miss. Laina  female  26.0
9      Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)  female  27.0
10                                Nasser, Mrs. Nicholas (Adele Achem)  female  14.0
11                                Sandstrom, Miss. Marguerite Rut  female   4.0
12                                Bonnell, Miss. Elizabeth  female  58.0
...
875                                Abelson, Mrs. Samuel (Hannah Wozosky)  female  28.0
876                                Najib, Miss. Adele Kiamie "Jane"  female  15.0
881      Shelley, Mrs. William (Imanita Parrish Hall)  female  25.0
888                                Graham, Miss. Margaret Edith  female  19.0
890                                Behr, Mr. Karl Howell    male   26.0

```

```

      SibSp  Parch      Ticket    Fare Cabin Embarked
PassengerId
3          0      0  STON/O2. 3101282    7.9250   NaN      S
9          0      2        347742   11.1333   NaN      S
10         1      0        237736   30.0708   NaN      C
11         1      1         PP 9549   16.7000    G6      S
12         0      0        113783   26.5500  C103      S

```

...	
875		1	0	P/PP	3381	24.0000	NaN	C
876		0	0		2667	7.2250	NaN	C
881		0	1		230433	26.0000	NaN	S
888		0	0		112053	30.0000	B42	S
890		0	0		111369	30.0000	C148	C

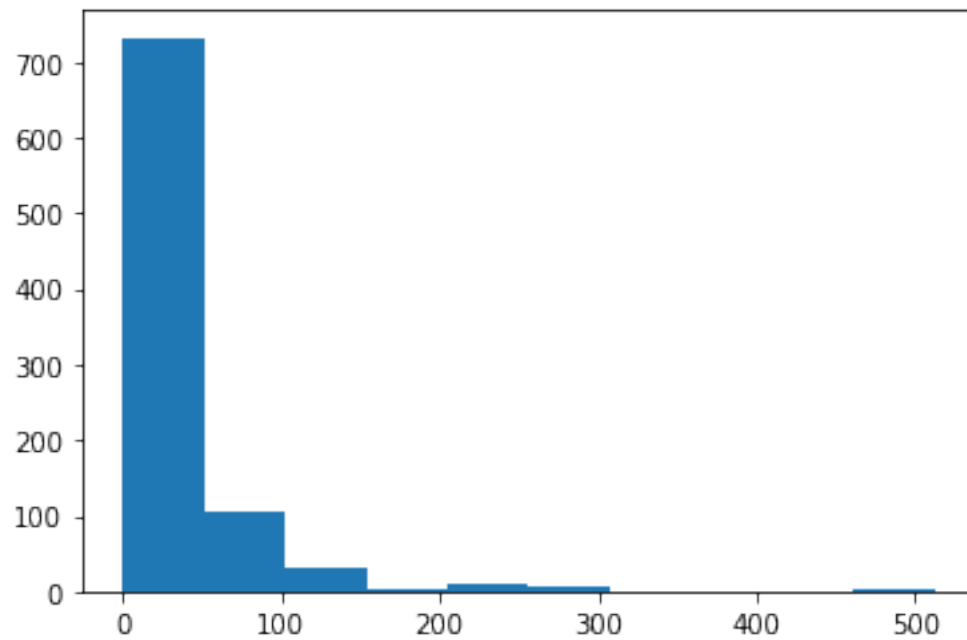
[216 rows x 11 columns]

```
[76]: df.Fare
```

```
[76]: PassengerId
1      7.2500
2     71.2833
3      7.9250
4     53.1000
5      8.0500
...
887    13.0000
888    30.0000
889    23.4500
890    30.0000
891     7.7500
Name: Fare, Length: 891, dtype: float64
```

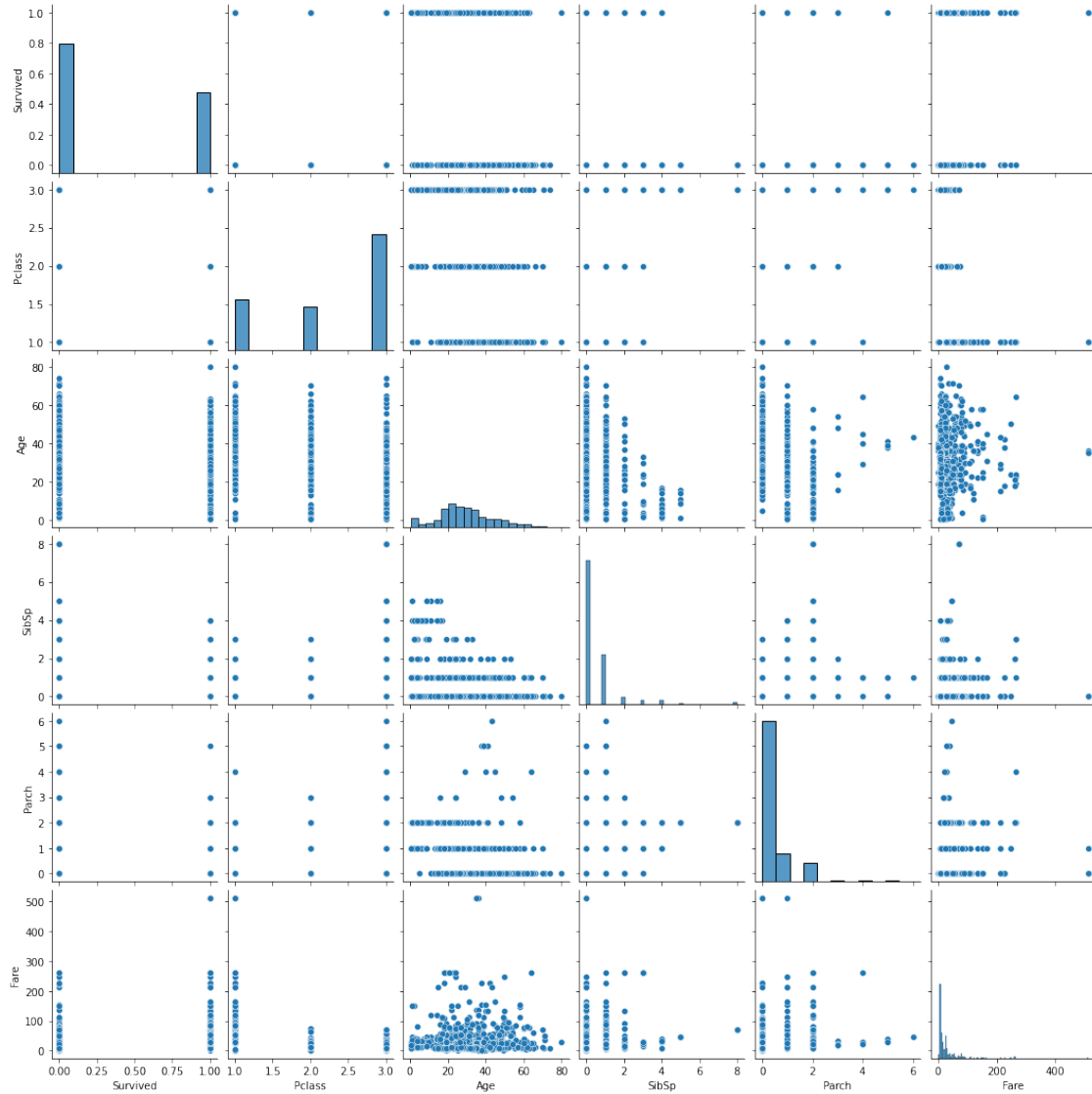
```
[32]: plt.hist(df.Fare)
```

```
[32]: (array([732., 106., 31., 2., 11., 6., 0., 0., 0., 3.]),
array([ 0., 51.23292, 102.46584, 153.69876, 204.93168, 256.1646 ,
307.39752, 358.63044, 409.86336, 461.09628, 512.3292 ]),
<BarContainer object of 10 artists>)
```



```
[22]: sns.pairplot(df[[]])
```

```
[22]: <seaborn.axisgrid.PairGrid at 0x20c463b9a90>
```



```
[ ]:
```

```
[38]: df.columns
```

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

```
[39]: df.Embarked.unique()
```

```
[39]: array(['S', 'C', 'Q', nan], dtype=object)
```

```
[45]: df['Sex'].unique()
```

```
[45]: array(['male', 'female'], dtype=object)
```

```
[41]: df.Pclass.unique()
```

```
[41]: array([3, 1, 2], dtype=int64)
```

```
[43]: for i in df.columns :  
      try:  
          print(i, " = \n", df[i].max())  
      except:  
          pass
```

```
Survived =  
1  
Pclass =  
3  
Name =  
van Melkebeke, Mr. Philemon  
Sex =  
male  
Age =  
80.0  
SibSp =  
8  
Parch =  
6  
Ticket =  
WE/P 5735  
Fare =  
512.3292
```

```
[44]: # only datatypes of each col  
  
df.dtypes
```

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



```
[40]: # datatypes, rows, cols, size
```

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 891 entries, 1 to 891
Data columns (total 11 columns):
 #   Column      Non-Null Count  Dtype
---  -
 0   Survived    891 non-null    int64
 1   Pclass      891 non-null    int64
 2   Name        891 non-null    object
 3   Sex         891 non-null    object
 4   Age         714 non-null    float64
 5   SibSp       891 non-null    int64
 6   Parch       891 non-null    int64
 7   Ticket      891 non-null    object
 8   Fare        891 non-null    float64
 9   Cabin       204 non-null    object
10   Embarked    889 non-null    object
dtypes: float64(2), int64(4), object(5)
memory usage: 115.8+ KB
```

```
[24]: df.isnull().sum()
```

```
[24]: Survived      0
      Pclass       0
      Name         0
      Sex          0
      Age         177
      SibSp        0
      Parch        0
      Ticket       0
      Fare         0
      Cabin       687
      Embarked     2
      dtype: int64
```

```
[50]: (df.isnull().sum() / len(df)) * 100
```

```
[50]: Survived      0.000000
      Pclass       0.000000
      Name         0.000000
      Sex          0.000000
      Age         19.865320
      SibSp        0.000000
      Parch        0.000000
```

```
Ticket      0.000000
Fare        0.000000
Cabin       77.104377
Embarked    0.224467
dtype: float64
```

```
[49]: df.count()
```

```
[49]: Survived      891
Pclass           891
Name             891
Sex              891
Age             714
SibSp           891
Parch           891
Ticket          891
Fare            891
Cabin           204
Embarked        889
dtype: int64
```

```
[51]: df.dtypes
```

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

```
[44]: df.dtypes == "object"
```

```
[44]: Survived      False
Pclass           False
Name             True
Sex              True
Age             False
SibSp           False
Parch           False
Ticket          True
Fare            False
```

```
Cabin      True
Embarked   True
dtype: bool
```

```
[45]: df.dtypes[df.dtypes == 'object']
```

```
[45]: Name      object
      Sex      object
      Ticket   object
      Cabin    object
      Embarked  object
      dtype: object
```

```
[48]: df.dtypes[df.dtypes == 'object'].index
```

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

```
[49]: df_cat = df[df.dtypes[df.dtypes == 'object'].index]
      df_cat.head()
```

```
[49]:
```

	Name	Sex	\
PassengerId			
1	Braund, Mr. Owen Harris	male	
2	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	
3	Heikkinen, Miss. Laina	female	
4	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	
5	Allen, Mr. William Henry	male	

	Ticket	Cabin	Embarked
PassengerId			
1	A/5 21171	NaN	S
2	PC 17599	C85	C
3	STON/O2. 3101282	NaN	S
4	113803	C123	S
5	373450	NaN	S

```
[50]: for i in df_cat.columns :
      print(i, " = \n", df[i].unique())
```

```
Name =
['Braund, Mr. Owen Harris'
 'Cumings, Mrs. John Bradley (Florence Briggs Thayer)'
 'Heikkinen, Miss. Laina' 'Futrelle, Mrs. Jacques Heath (Lily May Peel)'
 'Allen, Mr. William Henry' 'Moran, Mr. James' 'McCarthy, Mr. Timothy J'
 'Palsson, Master. Gosta Leonard'
 'Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)'
 'Nasser, Mrs. Nicholas (Adele Achem)' 'Sandstrom, Miss. Marguerite Rut'
```

'Bonnell, Miss. Elizabeth' 'Saundercock, Mr. William Henry'
 'Andersson, Mr. Anders Johan' 'Vestrom, Miss. Hulda Amanda Adolfina'
 'Hewlett, Mrs. (Mary D Kingcome)' 'Rice, Master. Eugene'
 'Williams, Mr. Charles Eugene'
 'Vander Planke, Mrs. Julius (Emelia Maria Vandemoortele)'
 'Masselmani, Mrs. Fatima' 'Fynney, Mr. Joseph J' 'Beesley, Mr. Lawrence'
 'McGowan, Miss. Anna "Annie"' 'Sloper, Mr. William Thompson'
 'Palsson, Miss. Torborg Danira'
 'Asplund, Mrs. Carl Oscar (Selma Augusta Emilia Johansson)'
 'Emir, Mr. Farred Chehab' 'Fortune, Mr. Charles Alexander'
 'O'Dwyer, Miss. Ellen "Nellie"' 'Todoroff, Mr. Lalio'
 'Uruchurtu, Don. Manuel E'
 'Spencer, Mrs. William Augustus (Marie Eugenie)'
 'Glynn, Miss. Mary Agatha' 'Wheadon, Mr. Edward H'
 'Meyer, Mr. Edgar Joseph' 'Holverson, Mr. Alexander Oskar'
 'Mamee, Mr. Hanna' 'Cann, Mr. Ernest Charles'
 'Vander Planke, Miss. Augusta Maria' 'Nicola-Yarred, Miss. Jamila'
 'Ahlin, Mrs. Johan (Johanna Persdotter Larsson)'
 'Turpin, Mrs. William John Robert (Dorothy Ann Wonnacott)'
 'Kraeff, Mr. Theodor' 'Laroche, Miss. Simonne Marie Anne Andree'
 'Devaney, Miss. Margaret Delia' 'Rogers, Mr. William John'
 'Lennon, Mr. Denis' 'O'Driscoll, Miss. Bridget' 'Samaan, Mr. Youssef'
 'Arnold-Franchi, Mrs. Josef (Josefine Franchi)'
 'Panula, Master. Juha Niilo' 'Nosworthy, Mr. Richard Cater'
 'Harper, Mrs. Henry Sleeper (Myna Haxtun)'
 'Faunthorpe, Mrs. Lizzie (Elizabeth Anne Wilkinson)'
 'Ostby, Mr. Engelhart Cornelius' 'Woolner, Mr. Hugh' 'Rugg, Miss. Emily'
 'Novel, Mr. Mansouer' 'West, Miss. Constance Mirium'
 'Goodwin, Master. William Frederick' 'Sirayanian, Mr. Orsen'
 'Icard, Miss. Amelie' 'Harris, Mr. Henry Birkhardt'
 'Skoog, Master. Harald' 'Stewart, Mr. Albert A'
 'Moubarek, Master. Gerios' 'Nye, Mrs. (Elizabeth Ramell)'
 'Crease, Mr. Ernest James' 'Andersson, Miss. Erna Alexandra'
 'Kink, Mr. Vincenz' 'Jenkin, Mr. Stephen Curnow'
 'Goodwin, Miss. Lillian Amy' 'Hood, Mr. Ambrose Jr'
 'Chronopoulos, Mr. Apostolos' 'Bing, Mr. Lee' 'Moen, Mr. Sigurd Hansen'
 'Staneff, Mr. Ivan' 'Moutal, Mr. Rahamin Haim'
 'Caldwell, Master. Alden Gates' 'Dowdell, Miss. Elizabeth'
 'Waelens, Mr. Achille' 'Sheerlinck, Mr. Jan Baptist'
 'McDermott, Miss. Brigdet Delia' 'Carrau, Mr. Francisco M'
 'Ilett, Miss. Bertha'
 'Backstrom, Mrs. Karl Alfred (Maria Mathilda Gustafsson)'
 'Ford, Mr. William Neal' 'Slocovski, Mr. Selman Francis'
 'Fortune, Miss. Mabel Helen' 'Celotti, Mr. Francesco'
 'Christmann, Mr. Emil' 'Andreasson, Mr. Paul Edvin'
 'Chaffee, Mr. Herbert Fuller' 'Dean, Mr. Bertram Frank'
 'Coxon, Mr. Daniel' 'Shorney, Mr. Charles Joseph'
 'Goldschmidt, Mr. George B' 'Greenfield, Mr. William Bertram'

'Doling, Mrs. John T (Ada Julia Bone)' 'Kantor, Mr. Sinai'
 'Petranec, Miss. Matilda' 'Petroff, Mr. Pastcho ("Pentcho")'
 'White, Mr. Richard Frasar' 'Johansson, Mr. Gustaf Joel'
 'Gustafsson, Mr. Anders Vilhelm' 'Mionoff, Mr. Stoytcho'
 'Salkjelsvik, Miss. Anna Kristine' 'Moss, Mr. Albert Johan'
 'Rekic, Mr. Tido' 'Moran, Miss. Bertha' 'Porter, Mr. Walter Chamberlain'
 'Zabour, Miss. Hileni' 'Barton, Mr. David John' 'Jussila, Miss. Katriina'
 'Attalah, Miss. Malake' 'Pekoniemi, Mr. Edvard' 'Connors, Mr. Patrick'
 'Turpin, Mr. William John Robert' 'Baxter, Mr. Quigg Edmond'
 'Andersson, Miss. Ellis Anna Maria' 'Hickman, Mr. Stanley George'
 'Moore, Mr. Leonard Charles' 'Nasser, Mr. Nicholas' 'Webber, Miss. Susan'
 'White, Mr. Percival Wayland' 'Nicola-Yarred, Master. Elias'
 'McMahon, Mr. Martin' 'Madsen, Mr. Fridtjof Arne' 'Peter, Miss. Anna'
 'Ekstrom, Mr. Johan' 'Drazenoic, Mr. Jozef'
 'Coelho, Mr. Domingos Fernando'
 'Robins, Mrs. Alexander A (Grace Charity Laury)'
 'Weisz, Mrs. Leopold (Mathilde Francoise Pede)'
 'Sobey, Mr. Samuel James Hayden' 'Richard, Mr. Emile'
 'Newsom, Miss. Helen Monypeny' 'Futrelle, Mr. Jacques Heath'
 'Osen, Mr. Olaf Elon' 'Giglio, Mr. Victor'
 'Boulos, Mrs. Joseph (Sultana)' 'Nysten, Miss. Anna Sofia'
 'Hakkarainen, Mrs. Pekka Pietari (Elin Matilda Dolck)'
 'Burke, Mr. Jeremiah' 'Andrew, Mr. Edgardo Samuel'
 'Nicholls, Mr. Joseph Charles'
 'Andersson, Mr. August Edvard ("Wennerstrom")'
 'Ford, Miss. Robina Maggie "Ruby"'
 'Navratil, Mr. Michel ("Louis M Hoffman")'
 'Byles, Rev. Thomas Roussel Davids' 'Bateman, Rev. Robert James'
 'Pears, Mrs. Thomas (Edith Wearne)' 'Meo, Mr. Alfonzo'
 'van Billiard, Mr. Austin Blyler' 'Olsen, Mr. Ole Martin'
 'Williams, Mr. Charles Duane' 'Gilnagh, Miss. Katherine "Katie"'
 'Corn, Mr. Harry' 'Smiljanic, Mr. Mile' 'Sage, Master. Thomas Henry'
 'Cribb, Mr. John Hatfield'
 'Watt, Mrs. James (Elizabeth "Bessie" Inglis Milne)'
 'Bengtsson, Mr. John Viktor' 'Calic, Mr. Jovo'
 'Panula, Master. Eino Viljami'
 'Goldsmith, Master. Frank John William "Frankie"'
 'Chibnall, Mrs. (Edith Martha Bowerman)'
 'Skoog, Mrs. William (Anna Bernhardina Karlsson)' 'Baumann, Mr. John D'
 'Ling, Mr. Lee' 'Van der hoef, Mr. Wyckoff' 'Rice, Master. Arthur'
 'Johnson, Miss. Eleanor Ileen' 'Sivola, Mr. Antti Wilhelm'
 'Smith, Mr. James Clinch' 'Klasen, Mr. Klas Albin'
 'Lefebre, Master. Henry Forbes' 'Isham, Miss. Ann Elizabeth'
 'Hale, Mr. Reginald' 'Leonard, Mr. Lionel' 'Sage, Miss. Constance Gladys'
 'Pernot, Mr. Rene' 'Asplund, Master. Clarence Gustaf Hugo'
 'Becker, Master. Richard F' 'Kink-Heilmann, Miss. Luise Gretchen'
 'Rood, Mr. Hugh Roscoe'
 'O\'Brien, Mrs. Thomas (Johanna "Hannah" Godfrey)'

'Romaine, Mr. Charles Hallace ("Mr C Rolmane")' 'Bourke, Mr. John'
 'Turcin, Mr. Stjepan' 'Pinsky, Mrs. (Rosa)' 'Carbines, Mr. William'
 'Andersen-Jensen, Miss. Carla Christine Nielsine'
 'Navratil, Master. Michel M' 'Brown, Mrs. James Joseph (Margaret Tobin)'
 'Lurette, Miss. Elise' 'Mernagh, Mr. Robert'
 'Olsen, Mr. Karl Siegwart Andreas' 'Madigan, Miss. Margaret "Maggie"'
 'Yrois, Miss. Henriette ("Mrs Harbeck")' 'Vande Walle, Mr. Nestor Cyriel'
 'Sage, Mr. Frederick' 'Johanson, Mr. Jakob Alfred' 'Youseff, Mr. Gerious'
 'Cohen, Mr. Gurshon "Gus"' 'Strom, Miss. Telma Matilda'
 'Backstrom, Mr. Karl Alfred' 'Albimona, Mr. Nassef Cassem'
 'Carr, Miss. Helen "Ellen"' 'Blank, Mr. Henry' 'Ali, Mr. Ahmed'
 'Cameron, Miss. Clear Annie' 'Perkin, Mr. John Henry'
 'Givard, Mr. Hans Kristensen' 'Kiernan, Mr. Philip'
 'Newell, Miss. Madeleine' 'Honkanen, Miss. Eliina'
 'Jacobsohn, Mr. Sidney Samuel' 'Bazzani, Miss. Albina'
 'Harris, Mr. Walter' 'Sunderland, Mr. Victor Francis'
 'Bracken, Mr. James H' 'Green, Mr. George Henry' 'Nenkoff, Mr. Christo'
 'Hoyt, Mr. Frederick Maxfield' 'Berglund, Mr. Karl Ivar Sven'
 'Mellors, Mr. William John' 'Lovell, Mr. John Hall ("Henry")'
 'Fahlstrom, Mr. Arne Jonas' 'Lefebvre, Miss. Mathilde'
 'Harris, Mrs. Henry Birkhardt (Irene Wallach)' 'Larsson, Mr. Bengt Edvin'
 'Sjostedt, Mr. Ernst Adolf' 'Asplund, Miss. Lillian Gertrud'
 'Leyson, Mr. Robert William Norman' 'Harknett, Miss. Alice Phoebe'
 'Hold, Mr. Stephen' 'Collyer, Miss. Marjorie "Lottie"'
 'Pengelly, Mr. Frederick William' 'Hunt, Mr. George Henry'
 'Zabour, Miss. Thamine' 'Murphy, Miss. Katherine "Kate"'
 'Coleridge, Mr. Reginald Charles' 'Maenpaa, Mr. Matti Alexanteri'
 'Attalah, Mr. Sleiman' 'Minahan, Dr. William Edward'
 'Lindahl, Miss. Agda Thorilda Viktoria' 'Hamalainen, Mrs. William (Anna)'
 'Beckwith, Mr. Richard Leonard' 'Carter, Rev. Ernest Courtenay'
 'Reed, Mr. James George' 'Strom, Mrs. Wilhelm (Elna Matilda Persson)'
 'Stead, Mr. William Thomas' 'Lobb, Mr. William Arthur'
 'Rosblom, Mrs. Viktor (Helena Wilhelmina)'
 'Touma, Mrs. Darwis (Hanne Youssef Razi)'
 'Thorne, Mrs. Gertrude Maybelle' 'Cherry, Miss. Gladys'
 'Ward, Miss. Anna' 'Parrish, Mrs. (Lutie Davis)' 'Smith, Mr. Thomas'
 'Asplund, Master. Edvin Rojj Felix' 'Taussig, Mr. Emil'
 'Harrison, Mr. William' 'Henry, Miss. Delia' 'Reeves, Mr. David'
 'Panula, Mr. Ernesti Arvid' 'Persson, Mr. Ernst Ulrik'
 'Graham, Mrs. William Thompson (Edith Junkins)' 'Bissette, Miss. Amelia'
 'Cairns, Mr. Alexander' 'Tornquist, Mr. William Henry'
 'Mellinger, Mrs. (Elizabeth Anne Maidment)' 'Natsch, Mr. Charles H'
 'Healy, Miss. Hanora "Nora"' 'Andrews, Miss. Kornelia Theodosia'
 'Lindblom, Miss. Augusta Charlotta' 'Parkes, Mr. Francis "Frank"'
 'Rice, Master. Eric' 'Abbott, Mrs. Stanton (Rosa Hunt)'
 'Duane, Mr. Frank' 'Olsson, Mr. Nils Johan Goransson'
 'de Pelsmaecker, Mr. Alfons' 'Dorking, Mr. Edward Arthur'
 'Smith, Mr. Richard William' 'Stankovic, Mr. Ivan'

'de Mulder, Mr. Theodore' 'Naidenoff, Mr. Penko' 'Hosono, Mr. Masabumi'
 'Connolly, Miss. Kate' 'Barber, Miss. Ellen "Nellie"'
 'Bishop, Mrs. Dickinson H (Helen Walton)' 'Levy, Mr. Rene Jacques'
 'Haas, Miss. Aloisia' 'Mineff, Mr. Ivan' 'Lewy, Mr. Ervin G'
 'Hanna, Mr. Mansour' 'Allison, Miss. Helen Loraine'
 'Saalfeld, Mr. Adolphe' 'Baxter, Mrs. James (Helene DeLaudeniere Chaput)'
 'Kelly, Miss. Anna Katherine "Annie Kate"' 'McCoy, Mr. Bernard'
 'Johnson, Mr. William Cahoon Jr' 'Keane, Miss. Nora A'
 'Williams, Mr. Howard Hugh "Harry"' 'Allison, Master. Hudson Trevor'
 'Fleming, Miss. Margaret'
 'Penasco y Castellana, Mrs. Victor de Satode (Maria Josefa Perez de Soto y Vallejo)'
 'Abelson, Mr. Samuel' 'Francatelli, Miss. Laura Mabel'
 'Hays, Miss. Margaret Bechstein' 'Ryerson, Miss. Emily Borie'
 'Lahtinen, Mrs. William (Anna Sylfven)' 'Hendekovic, Mr. Ignjac'
 'Hart, Mr. Benjamin' 'Nilsson, Miss. Helmina Josefina'
 'Kantor, Mrs. Sinai (Miriam Sternin)' 'Moraweck, Dr. Ernest'
 'Wick, Miss. Mary Natalie'
 'Spedden, Mrs. Frederic Oakley (Margaretta Corning Stone)'
 'Dennis, Mr. Samuel' 'Danoff, Mr. Yoto' 'Slayter, Miss. Hilda Mary'
 'Caldwell, Mrs. Albert Francis (Sylvia Mae Harbaugh)'
 'Sage, Mr. George John Jr' 'Young, Miss. Marie Grice'
 'Nysveen, Mr. Johan Hansen' 'Ball, Mrs. (Ada E Hall)'
 'Goldsmith, Mrs. Frank John (Emily Alice Brown)'
 'Hippach, Miss. Jean Gertrude' 'McCoy, Miss. Agnes' 'Partner, Mr. Austen'
 'Graham, Mr. George Edward' 'Vander Planke, Mr. Leo Edmondus'
 'Frauenthal, Mrs. Henry William (Clara Heinsheimer)' 'Denkoff, Mr. Mitto'
 'Pears, Mr. Thomas Clinton' 'Burns, Miss. Elizabeth Margaret'
 'Dahl, Mr. Karl Edwart' 'Blackwell, Mr. Stephen Weart'
 'Navratil, Master. Edmond Roger' 'Fortune, Miss. Alice Elizabeth'
 'Collander, Mr. Erik Gustaf' 'Sedgwick, Mr. Charles Frederick Waddington'
 'Fox, Mr. Stanley Hubert' 'Brown, Miss. Amelia "Mildred"'
 'Smith, Miss. Marion Elsie' 'Davison, Mrs. Thomas Henry (Mary E Finck)'
 'Coutts, Master. William Loch "William"' 'Dimic, Mr. Jovan'
 'Odahl, Mr. Nils Martin' 'Williams-Lambert, Mr. Fletcher Fellows'
 'Elias, Mr. Tannous' 'Arnold-Franchi, Mr. Josef' 'Yousif, Mr. Wazli'
 'Vanden Steen, Mr. Leo Peter' 'Bowerman, Miss. Elsie Edith'
 'Funk, Miss. Annie Clemmer' 'McGovern, Miss. Mary'
 'Mockler, Miss. Helen Mary "Ellie"' 'Skoog, Mr. Wilhelm'
 'del Carlo, Mr. Sebastiano' 'Barbara, Mrs. (Catherine David)'
 'Asim, Mr. Adola' 'O'Brien, Mr. Thomas' 'Adahl, Mr. Mauritz Nils Martin'
 'Warren, Mrs. Frank Manley (Anna Sophia Atkinson)'
 'Moussa, Mrs. (Mantoura Boulos)' 'Jermyn, Miss. Annie'
 'Aubart, Mme. Leontine Pauline' 'Harder, Mr. George Achilles'
 'Wiklund, Mr. Jakob Alfred' 'Beavan, Mr. William Thomas'
 'Ringhini, Mr. Sante' 'Palsson, Miss. Stina Viola'
 'Meyer, Mrs. Edgar Joseph (Leila Saks)' 'Landergren, Miss. Aurora Adelia'
 'Widener, Mr. Harry Elkins' 'Betros, Mr. Tannous'

'Gustafsson, Mr. Karl Gideon' 'Bidois, Miss. Rosalie'
 'Nakid, Miss. Maria ("Mary")' 'Tikkanen, Mr. Juho'
 'Holverson, Mrs. Alexander Oskar (Mary Aline Towner)'
 'Plotcharsky, Mr. Vasil' 'Davies, Mr. Charles Henry'
 'Goodwin, Master. Sidney Leonard' 'Buss, Miss. Kate'
 'Sadlier, Mr. Matthew' 'Lehmann, Miss. Bertha'
 'Carter, Mr. William Ernest' 'Jansson, Mr. Carl Olof'
 'Gustafsson, Mr. Johan Birger' 'Newell, Miss. Marjorie'
 'Sandstrom, Mrs. Hjalmar (Agnes Charlotta Bengtsson)'
 'Johansson, Mr. Erik' 'Olsson, Miss. Elina' 'McKane, Mr. Peter David'
 'Pain, Dr. Alfred' 'Trout, Mrs. William H (Jessie L)'
 'Niskanen, Mr. Juha' 'Adams, Mr. John' 'Jussila, Miss. Mari Aina'
 'Hakkarainen, Mr. Pekka Pietari' 'Oreskovic, Miss. Marija'
 'Gale, Mr. Shadrach' 'Widegren, Mr. Carl/Charles Peter'
 'Richards, Master. William Rowe' 'Birkeland, Mr. Hans Martin Monsen'
 'Lefebvre, Miss. Ida' 'Sdycoff, Mr. Todor' 'Hart, Mr. Henry'
 'Minahan, Miss. Daisy E' 'Cunningham, Mr. Alfred Fleming'
 'Sundman, Mr. Johan Julian' 'Meek, Mrs. Thomas (Annie Louise Rowley)'
 'Drew, Mrs. James Vivian (Lulu Thorne Christian)'
 'Silven, Miss. Lyyli Karoliina' 'Matthews, Mr. William John'
 'Van Impe, Miss. Catharina' 'Gheorgheff, Mr. Stanio'
 'Charters, Mr. David' 'Zimmerman, Mr. Leo'
 'Danbom, Mrs. Ernst Gilbert (Anna Sigrid Maria Brogren)'
 'Rosblom, Mr. Viktor Richard' 'Wiseman, Mr. Phillippe'
 'Clarke, Mrs. Charles V (Ada Maria Winfield)'
 'Phillips, Miss. Kate Florence ("Mrs Kate Louise Phillips Marshall")'
 'Flynn, Mr. James' 'Pickard, Mr. Berk (Berk Trembisky)'
 'Bjornstrom-Steffansson, Mr. Mauritz Hakan'
 'Thorneycroft, Mrs. Percival (Florence Kate White)'
 'Louch, Mrs. Charles Alexander (Alice Adelaide Slow)'
 'Kallio, Mr. Nikolai Erland' 'Silvey, Mr. William Baird'
 'Carter, Miss. Lucile Polk' 'Ford, Miss. Doolina Margaret "Daisy"
 'Richards, Mrs. Sidney (Emily Hocking)' 'Fortune, Mr. Mark'
 'Kvillner, Mr. Johan Henrik Johannesson'
 'Hart, Mrs. Benjamin (Esther Ada Bloomfield)' 'Hampe, Mr. Leon'
 'Pettersen, Mr. Johan Emil' 'Reynaldo, Ms. Encarnacion'
 'Johannesen-Bratthammer, Mr. Bernt' 'Dodge, Master. Washington'
 'Mellinger, Miss. Madeleine Violet' 'Seward, Mr. Frederic Kimber'
 'Baclini, Miss. Marie Catherine' 'Peuchen, Major. Arthur Godfrey'
 'West, Mr. Edwy Arthur' 'Hagland, Mr. Ingvald Olai Olsen'
 'Foreman, Mr. Benjamin Laventall' 'Goldenberg, Mr. Samuel L'
 'Peduzzi, Mr. Joseph' 'Jalsevac, Mr. Ivan' 'Millet, Mr. Francis Davis'
 'Kenyon, Mrs. Frederick R (Marion)' 'Toomey, Miss. Ellen'
 'O'Connor, Mr. Maurice' 'Anderson, Mr. Harry' 'Morley, Mr. William'
 'Gee, Mr. Arthur H' 'Milling, Mr. Jacob Christian' 'Maisner, Mr. Simon'
 'Goncalves, Mr. Manuel Estanslas' 'Campbell, Mr. William'
 'Smart, Mr. John Montgomery' 'Scanlan, Mr. James'
 'Baclini, Miss. Helene Barbara' 'Keefe, Mr. Arthur' 'Cacic, Mr. Luka'

'West, Mrs. Edwy Arthur (Ada Mary Worth)'
 'Jerwan, Mrs. Amin S (Marie Marthe Thuillard)'
 'Strandberg, Miss. Ida Sofia' 'Clifford, Mr. George Quincy'
 'Renouf, Mr. Peter Henry' 'Braund, Mr. Lewis Richard'
 'Karlsson, Mr. Nils August' 'Hirvonen, Miss. Hildur E'
 'Goodwin, Master. Harold Victor' 'Frost, Mr. Anthony Wood "Archie" '
 'Rouse, Mr. Richard Henry' 'Turkula, Mrs. (Hedwig) '
 'Bishop, Mr. Dickinson H' 'Lefebvre, Miss. Jeannie'
 'Hoyt, Mrs. Frederick Maxfield (Jane Anne Forby) '
 'Kent, Mr. Edward Austin' 'Somerton, Mr. Francis William'
 'Coutts, Master. Eden Leslie "Neville" '
 'Hagland, Mr. Konrad Mathias Reiersen' 'Windelov, Mr. Einar'
 'Molson, Mr. Harry Markland' 'Artagaveytia, Mr. Ramon'
 'Stanley, Mr. Edward Roland' 'Yousseff, Mr. Gerious'
 'Eustis, Miss. Elizabeth Mussey' 'Shellard, Mr. Frederick William'
 'Allison, Mrs. Hudson J C (Bessie Waldo Daniels)' 'Svensson, Mr. Olof'
 'Calic, Mr. Petar' 'Canavan, Miss. Mary' 'O'Sullivan, Miss. Bridget Mary'
 'Laitinen, Miss. Kristina Sofia' 'Maioni, Miss. Roberta'
 'Penasco y Castellana, Mr. Victor de Satode'
 'Quick, Mrs. Frederick Charles (Jane Richards) '
 'Bradley, Mr. George ("George Arthur Brayton") '
 'Olsen, Mr. Henry Margido' 'Lang, Mr. Fang' 'Daly, Mr. Eugene Patrick'
 'Webber, Mr. James' 'McGough, Mr. James Robert'
 'Rothschild, Mrs. Martin (Elizabeth L. Barrett)' 'Coleff, Mr. Satio'
 'Walker, Mr. William Anderson' 'Lemore, Mrs. (Amelia Milley) '
 'Ryan, Mr. Patrick'
 'Angle, Mrs. William A (Florence "Mary" Agnes Hughes) '
 'Pavlovic, Mr. Stefo' 'Perreault, Miss. Anne' 'Vovk, Mr. Janko'
 'Lahoud, Mr. Sarkis' 'Hippach, Mrs. Louis Albert (Ida Sophia Fischer) '
 'Kassem, Mr. Fared' 'Farrell, Mr. James' 'Ridsdale, Miss. Lucy'
 'Farthing, Mr. John' 'Salonen, Mr. Johan Werner'
 'Hocking, Mr. Richard George' 'Quick, Miss. Phyllis May'
 'Toufik, Mr. Nakli' 'Elias, Mr. Joseph Jr'
 'Peter, Mrs. Catherine (Catherine Rizk)' 'Cacic, Miss. Marija'
 'Hart, Miss. Eva Miriam' 'Butt, Major. Archibald Willingham'
 'LeRoy, Miss. Bertha' 'Risien, Mr. Samuel Beard'
 'Frolicher, Miss. Hedwig Margaritha' 'Crosby, Miss. Harriet R'
 'Andersson, Miss. Ingeborg Constanzia'
 'Andersson, Miss. Sigrid Elisabeth' 'Beane, Mr. Edward'
 'Douglas, Mr. Walter Donald' 'Nicholson, Mr. Arthur Ernest'
 'Beane, Mrs. Edward (Ethel Clarke)' 'Padro y Manent, Mr. Julian'
 'Goldsmith, Mr. Frank John' 'Davies, Master. John Morgan Jr'
 'Thayer, Mr. John Borland Jr' 'Sharp, Mr. Percival James R'
 'O'Brien, Mr. Timothy' 'Leeni, Mr. Fahim ("Philip Zenni") '
 'Ohman, Miss. Velin' 'Wright, Mr. George'
 'Duff Gordon, Lady. (Lucille Christiana Sutherland) ("Mrs Morgan") '
 'Robbins, Mr. Victor' 'Taussig, Mrs. Emil (Tillie Mandelbaum) '
 'de Messemaeker, Mrs. Guillaume Joseph (Emma)' 'Morrow, Mr. Thomas Rowan'

'Sivic, Mr. Husein' 'Norman, Mr. Robert Douglas' 'Simmons, Mr. John'
 'Meanwell, Miss. (Marion Ogden)' 'Davies, Mr. Alfred J'
 'Stoytcheff, Mr. Ilia' 'Palsson, Mrs. Nils (Alma Cornelia Berglund)'
 'Doharr, Mr. Tannous' 'Jonsson, Mr. Carl' 'Harris, Mr. George'
 'Appleton, Mrs. Edward Dale (Charlotte Lamson)'
 'Flynn, Mr. John Irwin ("Irving")' 'Kelly, Miss. Mary'
 'Rush, Mr. Alfred George John' 'Patchett, Mr. George'
 'Garside, Miss. Ethel' 'Silvey, Mrs. William Baird (Alice Munger)'
 'Caram, Mrs. Joseph (Maria Elias)' 'Jussila, Mr. Eiriik'
 'Christy, Miss. Julie Rachel'
 'Thayer, Mrs. John Borland (Marian Longstreth Morris)'
 'Downton, Mr. William James' 'Ross, Mr. John Hugo' 'Paulner, Mr. Uscher'
 'Taussig, Miss. Ruth' 'Jarvis, Mr. John Denzil'
 'Frolicher-Stehli, Mr. Maxmillian' 'Gilinski, Mr. Eliezer'
 'Murdlin, Mr. Joseph' 'Rintamaki, Mr. Matti'
 'Stephenson, Mrs. Walter Bertram (Martha Eustis)'
 'Elsbury, Mr. William James' 'Bourke, Miss. Mary'
 'Chapman, Mr. John Henry' 'Van Impe, Mr. Jean Baptiste'
 'Leitch, Miss. Jessie Wills' 'Johnson, Mr. Alfred' 'Boulos, Mr. Hanna'
 'Duff Gordon, Sir. Cosmo Edmund ("Mr Morgan")'
 'Jacobsohn, Mrs. Sidney Samuel (Amy Frances Christy)'
 'Slabenoff, Mr. Petco' 'Harrington, Mr. Charles H'
 'Torber, Mr. Ernst William' 'Homer, Mr. Harry ("Mr E Haven")'
 'Lindell, Mr. Edvard Bengtsson' 'Karaic, Mr. Milan'
 'Daniel, Mr. Robert Williams'
 'Laroche, Mrs. Joseph (Juliette Marie Louise Lafargue)'
 'Shutes, Miss. Elizabeth W'
 'Andersson, Mrs. Anders Johan (Alfrida Konstantia Brogren)'
 'Jardin, Mr. Jose Neto' 'Murphy, Miss. Margaret Jane' 'Horgan, Mr. John'
 'Brocklebank, Mr. William Alfred' 'Herman, Miss. Alice'
 'Danbom, Mr. Ernst Gilbert'
 'Lobb, Mrs. William Arthur (Cordelia K Stanlick)'
 'Becker, Miss. Marion Louise' 'Gavey, Mr. Lawrence' 'Yasbeck, Mr. Antoni'
 'Kimball, Mr. Edwin Nelson Jr' 'Nakid, Mr. Sahid'
 'Hansen, Mr. Henry Damsgaard' 'Bowen, Mr. David John "Dai"'
 'Sutton, Mr. Frederick' 'Kirkland, Rev. Charles Leonard'
 'Longley, Miss. Gretchen Fiske' 'Bostandyeff, Mr. Guentcho'
 'O'Connell, Mr. Patrick D' 'Barkworth, Mr. Algernon Henry Wilson'
 'Lundahl, Mr. Johan Svensson' 'Stahelin-Maeglin, Dr. Max'
 'Parr, Mr. William Henry Marsh' 'Skoog, Miss. Mabel' 'Davis, Miss. Mary'
 'Leinonen, Mr. Antti Gustaf' 'Collyer, Mr. Harvey'
 'Panula, Mrs. Juha (Maria Emilia Ojala)' 'Thorneycroft, Mr. Percival'
 'Jensen, Mr. Hans Peder' 'Sagesser, Mlle. Emma'
 'Skoog, Miss. Margit Elizabeth' 'Foo, Mr. Choong'
 'Baclini, Miss. Eugenie' 'Harper, Mr. Henry Sleeper' 'Cor, Mr. Liudevit'
 'Simonius-Blumer, Col. Oberst Alfons' 'Willey, Mr. Edward'
 'Stanley, Miss. Amy Zillah Elsie' 'Mitkoff, Mr. Mito'
 'Doling, Miss. Elsie' 'Kalvik, Mr. Johannes Halvorsen'

'O\ 'Leary, Miss. Hanora "Norah"' 'Hegarty, Miss. Hanora "Nora"'
 'Hickman, Mr. Leonard Mark' 'Radeff, Mr. Alexander'
 'Bourke, Mrs. John (Catherine)' 'Eitemiller, Mr. George Floyd'
 'Newell, Mr. Arthur Webster' 'Frauenthal, Dr. Henry William'
 'Badt, Mr. Mohamed' 'Colley, Mr. Edward Pomeroy' 'Coleff, Mr. Peju'
 'Lindqvist, Mr. Eino William' 'Hickman, Mr. Lewis'
 'Butler, Mr. Reginald Fenton' 'Rommetvedt, Mr. Knud Paust'
 'Cook, Mr. Jacob' 'Taylor, Mrs. Elmer Zebley (Juliet Cummins Wright)'
 'Brown, Mrs. Thomas William Solomon (Elizabeth Catherine Ford)'
 'Davidson, Mr. Thornton' 'Mitchell, Mr. Henry Michael'
 'Wilhelms, Mr. Charles' 'Watson, Mr. Ennis Hastings'
 'Edvardsson, Mr. Gustaf Hjalmar' 'Sawyer, Mr. Frederick Charles'
 'Turja, Miss. Anna Sofia' 'Goodwin, Mrs. Frederick (Augusta Tyler)'
 'Cardeza, Mr. Thomas Drake Martinez' 'Peters, Miss. Katie'
 'Hassab, Mr. Hammad' 'Olsvigen, Mr. Thor Anderson'
 'Goodwin, Mr. Charles Edward' 'Brown, Mr. Thomas William Solomon'
 'Laroche, Mr. Joseph Philippe Lemercier' 'Panula, Mr. Jaako Arnold'
 'Dakic, Mr. Branko' 'Fischer, Mr. Eberhard Thelander'
 'Madill, Miss. Georgette Alexandra' 'Dick, Mr. Albert Adrian'
 'Karun, Miss. Manca' 'Lam, Mr. Ali' 'Saad, Mr. Khalil' 'Weir, Col. John'
 'Chapman, Mr. Charles Henry' 'Kelly, Mr. James'
 'Mullens, Miss. Katherine "Katie"' 'Thayer, Mr. John Borland'
 'Humblen, Mr. Adolf Mathias Nicolai Olsen'
 'Astor, Mrs. John Jacob (Madeleine Talmadge Force)'
 'Silverthorne, Mr. Spencer Victor' 'Barbara, Miss. Saiide'
 'Gallagher, Mr. Martin' 'Hansen, Mr. Henrik Juul'
 'Morley, Mr. Henry Samuel ("Mr Henry Marshall")'
 'Kelly, Mrs. Florence "Fannie"' 'Calderhead, Mr. Edward Pennington'
 'Cleaver, Miss. Alice'
 'Moubarek, Master. Halim Gonios ("William George")'
 'Mayne, Mlle. Berthe Antonine ("Mrs de Villiers")' 'Klaber, Mr. Herman'
 'Taylor, Mr. Elmer Zebley' 'Larsson, Mr. August Viktor'
 'Greenberg, Mr. Samuel' 'Soholt, Mr. Peter Andreas Lauritz Andersen'
 'Endres, Miss. Caroline Louise' 'Troutt, Miss. Edwina Celia "Winnie"
 'McEvoy, Mr. Michael' 'Johnson, Mr. Malkolm Joackim'
 'Harper, Miss. Annie Jessie "Nina"' 'Jensen, Mr. Svend Lauritz'
 'Gillespie, Mr. William Henry' 'Hodges, Mr. Henry Price'
 'Chambers, Mr. Norman Campbell' 'Oreskovic, Mr. Luka'
 'Renouf, Mrs. Peter Henry (Lillian Jefferys)' 'Mannion, Miss. Margareth'
 'Bryhl, Mr. Kurt Arnold Gottfrid' 'Ilmakangas, Miss. Pieta Sofia'
 'Allen, Miss. Elisabeth Walton' 'Hassan, Mr. Houssein G N'
 'Knight, Mr. Robert J' 'Berriman, Mr. William John'
 'Troupiansky, Mr. Moses Aaron' 'Williams, Mr. Leslie'
 'Ford, Mrs. Edward (Margaret Ann Watson)' 'Lesurer, Mr. Gustave J'
 'Ivanoff, Mr. Kanio' 'Nankoff, Mr. Minko' 'Hawksford, Mr. Walter James'
 'Cavendish, Mr. Tyrell William' 'Ryerson, Miss. Susan Parker "Suzette"
 'McNamee, Mr. Neal' 'Stranden, Mr. Juho' 'Crosby, Capt. Edward Gifford'
 'Abbott, Mr. Rossmore Edward' 'Sinkkonen, Miss. Anna'

'Marvin, Mr. Daniel Warner' 'Connaghton, Mr. Michael' 'Wells, Miss. Joan'
 'Moor, Master. Meier' 'Vande Velde, Mr. Johannes Joseph'
 'Jonkoff, Mr. Lallio' 'Herman, Mrs. Samuel (Jane Laver)'
 'Hamalainen, Master. Viljo' 'Carlsson, Mr. August Sigfrid'
 'Bailey, Mr. Percy Andrew' 'Theobald, Mr. Thomas Leonard'
 'Roths, the Countess. of (Lucy Noel Martha Dyer-Edwards)'
 'Garfirth, Mr. John' 'Nirva, Mr. Iisakki Antino Aijo'
 'Barah, Mr. Hanna Assi' 'Carter, Mrs. William Ernest (Lucile Polk)'
 'Eklund, Mr. Hans Linus' 'Hogeboom, Mrs. John C (Anna Andrews)'
 'Brewer, Dr. Arthur Jackson' 'Mangan, Miss. Mary' 'Moran, Mr. Daniel J'
 'Gronnestad, Mr. Daniel Danielsen' 'Lievens, Mr. Rene Aime'
 'Jensen, Mr. Niels Peder' 'Mack, Mrs. (Mary)' 'Elias, Mr. Dibo'
 'Hocking, Mrs. Elizabeth (Eliza Needs)'
 'Myhrman, Mr. Pehr Fabian Oliver Malkolm' 'Tobin, Mr. Roger'
 'Emanuel, Miss. Virginia Ethel' 'Kilgannon, Mr. Thomas J'
 'Robert, Mrs. Edward Scott (Elisabeth Walton McMillan)'
 'Ayoub, Miss. Banoura' 'Dick, Mrs. Albert Adrian (Vera Gillespie)'
 'Long, Mr. Milton Clyde' 'Johnston, Mr. Andrew G' 'Ali, Mr. William'
 'Harmer, Mr. Abraham (David Lishin)' 'Sjoblom, Miss. Anna Sofia'
 'Rice, Master. George Hugh' 'Dean, Master. Bertram Vere'
 'Guggenheim, Mr. Benjamin' 'Keane, Mr. Andrew "Andy" '
 'Gaskell, Mr. Alfred' 'Sage, Miss. Stella Anna'
 'Hoyt, Mr. William Fisher' 'Dantcheff, Mr. Ristiu' 'Otter, Mr. Richard'
 'Leader, Dr. Alice (Farnham)' 'Osman, Mrs. Mara'
 'Ibrahim Shawah, Mr. Yousseff'
 'Van Impe, Mrs. Jean Baptiste (Rosalie Paula Govaert)'
 'Ponesell, Mr. Martin' 'Collyer, Mrs. Harvey (Charlotte Annie Tate)'
 'Carter, Master. William Thornton II' 'Thomas, Master. Assad Alexander'
 'Hedman, Mr. Oskar Arvid' 'Johansson, Mr. Karl Johan'
 'Andrews, Mr. Thomas Jr' 'Pettersson, Miss. Ellen Natalia'
 'Meyer, Mr. August' 'Chambers, Mrs. Norman Campbell (Bertha Griggs)'
 'Alexander, Mr. William' 'Lester, Mr. James' 'Slemen, Mr. Richard James'
 'Andersson, Miss. Ebba Iris Alfrida' 'Tomlin, Mr. Ernest Portage'
 'Fry, Mr. Richard' 'Heininen, Miss. Wendla Maria' 'Mallet, Mr. Albert'
 'Holm, Mr. John Fredrik Alexander' 'Skoog, Master. Karl Thorsten'
 'Hays, Mrs. Charles Melville (Clara Jennings Gregg)' 'Lulic, Mr. Nikola'
 'Reuchlin, Jonkheer. John George' 'Moor, Mrs. (Beila)'
 'Panula, Master. Urho Abraham' 'Flynn, Mr. John' 'Lam, Mr. Len'
 'Mallet, Master. Andre' 'McCormack, Mr. Thomas Joseph'
 'Stone, Mrs. George Nelson (Martha Evelyn)'
 'Yasbeck, Mrs. Antoni (Selini Alexander)'
 'Richards, Master. George Sibley' 'Saad, Mr. Amin'
 'Augustsson, Mr. Albert' 'Allum, Mr. Owen George'
 'Compton, Miss. Sara Rebecca' 'Pasic, Mr. Jakob' 'Sirota, Mr. Maurice'
 'Chip, Mr. Chang' 'Marechal, Mr. Pierre' 'Alhomaki, Mr. Ilmari Rudolf'
 'Mudd, Mr. Thomas Charles' 'Serepeca, Miss. Augusta'
 'Lemberopolous, Mr. Peter L' 'Culumovic, Mr. Jeso' 'Abbing, Mr. Anthony'
 'Sage, Mr. Douglas Bullen' 'Markoff, Mr. Marin' 'Harper, Rev. John'

'Goldenberg, Mrs. Samuel L (Edwiga Grabowska)'
 'Andersson, Master. Sigvard Harald Elias' 'Svensson, Mr. Johan'
 'Boulos, Miss. Nourelain' 'Lines, Miss. Mary Conover'
 'Carter, Mrs. Ernest Courtenay (Lilian Hughes)'
 'Aks, Mrs. Sam (Leah Rosen)' 'Wick, Mrs. George Dennick (Mary Hitchcock)'
 'Daly, Mr. Peter Denis ' 'Baclini, Mrs. Solomon (Latifa Qurban)'
 'Razi, Mr. Raihed' 'Hansen, Mr. Claus Peter'
 'Giles, Mr. Frederick Edward'
 'Swift, Mrs. Frederick Joel (Margaret Welles Barron)'
 'Sage, Miss. Dorothy Edith "Dolly"' 'Gill, Mr. John William'
 'Bystrom, Mrs. (Karolina)' 'Duran y More, Miss. Asuncion'
 'Roebling, Mr. Washington Augustus II' 'van Melkebeke, Mr. Philemon'
 'Johnson, Master. Harold Theodor' 'Balkic, Mr. Cerin'
 'Beckwith, Mrs. Richard Leonard (Sallie Monypeny)'
 'Carlsson, Mr. Frans Olof' 'Vander Cruyssen, Mr. Victor'
 'Abelson, Mrs. Samuel (Hannah Wizosky)'
 'Najib, Miss. Adele Kiamie "Jane"' 'Gustafsson, Mr. Alfred Ossian'
 'Petroff, Mr. Nedelio' 'Laleff, Mr. Kristo'
 'Potter, Mrs. Thomas Jr (Lily Alexenia Wilson)'
 'Shelley, Mrs. William (Imanita Parrish Hall)' 'Markun, Mr. Johann'
 'Dahlberg, Miss. Gerda Ulrika' 'Banfield, Mr. Frederick James'
 'Sutehall, Mr. Henry Jr' 'Rice, Mrs. William (Margaret Norton)'
 'Montvila, Rev. Juozas' 'Graham, Miss. Margaret Edith'
 'Johnston, Miss. Catherine Helen "Carrie"' 'Behr, Mr. Karl Howell'
 'Dooley, Mr. Patrick']

Sex =

['male' 'female']

Ticket =

['A/5 21171' 'PC 17599' 'STON/O2. 3101282' '113803' '373450' '330877'
 '17463' '349909' '347742' '237736' 'PP 9549' '113783' 'A/5. 2151'
 '347082' '350406' '248706' '382652' '244373' '345763' '2649' '239865'
 '248698' '330923' '113788' '347077' '2631' '19950' '330959' '349216'
 'PC 17601' 'PC 17569' '335677' 'C.A. 24579' 'PC 17604' '113789' '2677'
 'A./5. 2152' '345764' '2651' '7546' '11668' '349253' 'SC/Paris 2123'
 '330958' 'S.C./A.4. 23567' '370371' '14311' '2662' '349237' '3101295'
 'A/4. 39886' 'PC 17572' '2926' '113509' '19947' 'C.A. 31026' '2697'
 'C.A. 34651' 'CA 2144' '2669' '113572' '36973' '347088' 'PC 17605' '2661'
 'C.A. 29395' 'S.P. 3464' '3101281' '315151' 'C.A. 33111' 'S.O.C. 14879'
 '2680' '1601' '348123' '349208' '374746' '248738' '364516' '345767'
 '345779' '330932' '113059' 'SO/C 14885' '3101278' 'W./C. 6608'
 'SOTON/OQ 392086' '343275' '343276' '347466' 'W.E.P. 5734' 'C.A. 2315'
 '364500' '374910' 'PC 17754' 'PC 17759' '231919' '244367' '349245'
 '349215' '35281' '7540' '3101276' '349207' '343120' '312991' '349249'
 '371110' '110465' '2665' '324669' '4136' '2627' 'STON/O 2. 3101294'
 '370369' 'PC 17558' 'A4. 54510' '27267' '370372' 'C 17369' '2668'
 '347061' '349241' 'SOTON/O.Q. 3101307' 'A/5. 3337' '228414' 'C.A. 29178'
 'SC/PARIS 2133' '11752' '7534' 'PC 17593' '2678' '347081'
 'STON/O2. 3101279' '365222' '231945' 'C.A. 33112' '350043' '230080']

'244310' 'S.O.P. 1166' '113776' 'A.5. 11206' 'A/5. 851' 'Fa 265302'
 'PC 17597' '35851' 'SOTON/OQ 392090' '315037' 'CA. 2343' '371362'
 'C.A. 33595' '347068' '315093' '363291' '113505' 'PC 17318' '111240'
 'STON/O 2. 3101280' '17764' '350404' '4133' 'PC 17595' '250653' 'LINE'
 'SC/PARIS 2131' '230136' '315153' '113767' '370365' '111428' '364849'
 '349247' '234604' '28424' '350046' 'PC 17610' '368703' '4579' '370370'
 '248747' '345770' '3101264' '2628' 'A/5 3540' '347054' '2699' '367231'
 '112277' 'SOTON/O.Q. 3101311' 'F.C.C. 13528' 'A/5 21174' '250646'
 '367229' '35273' 'STON/O2. 3101283' '243847' '11813' 'W/C 14208'
 'SOTON/OQ 392089' '220367' '21440' '349234' '19943' 'PP 4348' 'SW/PP 751'
 'A/5 21173' '236171' '347067' '237442' 'C.A. 29566' 'W./C. 6609' '26707'
 'C.A. 31921' '28665' 'SC/W 1585' '367230' 'W./C. 14263'
 'STON/O 2. 3101275' '2694' '19928' '347071' '250649' '11751' '244252'
 '362316' '113514' 'A/5. 3336' '370129' '2650' 'PC 17585' '110152'
 'PC 17755' '230433' '384461' '110413' '112059' '382649' 'C.A. 17248'
 '347083' 'PC 17582' 'PC 17760' '113798' '250644' 'PC 17596' '370375'
 '13502' '347073' '239853' 'C.A. 2673' '336439' '347464' '345778'
 'A/5. 10482' '113056' '349239' '345774' '349206' '237798' '370373'
 '19877' '11967' 'SC/Paris 2163' '349236' '349233' 'PC 17612' '2693'
 '113781' '19988' '9234' '367226' '226593' 'A/5 2466' '17421' 'PC 17758'
 'P/PP 3381' 'PC 17485' '11767' 'PC 17608' '250651' '349243'
 'F.C.C. 13529' '347470' '29011' '36928' '16966' 'A/5 21172' '349219'
 '234818' '345364' '28551' '111361' '113043' 'PC 17611' '349225' '7598'
 '113784' '248740' '244361' '229236' '248733' '31418' '386525'
 'C.A. 37671' '315088' '7267' '113510' '2695' '2647' '345783' '237671'
 '330931' '330980' 'SC/PARIS 2167' '2691' 'SOTON/O.Q. 3101310' 'C 7076'
 '110813' '2626' '14313' 'PC 17477' '11765' '3101267' '323951' 'C 7077'
 '113503' '2648' '347069' 'PC 17757' '2653' 'STON/O 2. 3101293' '349227'
 '27849' '367655' 'SC 1748' '113760' '350034' '3101277' '350052' '350407'
 '28403' '244278' '240929' 'STON/O 2. 3101289' '341826' '4137' '315096'
 '28664' '347064' '29106' '312992' '349222' '394140' 'STON/O 2. 3101269'
 '343095' '28220' '250652' '28228' '345773' '349254' 'A/5. 13032' '315082'
 '347080' 'A/4. 34244' '2003' '250655' '364851' 'SOTON/O.Q. 392078'
 '110564' '376564' 'SC/AH 3085' 'STON/O 2. 3101274' '13507' 'C.A. 18723'
 '345769' '347076' '230434' '65306' '33638' '113794' '2666' '113786'
 '65303' '113051' '17453' 'A/5 2817' '349240' '13509' '17464'
 'F.C.C. 13531' '371060' '19952' '364506' '111320' '234360' 'A/S 2816'
 'SOTON/O.Q. 3101306' '113792' '36209' '323592' '315089' 'SC/AH Basle 541'
 '7553' '31027' '3460' '350060' '3101298' '239854' 'A/5 3594' '4134'
 '11771' 'A.5. 18509' '65304' 'SOTON/OQ 3101317' '113787' 'PC 17609'
 'A/4 45380' '36947' 'C.A. 6212' '350035' '315086' '364846' '330909'
 '4135' '26360' '111427' 'C 4001' '382651' 'SOTON/OQ 3101316' 'PC 17473'
 'PC 17603' '349209' '36967' 'C.A. 34260' '226875' '349242' '12749'
 '349252' '2624' '2700' '367232' 'W./C. 14258' 'PC 17483' '3101296'
 '29104' '2641' '2690' '315084' '113050' 'PC 17761' '364498' '13568'
 'WE/P 5735' '2908' '693' 'SC/PARIS 2146' '244358' '330979' '2620'
 '347085' '113807' '11755' '345572' '372622' '349251' '218629'
 'SOTON/OQ 392082' 'SOTON/O.Q. 392087' 'A/4 48871' '349205' '2686'

'350417' 'S.W./PP 752' '11769' 'PC 17474' '14312' 'A/4. 20589' '358585'
 '243880' '2689' 'STON/O 2. 3101286' '237789' '13049' '3411' '237565'
 '13567' '14973' 'A./5. 3235' 'STON/O 2. 3101273' 'A/5 3902' '364848'
 'SC/AH 29037' '248727' '2664' '349214' '113796' '364511' '111426'
 '349910' '349246' '113804' 'SOTON/O.Q. 3101305' '370377' '364512'
 '220845' '31028' '2659' '11753' '350029' '54636' '36963' '219533'
 '349224' '334912' '27042' '347743' '13214' '112052' '237668'
 'STON/O 2. 3101292' '350050' '349231' '13213' 'S.O./P.P. 751' 'CA. 2314'
 '349221' '8475' '330919' '365226' '349223' '29751' '2623' '5727' '349210'
 'STON/O 2. 3101285' '234686' '312993' 'A/5 3536' '19996' '29750'
 'F.C. 12750' 'C.A. 24580' '244270' '239856' '349912' '342826' '4138'
 '330935' '6563' '349228' '350036' '24160' '17474' '349256' '2672'
 '113800' '248731' '363592' '35852' '348121' 'PC 17475' '36864' '350025'
 '223596' 'PC 17476' 'PC 17482' '113028' '7545' '250647' '348124' '34218'
 '36568' '347062' '350048' '12233' '250643' '113806' '315094' '36866'
 '236853' 'STON/O2. 3101271' '239855' '28425' '233639' '349201' '349218'
 '16988' '376566' 'STON/O 2. 3101288' '250648' '113773' '335097' '29103'
 '392096' '345780' '349204' '350042' '29108' '363294' 'SOTON/O2 3101272'
 '2663' '347074' '112379' '364850' '8471' '345781' '350047' 'S.O./P.P. 3'
 '2674' '29105' '347078' '383121' '36865' '2687' '113501' 'W./C. 6607'
 'SOTON/O.Q. 3101312' '374887' '3101265' '12460' 'PC 17600' '349203'
 '28213' '17465' '349244' '2685' '2625' '347089' '347063' '112050'
 '347087' '248723' '3474' '28206' '364499' '112058' 'STON/O2. 3101290'
 'S.C./PARIS 2079' 'C 7075' '315098' '19972' '368323' '367228' '2671'
 '347468' '2223' 'PC 17756' '315097' '392092' '11774' 'SOTON/O2 3101287'
 '2683' '315090' 'C.A. 5547' '349213' '347060' 'PC 17592' '392091'
 '113055' '2629' '350026' '28134' '17466' '233866' '236852'
 'SC/PARIS 2149' 'PC 17590' '345777' '349248' '695' '345765' '2667'
 '349212' '349217' '349257' '7552' 'C.A./SOTON 34068' 'SOTON/OQ 392076'
 '211536' '112053' '111369' '370376']

Cabin =

[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']

Embarked =

['S' 'C' 'Q' nan]

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

```
[51]:
```

	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

```
[52]: df.describe(include = np.object)    #List of data types to be included while
      ↪describing dataframe. Default is None
```

<ipython-input-52-a823d8da510e>:1: DeprecationWarning: `np.object` is a deprecated alias for the builtin `object`. To silence this warning, use `object` by itself. Doing this will not modify any behavior and is safe.
 Deprecated in NumPy 1.20; for more details and guidance:
<https://numpy.org/devdocs/release/1.20.0-notes.html#deprecations>
 df.describe(include = np.object)

```
[52]:
```

	Name	Sex	Ticket	Cabin	Embarked
count	891	891	891	204	889
unique	891	2	681	147	3
top	Frost, Mr. Anthony Wood "Archie"	male	1601	C23 C25 C27	S
freq	1	577	7	4	644

```
[78]: df.describe(include = 'all')    #List of data types to be included while
      ↪describing dataframe. Default is None
```

```
[78]:
```

	Survived	Pclass	Name	Sex	Age	\
count	891.000000	891.000000	891	891	714.000000	
unique	NaN	NaN	891	2	NaN	
top	NaN	NaN	Wells, Miss. Joan	male	NaN	
freq	NaN	NaN	1	577	NaN	
mean	0.383838	2.308642	NaN	NaN	29.699118	
std	0.486592	0.836071	NaN	NaN	14.526497	
min	0.000000	1.000000	NaN	NaN	0.420000	
25%	0.000000	2.000000	NaN	NaN	20.125000	
50%	0.000000	3.000000	NaN	NaN	28.000000	
75%	1.000000	3.000000	NaN	NaN	38.000000	
max	1.000000	3.000000	NaN	NaN	80.000000	

	SibSp	Parch	Ticket	Fare	Cabin	Embarked
count	891.000000	891.000000	891	891.000000	204	889
unique	NaN	NaN	681	NaN	147	3

top	NaN	NaN	CA. 2343	NaN	B96 B98	S
freq	NaN	NaN	7	NaN	4	644
mean	0.523008	0.381594	NaN	32.204208	NaN	NaN
std	1.102743	0.806057	NaN	49.693429	NaN	NaN
min	0.000000	0.000000	NaN	0.000000	NaN	NaN
25%	0.000000	0.000000	NaN	7.910400	NaN	NaN
50%	0.000000	0.000000	NaN	14.454200	NaN	NaN
75%	1.000000	0.000000	NaN	31.000000	NaN	NaN
max	8.000000	6.000000	NaN	512.329200	NaN	NaN

```
[28]: df.describe(percentiles = [0.1, 0.6, 0.9])
```

```
[28]:
```

	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
10%	0.000000	1.000000	14.000000	0.000000	0.000000	7.550000
50%	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
60%	0.000000	3.000000	31.800000	0.000000	0.000000	21.679200
90%	1.000000	3.000000	50.000000	1.000000	2.000000	77.958300
max	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

```
[53]: # list like data type of numbers between 0-1 to return the respective percentile
df.describe(include = 'all', percentiles = [0.1, 0.9])
```

```
[53]:
```

	Survived	Pclass	Name	Sex	\
count	891.000000	891.000000	891	891	
unique	NaN	NaN	891	2	
top	NaN	NaN	Frost, Mr. Anthony Wood "Archie"	male	
freq	NaN	NaN	1	577	
mean	0.383838	2.308642	NaN	NaN	
std	0.486592	0.836071	NaN	NaN	
min	0.000000	1.000000	NaN	NaN	
10%	0.000000	1.000000	NaN	NaN	
50%	0.000000	3.000000	NaN	NaN	
90%	1.000000	3.000000	NaN	NaN	
max	1.000000	3.000000	NaN	NaN	

	Age	SibSp	Parch	Ticket	Fare	Cabin	\
count	714.000000	891.000000	891.000000	891	891.000000	204	
unique	NaN	NaN	NaN	681	NaN	147	
top	NaN	NaN	NaN	1601	NaN	C23 C25 C27	
freq	NaN	NaN	NaN	7	NaN	4	
mean	29.699118	0.523008	0.381594	NaN	32.204208	NaN	
std	14.526497	1.102743	0.806057	NaN	49.693429	NaN	
min	0.420000	0.000000	0.000000	NaN	0.000000	NaN	

10%	14.000000	0.000000	0.000000	NaN	7.550000	NaN
50%	28.000000	0.000000	0.000000	NaN	14.454200	NaN
90%	50.000000	1.000000	2.000000	NaN	77.958300	NaN
max	80.000000	8.000000	6.000000	NaN	512.329200	NaN

	Embarked
count	889
unique	3
top	S
freq	644
mean	NaN
std	NaN
min	NaN
10%	NaN
50%	NaN
90%	NaN
max	NaN

```
[53]: df.drop(['Name', 'Ticket', "Cabin"], axis = 1, inplace = True)
```

```
[54]: df
```

```
[54]:
```

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
PassengerId								
1	0	3	male	22.0	1	0	7.2500	S
2	1	1	female	38.0	1	0	71.2833	C
3	1	3	female	26.0	0	0	7.9250	S
4	1	1	female	35.0	1	0	53.1000	S
5	0	3	male	35.0	0	0	8.0500	S
...
887	0	2	male	27.0	0	0	13.0000	S
888	1	1	female	19.0	0	0	30.0000	S
889	0	3	female	NaN	1	2	23.4500	S
890	1	1	male	26.0	0	0	30.0000	C
891	0	3	male	32.0	0	0	7.7500	Q

[891 rows x 8 columns]

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

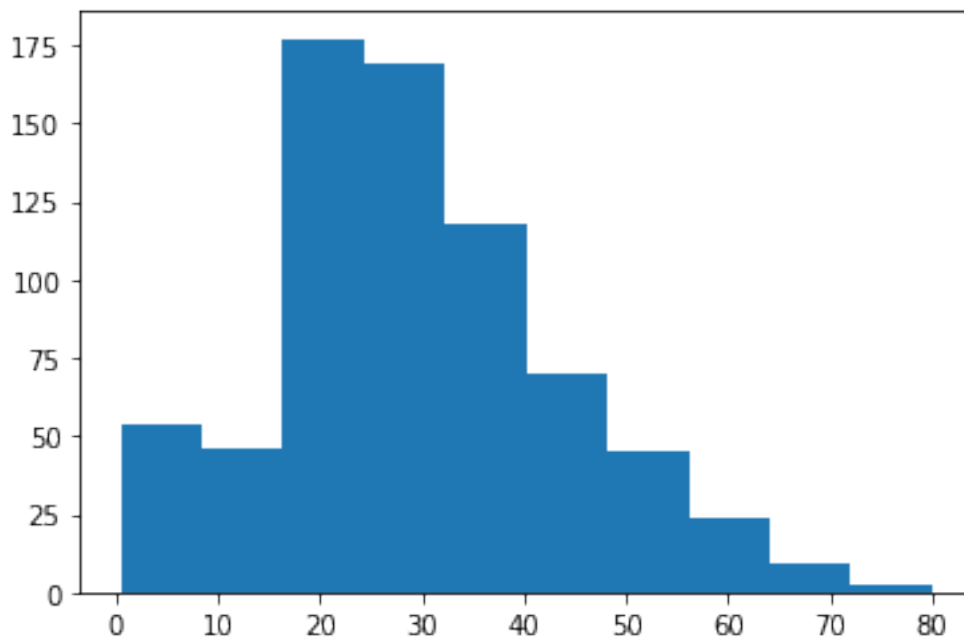
```
[55]:
```

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
PassengerId								
1	0	3	male	22.0	1	0	7.2500	S
2	1	1	female	38.0	1	0	71.2833	C
3	1	3	female	26.0	0	0	7.9250	S
4	1	1	female	35.0	1	0	53.1000	S
5	0	3	male	35.0	0	0	8.0500	S

```
[57]: df.fillna(value = 100, axis = 1).info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 891 entries, 1 to 891
Data columns (total 8 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         891 non-null    float64
 4   SibSp       891 non-null    int64
 5   Parch       891 non-null    int64
 6   Fare        891 non-null    float64
 7   Embarked    891 non-null    object
dtypes: float64(2), int64(4), object(2)
memory usage: 94.9+ KB
```

```
[61]: plt.hist(df.Age)  #This parameter is an optional parameter and it contains the
      ↳ integer or sequence or string.
      plt.show()
```



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

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 891 entries, 1 to 891
```

Data columns (total 8 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

dtypes: float64(2), int64(4), object(2)
memory usage: 94.9+ KB

```
[56]: df1 = df.copy()

# DataFrame.copy(deep=True)

# When deep=True (default), a new object will be created with a copy of the
#   ↳ calling object's data and indices.
# Modifications to the data or indices of the copy will not be reflected in the
#   ↳ original object (see notes below).

# When deep=False, a new object will be created without copying the calling
#   ↳ object's data or
# index (only references to the data and index are copied). Any changes to the
#   ↳ data of the original will
# be reflected in the shallow copy (and vice versa).
```

```
[57]: df1
```

```
[57]:
```

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
PassengerId								
1	0	3	male	22.0	1	0	7.2500	S
2	1	1	female	38.0	1	0	71.2833	C
3	1	3	female	26.0	0	0	7.9250	S
4	1	1	female	35.0	1	0	53.1000	S
5	0	3	male	35.0	0	0	8.0500	S
...
887	0	2	male	27.0	0	0	13.0000	S
888	1	1	female	19.0	0	0	30.0000	S
889	0	3	female	NaN	1	2	23.4500	S
890	1	1	male	26.0	0	0	30.0000	C
891	0	3	male	32.0	0	0	7.7500	Q

[891 rows x 8 columns]

```
[59]: df1['Age'].median()
```

```
[59]: 28.0
```

```
[58]: df1.fillna({"Age" : df1['Age'].median(), "Embarked" : "S"}, inplace = True)
# df1['Age'].fillna(df1['Age'].median(), inplace = True)
# df1['Embarked'].fillna('S', inplace= True)
```

```
[58]:
```

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
PassengerId								
1	0	3	male	22.0	1	0	7.2500	S
2	1	1	female	38.0	1	0	71.2833	C
3	1	3	female	26.0	0	0	7.9250	S
4	1	1	female	35.0	1	0	53.1000	S
5	0	3	male	35.0	0	0	8.0500	S
...
887	0	2	male	27.0	0	0	13.0000	S
888	1	1	female	19.0	0	0	30.0000	S
889	0	3	female	28.0	1	2	23.4500	S
890	1	1	male	26.0	0	0	30.0000	C
891	0	3	male	32.0	0	0	7.7500	Q

[891 rows x 8 columns]

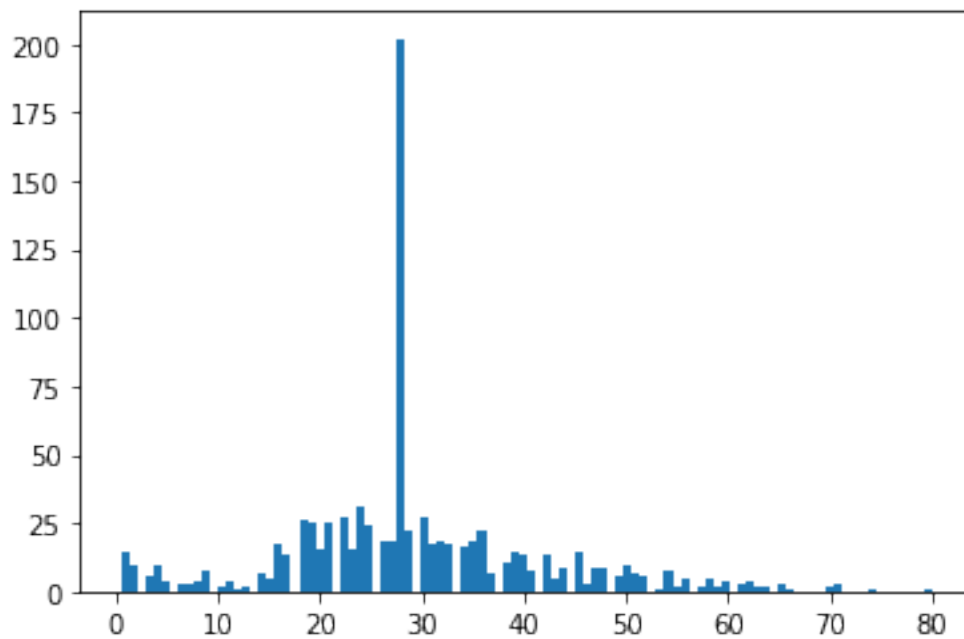
```
[67]: df['Age'].median()
```

```
[67]: 28.0
```

```
[68]: df1.info()
```

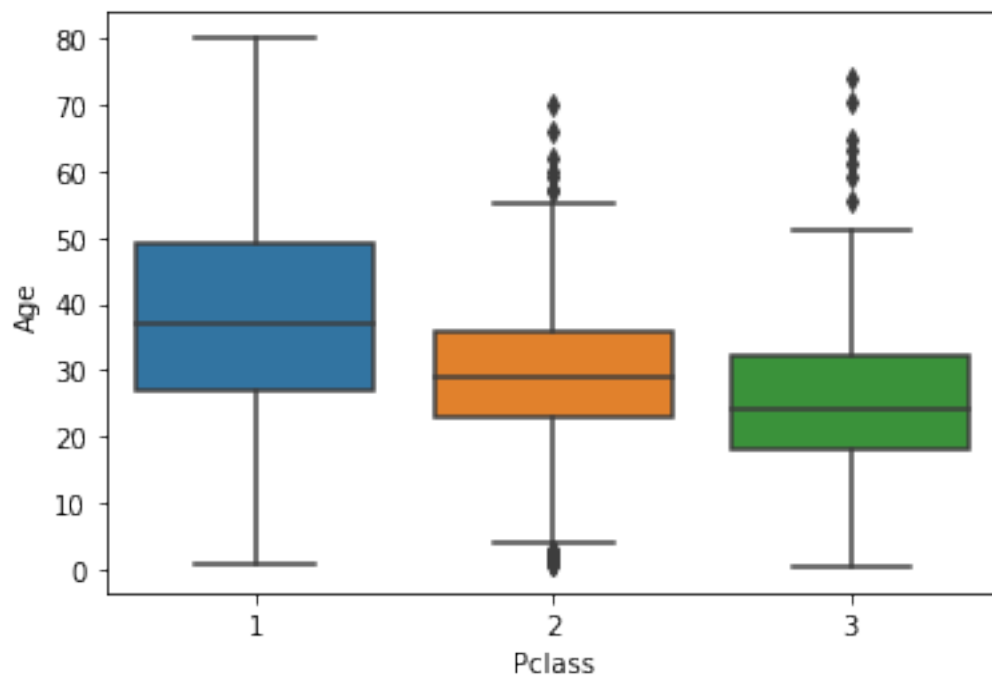
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 891 entries, 1 to 891
Data columns (total 8 columns):
#   Column      Non-Null Count  Dtype
---  -
0   Survived    891 non-null    int64
1   Pclass      891 non-null    int64
2   Sex         891 non-null    object
3   Age         891 non-null    float64
4   SibSp       891 non-null    int64
5   Parch       891 non-null    int64
6   Fare        891 non-null    float64
7   Embarked    891 non-null    object
dtypes: float64(2), int64(4), object(2)
memory usage: 94.9+ KB
```

```
[69]: plt.hist(df1.Age, bins = 100)
plt.show()
```



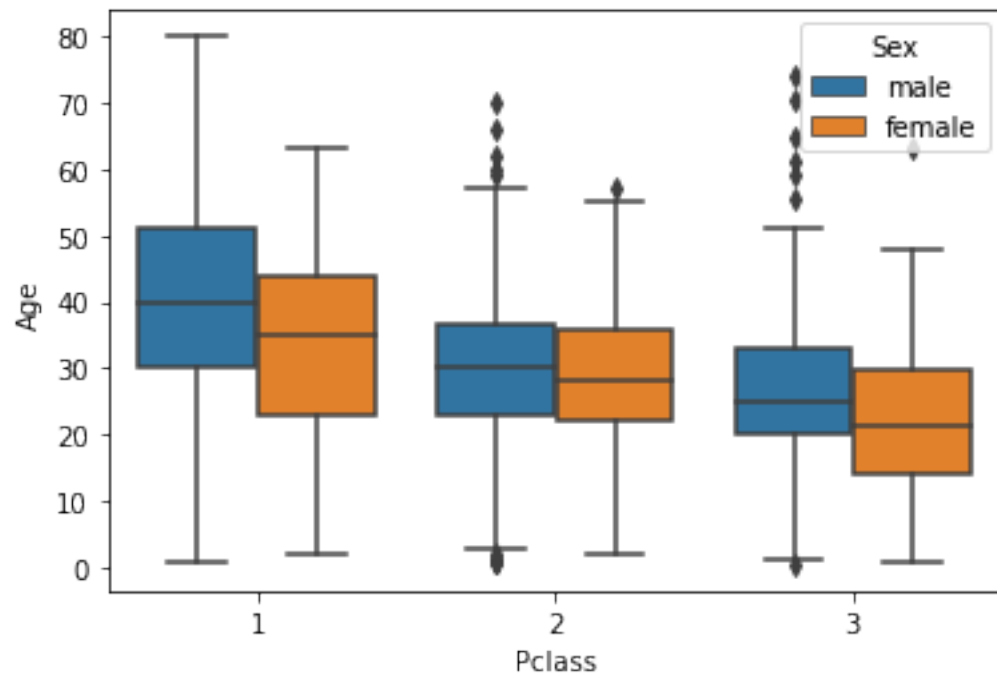
```
[70]: sns.boxplot(x = df.Pclass, y = df.Age)
```

```
[70]: <AxesSubplot:xlabel='Pclass', ylabel='Age'>
```



```
[71]: sns.boxplot(x = df.Pclass, y = df.Age, hue = df.Sex)
```

```
[71]: <AxesSubplot:xlabel='Pclass', ylabel='Age'>
```



```
[93]: df
```

```
[93]:
```

PassengerId	Survived	Pclass	\
1	0	3	
2	1	1	
3	1	3	
4	1	1	
5	0	3	
...	
887	0	2	
888	1	1	
889	0	3	
890	1	1	
891	0	3	

PassengerId	Name	Sex	Age	\
-------------	------	-----	-----	---

```

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

```

	SibSp	Parch	Ticket	Fare	Cabin	Embarked
PassengerId						
1	1	0	A/5 21171	7.2500	NaN	S
2	1	0	PC 17599	71.2833	C85	C
3	0	0	STON/O2. 3101282	7.9250	NaN	S
4	1	0	113803	53.1000	C123	S
5	0	0	373450	8.0500	NaN	S
...
887	0	0	211536	13.0000	NaN	S
888	0	0	112053	30.0000	B42	S
889	1	2	W./C. 6607	23.4500	NaN	S
890	0	0	111369	30.0000	C148	C
891	0	0	370376	7.7500	NaN	Q

[891 rows x 11 columns]

```
[166]: df2 = df.copy()
df2
```

```
[166]:
Survived  Pclass  \
PassengerId
1          0      3
2          1      1
3          1      3
4          1      1
5          0      3
...
887         0      2
888         1      1
889         0      3
890         1      1
891         0      3

```

	Name	Sex	Age	\
PassengerId				

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

PassengerId	SibSp	Parch	Ticket	Fare	Cabin	Embarked
1	1	0	A/5 21171	7.2500	NaN	S
2	1	0	PC 17599	71.2833	C85	C
3	0	0	STON/O2. 3101282	7.9250	NaN	S
4	1	0	113803	53.1000	C123	S
5	0	0	373450	8.0500	NaN	S
...
887	0	0	211536	13.0000	NaN	S
888	0	0	112053	30.0000	B42	S
889	1	2	W./C. 6607	23.4500	NaN	S
890	0	0	111369	30.0000	C148	C
891	0	0	370376	7.7500	NaN	Q

[891 rows x 11 columns]

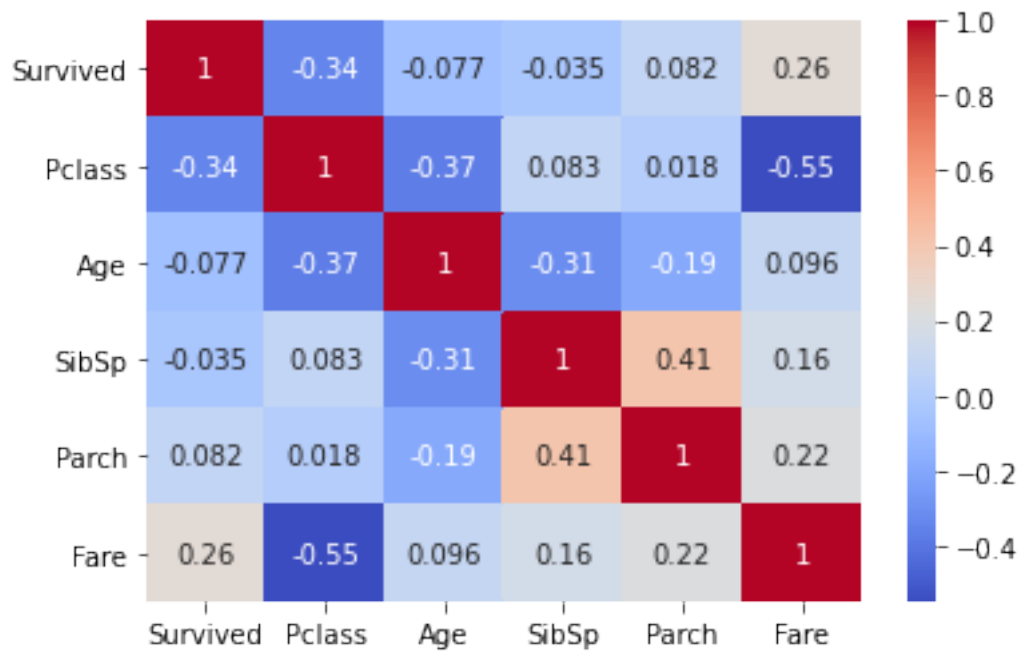
```
[167]: df2.corr()
```

```
[167]:
```

	Survived	Pclass	Age	SibSp	Parch	Fare
Survived	1.000000	-0.338481	-0.077221	-0.035322	0.081629	0.257307
Pclass	-0.338481	1.000000	-0.369226	0.083081	0.018443	-0.549500
Age	-0.077221	-0.369226	1.000000	-0.308247	-0.189119	0.096067
SibSp	-0.035322	0.083081	-0.308247	1.000000	0.414838	0.159651
Parch	0.081629	0.018443	-0.189119	0.414838	1.000000	0.216225
Fare	0.257307	-0.549500	0.096067	0.159651	0.216225	1.000000

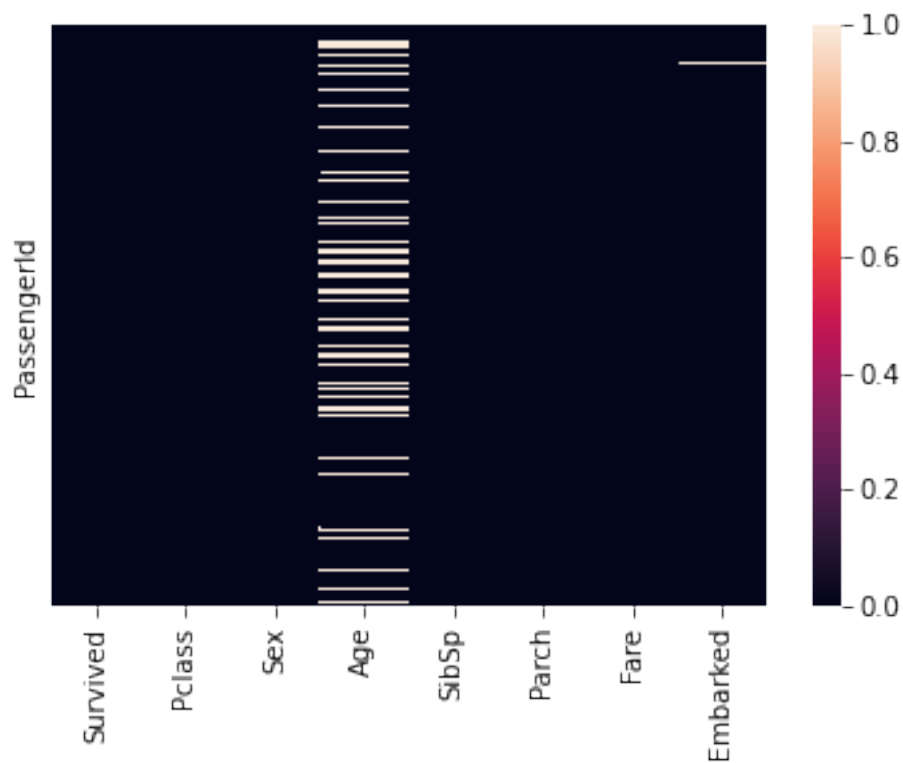
```
[169]: sns.heatmap(df2.corr(), annot = True, cmap = 'coolwarm')
```

```
[169]: <AxesSubplot:>
```



```
[74]: sns.heatmap(df2.isnull(), yticklabels=False)
```

```
[74]: <AxesSubplot:ylabel='PassengerId'>
```

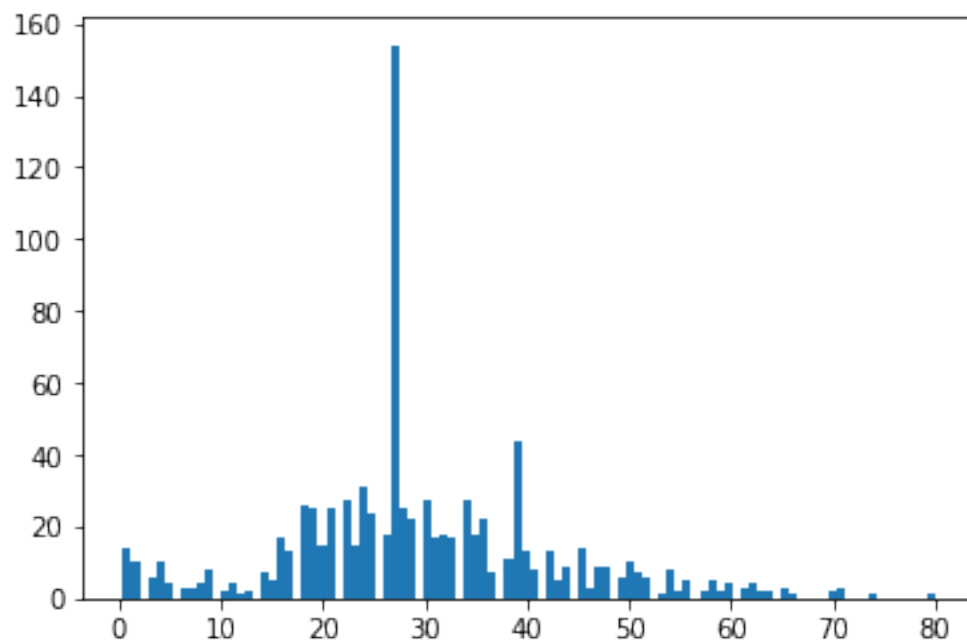


```
[ ]:
```

```
[75]: def fun(c) :  
      age = c[0]  
      pclass = c[1]  
  
      if pd.isnull(age) :  
          if pclass == 1 :  
              return 39  
          elif pclass == 2 :  
              return 34  
          else :  
              return 27  
      else :  
          return age
```

```
[76]: df2['Age'] = df2[['Age', 'Pclass']].apply(fun, axis = 1)
```

```
[77]: plt.hist(df2.Age, bins = 100)  
      plt.show()
```



```
[78]: df2.fillna({"Embarked" : "S"}, inplace = True)
```

```
[79]: df2.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 891 entries, 1 to 891
Data columns (total 8 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         891 non-null    float64
 4   SibSp       891 non-null    int64
 5   Parch       891 non-null    int64
 6   Fare        891 non-null    float64
 7   Embarked    891 non-null    object
dtypes: float64(2), int64(4), object(2)
memory usage: 94.9+ KB
```

```
[ ]:
```

```
[80]: df2.head()
```

```
[80]:
```

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
PassengerId								
1	0	3	male	22.0	1	0	7.2500	S
2	1	1	female	38.0	1	0	71.2833	C
3	1	3	female	26.0	0	0	7.9250	S
4	1	1	female	35.0	1	0	53.1000	S
5	0	3	male	35.0	0	0	8.0500	S

```
[81]: # univariate analysis : exploring one column at a time
```

```
[82]: # numerical
```

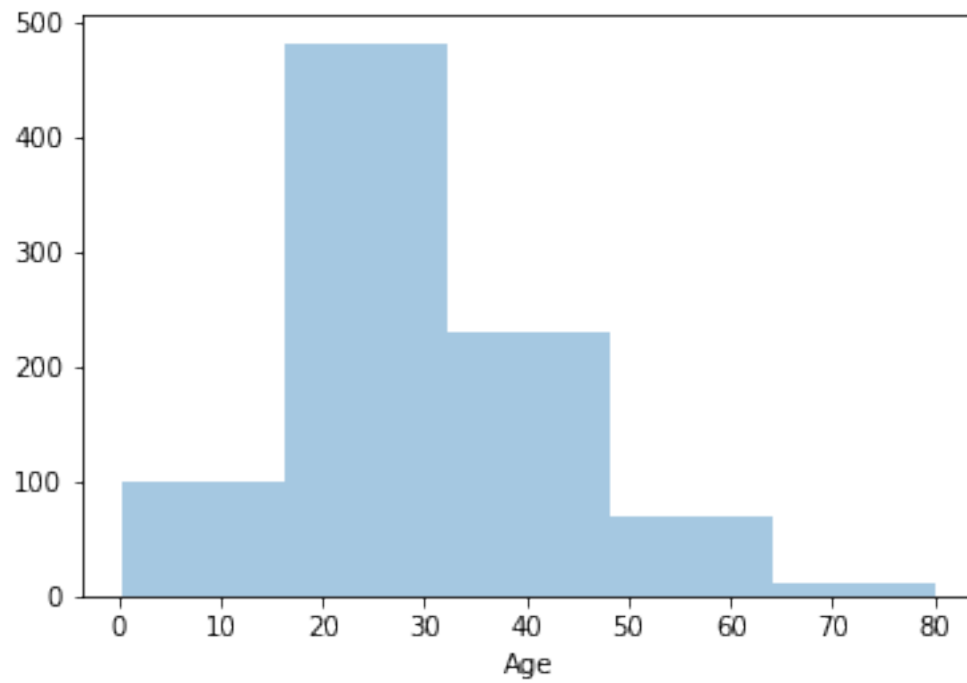
```
# sns.distplot(df.Age, kde = False, bins = 50 )
```

```
sns.distplot(df2.Age, kde = False, bins = 5 )
```

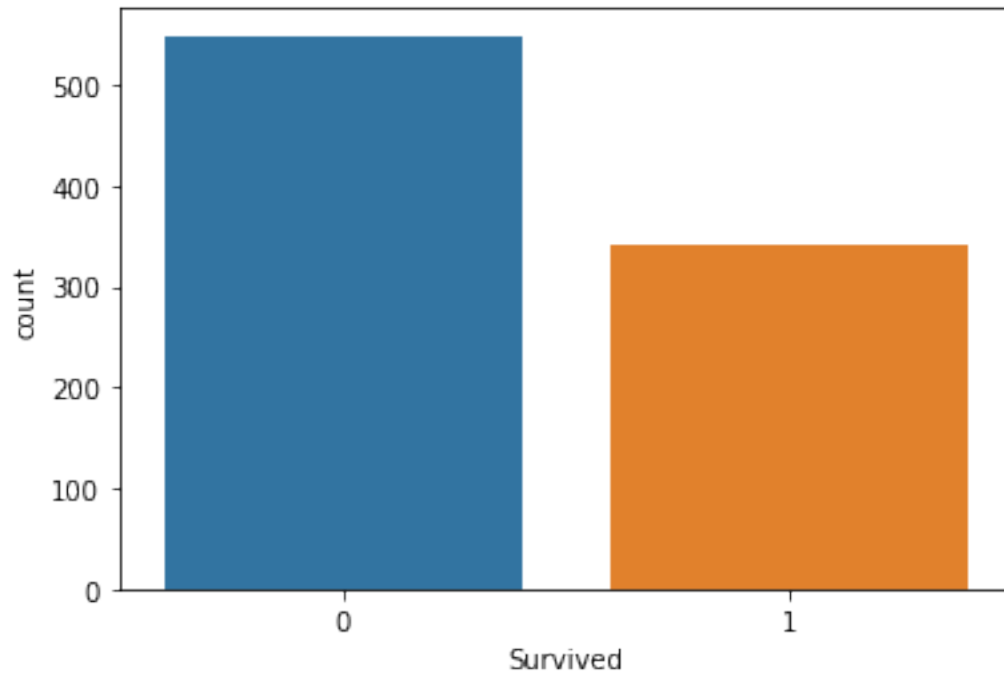
```
d:\installed softwares\python\lib\site-packages\seaborn\distributions.py:2557:
FutureWarning: `distplot` is a deprecated function and will be removed in a
future version. Please adapt your code to use either `displot` (a figure-level
function with similar flexibility) or `histplot` (an axes-level function for
histograms).
```

```
warnings.warn(msg, FutureWarning)
```

```
[82]: <AxesSubplot:xlabel='Age'>
```



```
[83]: # categorical :  
sns.countplot(x = "Survived", data = df2)  
plt.show()  
  
# sns.countplot(x = df2["Survived"])
```

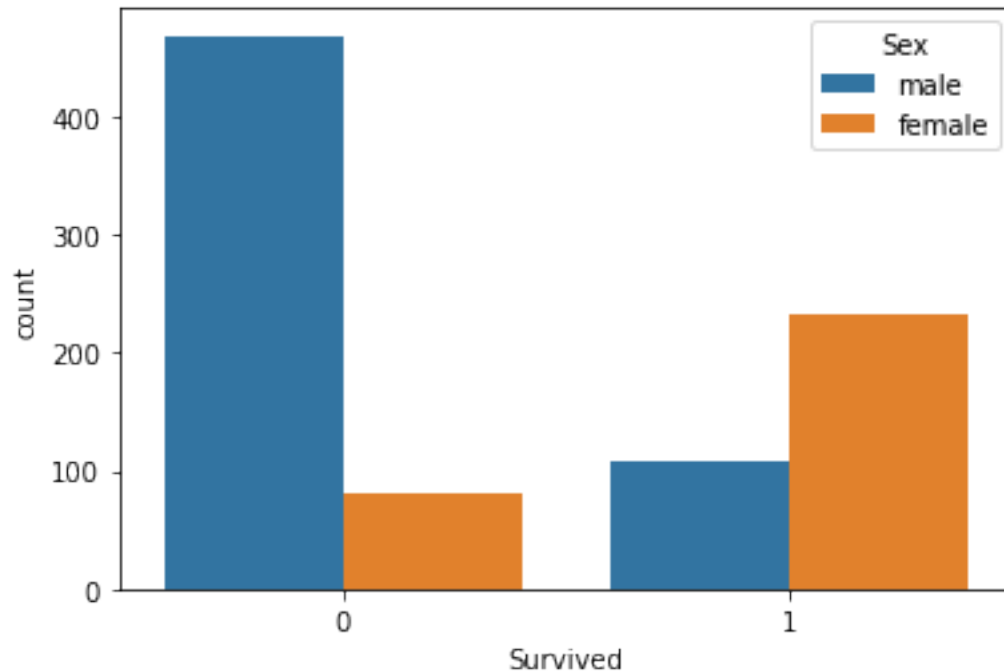


```
[84]: df1.Survived.value_counts()
```

```
[84]: 0    549  
      1    342  
      Name: Survived, dtype: int64
```

```
[85]: #sns.countplot(x = "Survived", data = df2, hue = "Pclass")  
      #plt.show()  
      sns.countplot(x = "Survived", data = df2, hue = "Sex")
```

```
[85]: <AxesSubplot:xlabel='Survived', ylabel='count'>
```



```
[86]: # sns.countplot(x = "Survived", data = df2, hue = "Age")
```

```
[ ]:
```

```
[ ]:
```

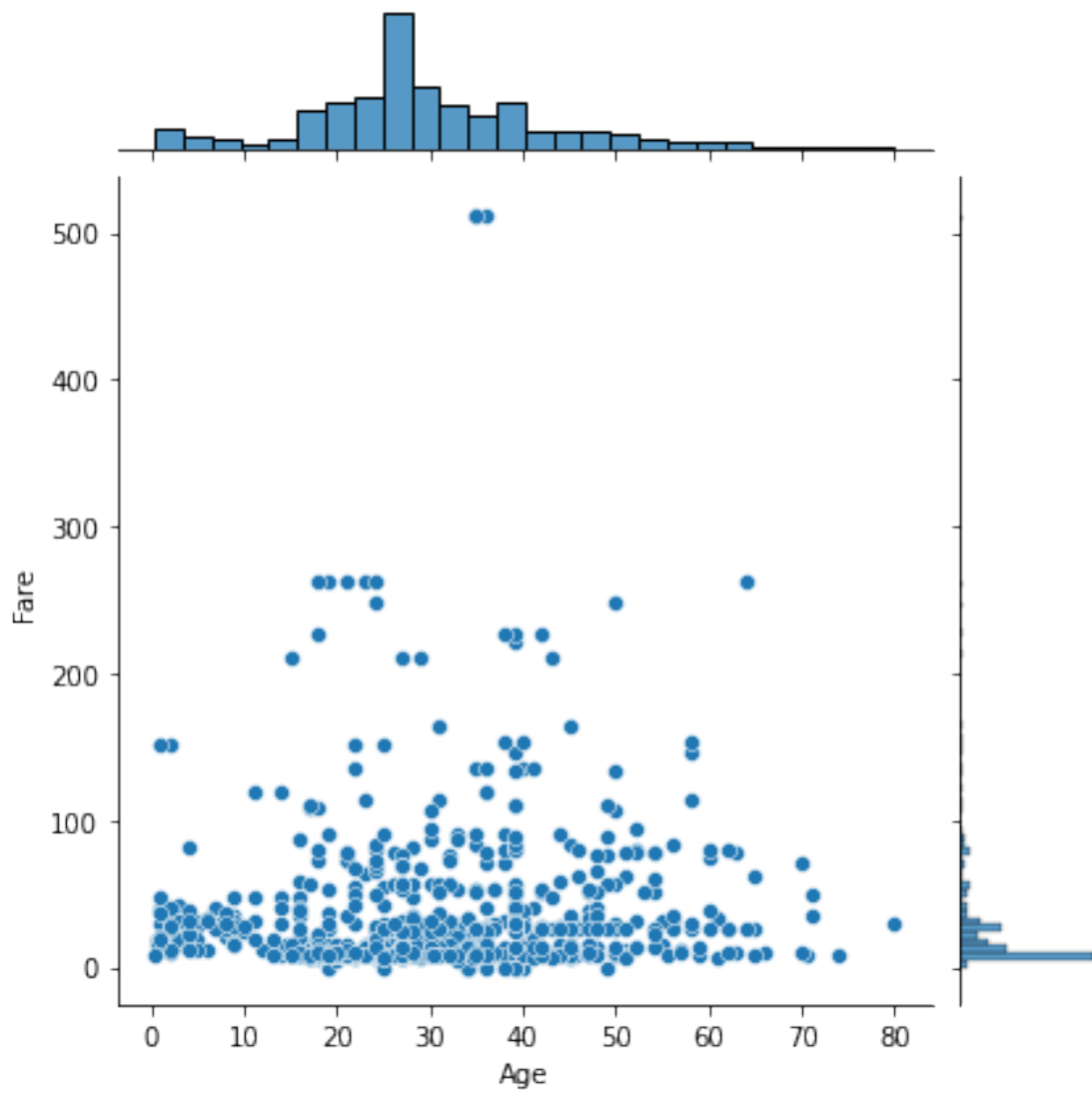
```
[87]: # bivariate analysis - exploring 2 columns at a time
# num vs num, cat vs num, cat vs cat
```

```
[88]: df2.head()
```

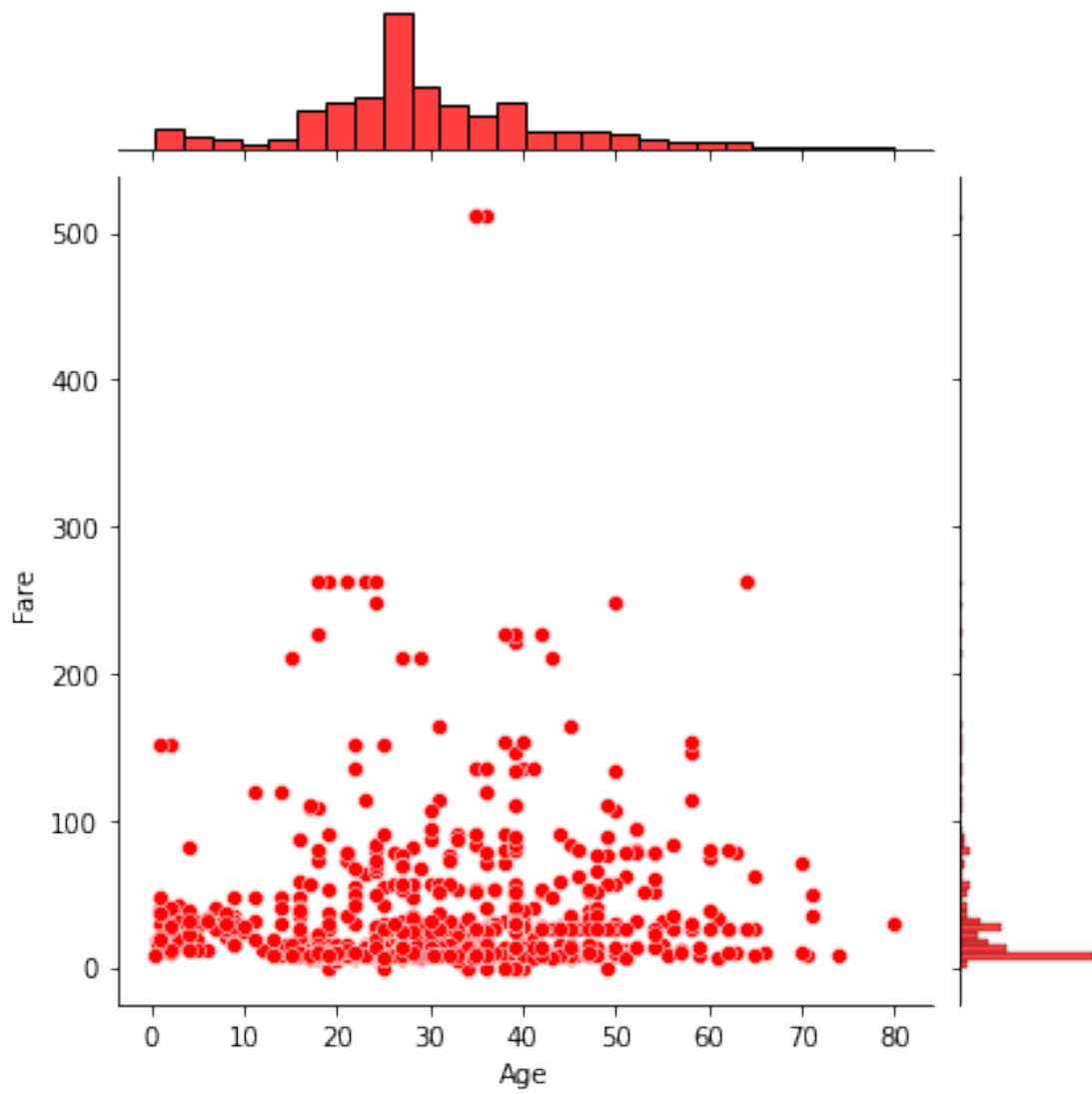
```
[88]:
```

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
PassengerId								
1	0	3	male	22.0	1	0	7.2500	S
2	1	1	female	38.0	1	0	71.2833	C
3	1	3	female	26.0	0	0	7.9250	S
4	1	1	female	35.0	1	0	53.1000	S
5	0	3	male	35.0	0	0	8.0500	S

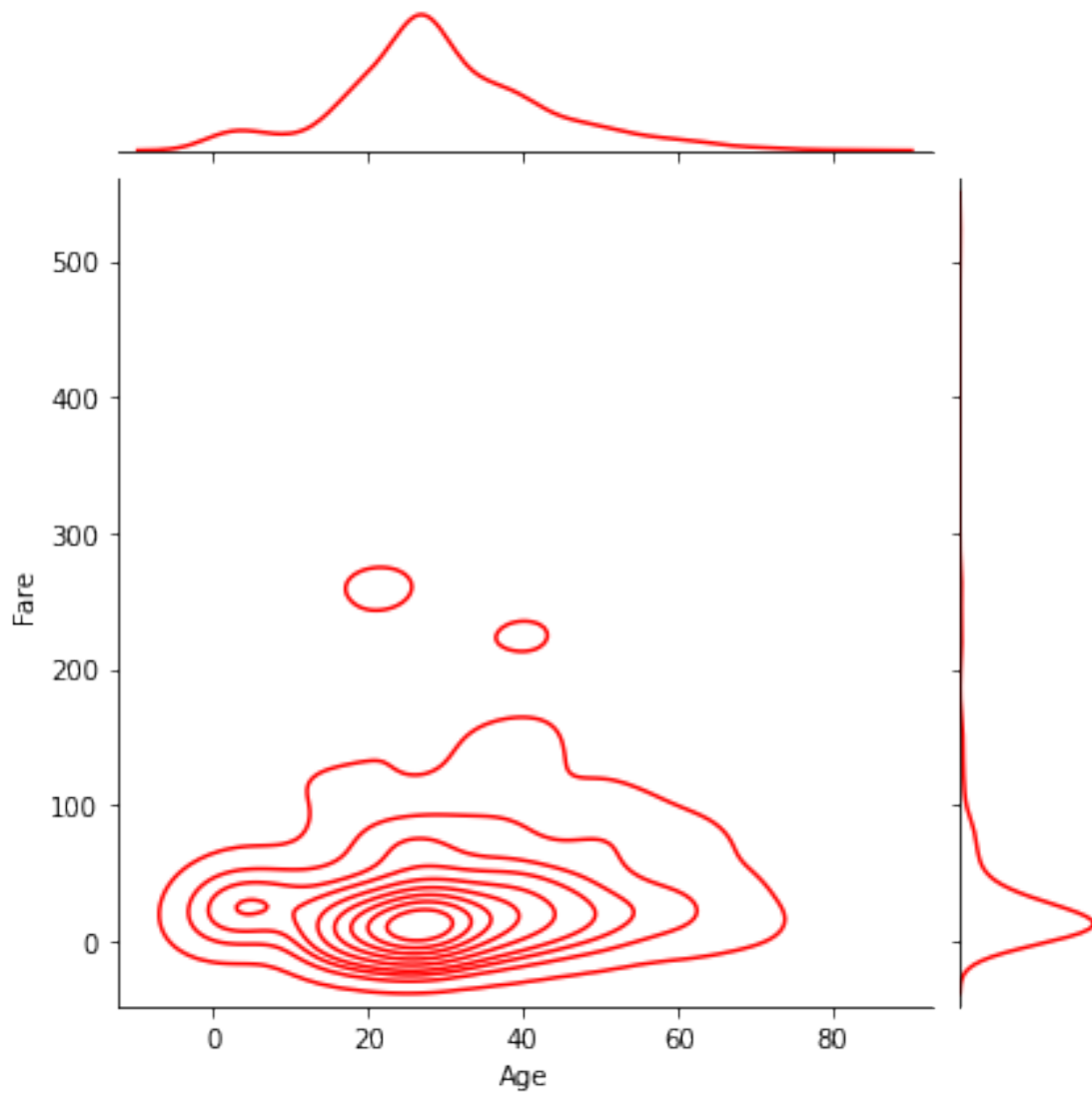
```
[89]: sns.jointplot(x = "Age", y = "Fare", data = df2)
plt.show()
```



```
[90]: sns.jointplot(x = "Age", y = "Fare", data = df2, color = 'red')  
plt.show()
```

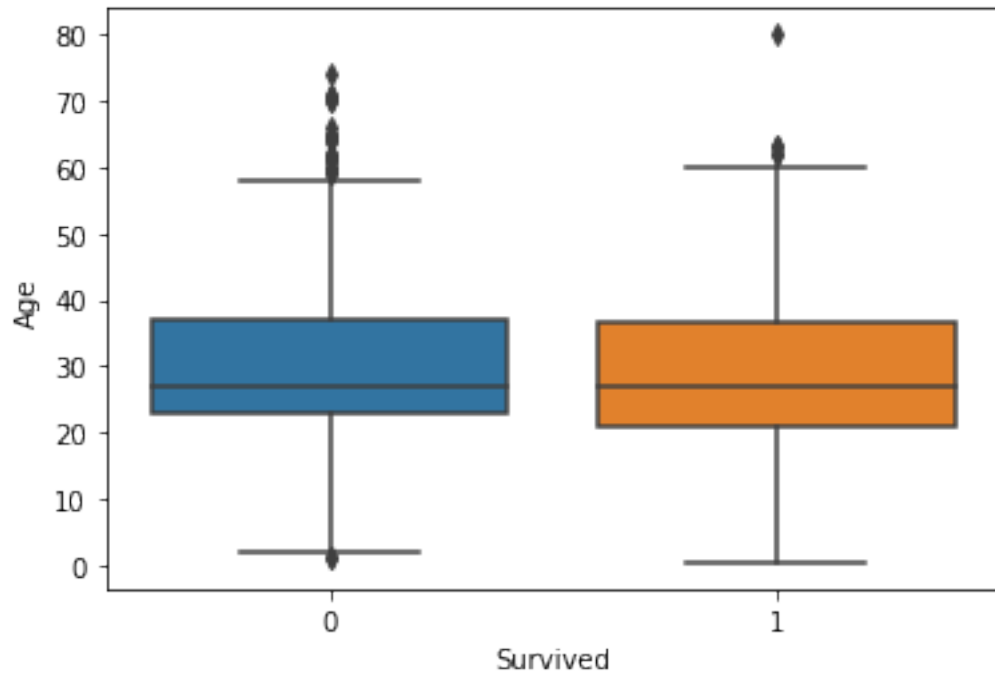



```
[91]: sns.jointplot(x = "Age", y = "Fare", data = df2, color = 'red', kind = "kde")  
plt.show()
```

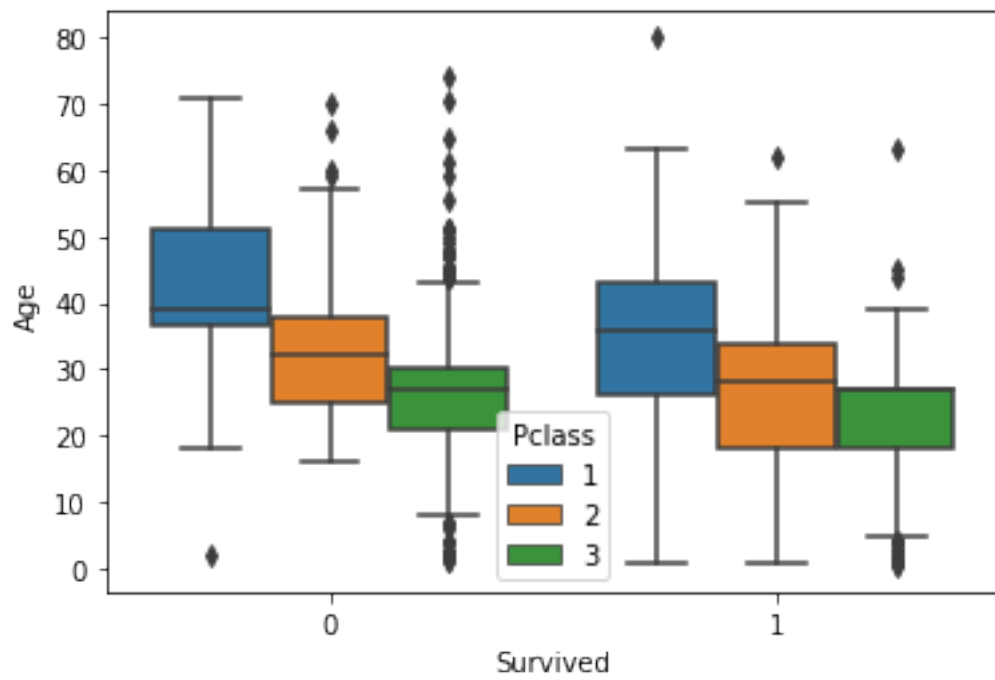


[]:

```
[92]: # cat vs num :  
  
# boxplot, barplot, stripplot, swarmplot, violinplot  
  
[93]: sns.boxplot(x = "Survived", y = "Age", data = df2)  
plt.show()
```



```
[94]: sns.boxplot(x = "Survived", y = "Age", data = df2, hue = "Pclass")  
plt.show()
```



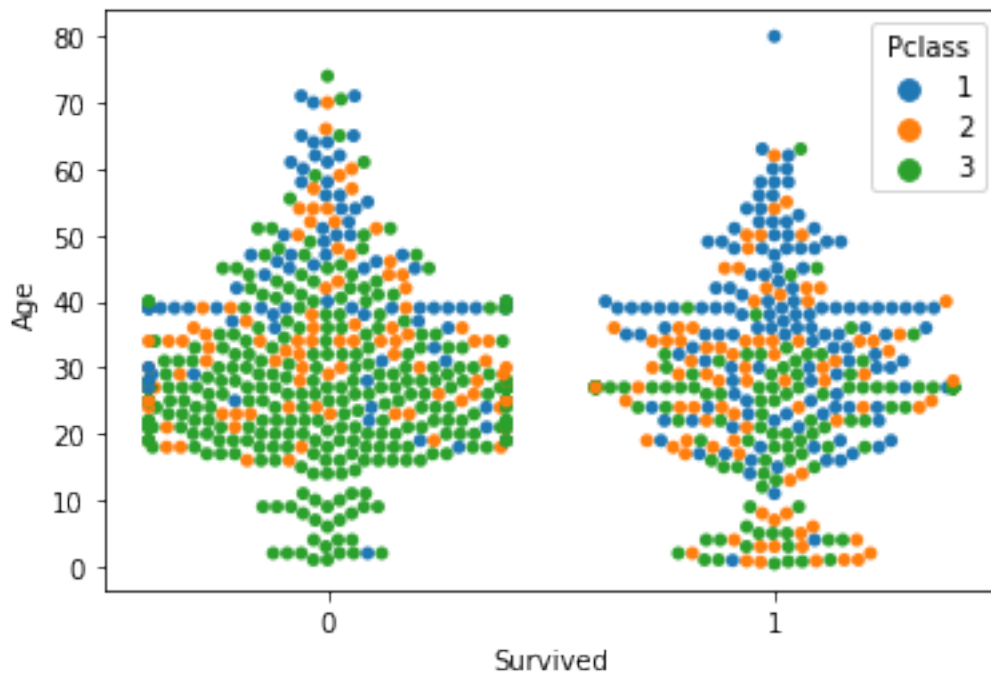
```
[95]: sns.swarmplot(x = "Survived", y = "Age", data = df2, hue = "Pclass")  
plt.show()
```

d:\installed softwares\python\lib\site-packages\seaborn\categorical.py:1296:
UserWarning: 26.4% of the points cannot be placed; you may want to decrease the
size of the markers or use stripplot.

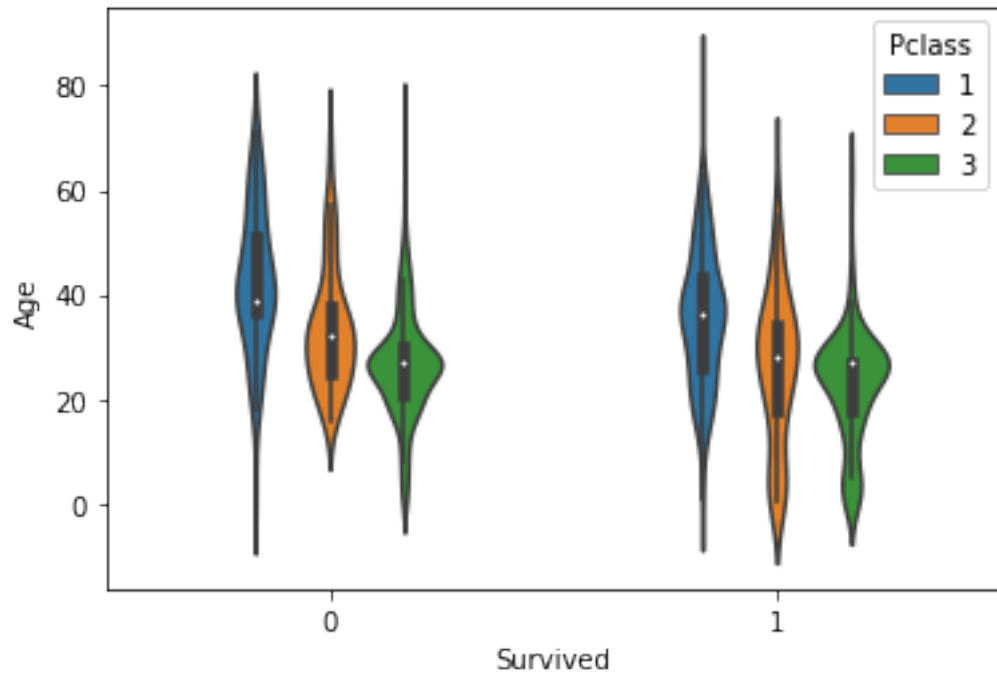
```
warnings.warn(msg, UserWarning)
```

d:\installed softwares\python\lib\site-packages\seaborn\categorical.py:1296:
UserWarning: 7.9% of the points cannot be placed; you may want to decrease the
size of the markers or use stripplot.

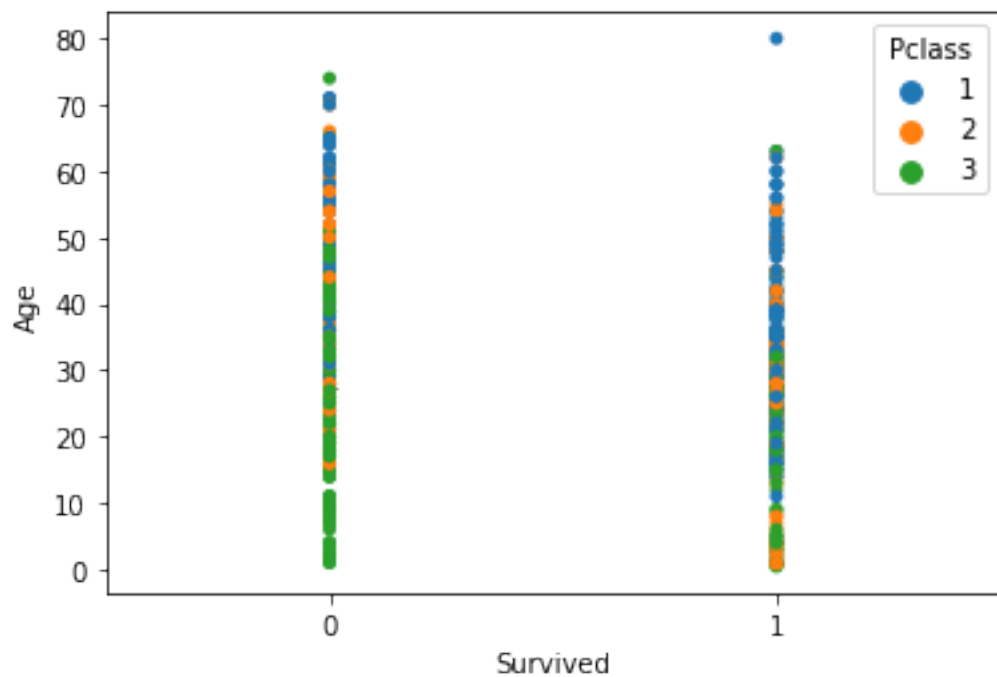
```
warnings.warn(msg, UserWarning)
```



```
[96]: sns.violinplot(x = "Survived", y = "Age", data = df2, hue = "Pclass", width = 0.5)  
plt.show()
```

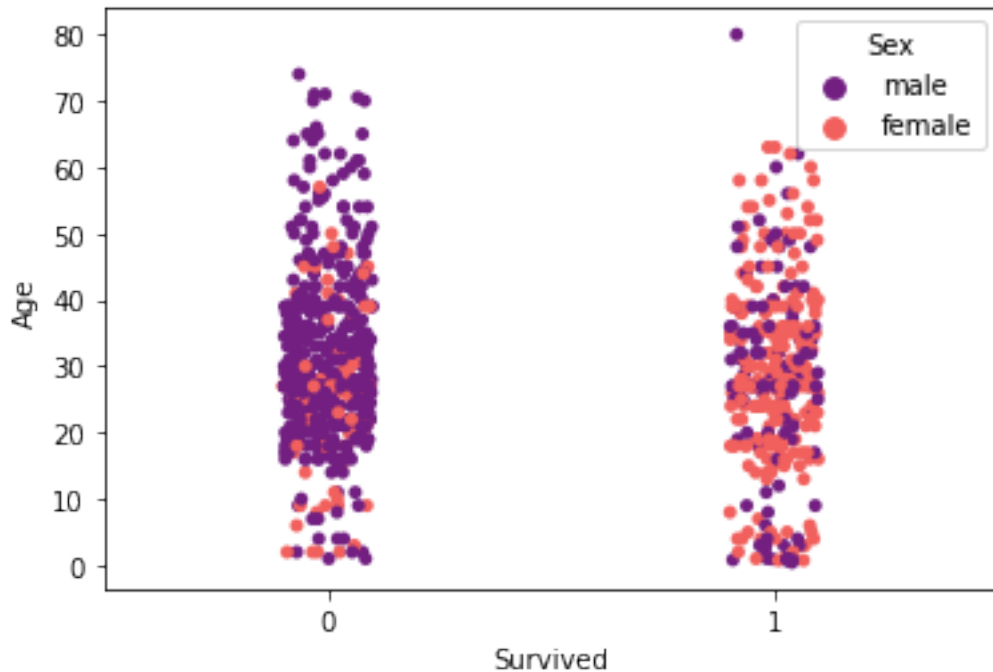


```
[97]: sns.stripplot(x = "Survived", y = "Age", data = df2, hue = "Pclass", jitter = False)
      # sns.color_palette("crest")
      plt.show()
```



```
[98]: sns.stripplot(x = "Survived", y = "Age", data = df2, hue = "Sex", palette = "magma")
```

```
[98]: <AxesSubplot:xlabel='Survived', ylabel='Age'>
```



```
[99]: df2.head()
```

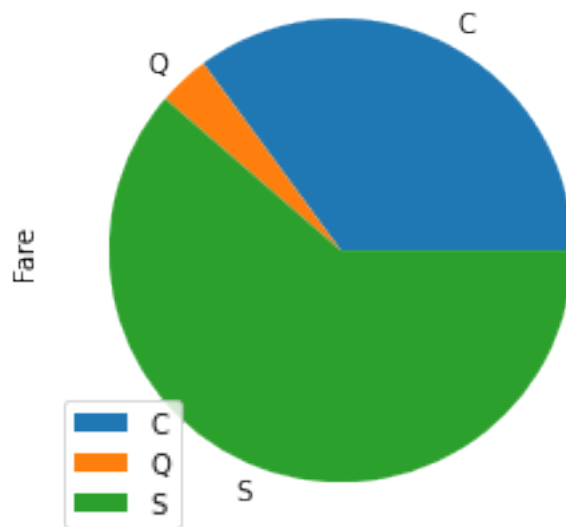
```
[99]:
```

PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
1	0	3	male	22.0	1	0	7.2500	S
2	1	1	female	38.0	1	0	71.2833	C
3	1	3	female	26.0	0	0	7.9250	S
4	1	1	female	35.0	1	0	53.1000	S
5	0	3	male	35.0	0	0	8.0500	S

```
[100]: df2.groupby(['Embarked']).sum()

df2.groupby(['Embarked']).sum().plot(kind = 'pie',y = "Fare")
```

```
[100]: <AxesSubplot:ylabel='Fare'>
```



```
[101]: # plt.pie()

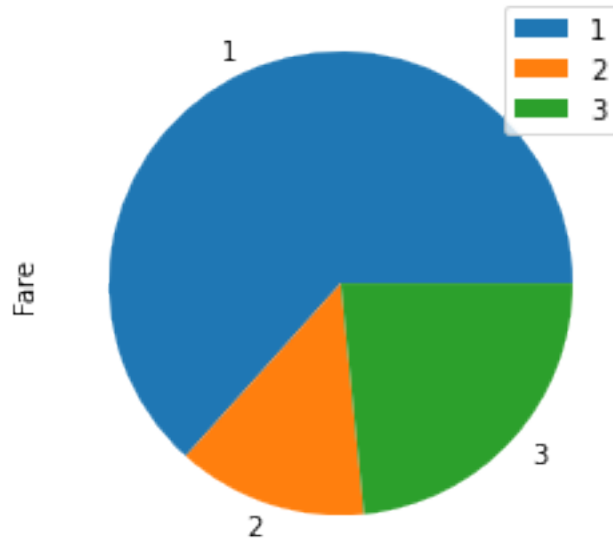
df2.groupby(['Embarked']).sum()
```

```
[101]:
```

	Survived	Pclass	Age	SibSp	Parch	Fare
Embarked						
C	93	317	5177.92	65	61	10072.2962
Q	30	224	2116.50	33	13	1022.2543
S	219	1516	19126.75	368	266	17599.3988

```
[102]: df2.groupby(['Pclass']).sum().plot(kind = 'pie',y = "Fare")
```

```
[102]: <AxesSubplot:ylabel='Fare'>
```



```
[103]: df2.groupby(['Pclass']).sum()
```

```
[103]:
```

	Survived	Age	SibSp	Parch	Fare
Pclass					
1	136	8281.42	90	77	18177.4125
2	87	5542.83	74	70	3801.8417
3	119	12596.92	302	193	6714.6951

```
[ ]:
```

```
[104]: # cat to num
# concat...
# joins, merge
```

```
[105]: df = df2.copy()
```

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

```
[106]:
```

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
PassengerId								
1	0	3	male	22.0	1	0	7.2500	S
2	1	1	female	38.0	1	0	71.2833	C
3	1	3	female	26.0	0	0	7.9250	S
4	1	1	female	35.0	1	0	53.1000	S
5	0	3	male	35.0	0	0	8.0500	S


```
[107]: # labelencoder, onehotencoder
# pandas get dummies
```

```
[108]: s_ = pd.get_dummies(df.Sex, drop_first=True)
s_.head()
```

```
[108]:
```

	male
PassengerId	
1	1
2	0
3	0
4	0
5	1

```
[109]: e_ = pd.get_dummies(df.Embarked, drop_first=True)
e_.head()
```

```
[109]:
```

	Q	S
PassengerId		
1	0	1
2	0	0
3	0	1
4	0	1
5	0	1

```
[110]: df = pd.concat([df, s_, e_], axis = 1)
```

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

```
[111]:
```

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked	\
PassengerId									
1	0	3	male	22.0	1	0	7.2500	S	
2	1	1	female	38.0	1	0	71.2833	C	
3	1	3	female	26.0	0	0	7.9250	S	
4	1	1	female	35.0	1	0	53.1000	S	
5	0	3	male	35.0	0	0	8.0500	S	


```
[111]:
```

	male	Q	S
PassengerId			
1	1	0	1
2	0	0	0
3	0	0	1
4	0	0	1
5	1	0	1

```
[112]: df.drop(['Sex', "Embarked"], axis = 1, inplace = True)
```

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

```
[113]:
```

	Survived	Pclass	Age	SibSp	Parch	Fare	male	Q	S
PassengerId									
1	0	3	22.0	1	0	7.2500	1	0	1
2	1	1	38.0	1	0	71.2833	0	0	0
3	1	3	26.0	0	0	7.9250	0	0	1
4	1	1	35.0	1	0	53.1000	0	0	1
5	0	3	35.0	0	0	8.0500	1	0	1

```
[114]: df2.head()
```

```
[114]:
```

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
PassengerId								
1	0	3	male	22.0	1	0	7.2500	S
2	1	1	female	38.0	1	0	71.2833	C
3	1	3	female	26.0	0	0	7.9250	S
4	1	1	female	35.0	1	0	53.1000	S
5	0	3	male	35.0	0	0	8.0500	S

```
[115]: df2['S1'] = df2.Sex.replace(to_replace = ["male", 'female'], value = [0, 1])
```

```
[116]: df2.head()
```

```
[116]:
```

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked	\
PassengerId									
1	0	3	male	22.0	1	0	7.2500	S	
2	1	1	female	38.0	1	0	71.2833	C	
3	1	3	female	26.0	0	0	7.9250	S	
4	1	1	female	35.0	1	0	53.1000	S	
5	0	3	male	35.0	0	0	8.0500	S	

```

S1
PassengerId
1          0
2          1
3          1
4          1
5          0
```

```
[ ]:
```

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

```
[117]:
```

	Survived	Pclass	Age	SibSp	Parch	Fare	male	Q	S
PassengerId									
1	0	3	22.0	1	0	7.2500	1	0	1

2	1	1	38.0	1	0	71.2833	0	0	0
3	1	3	26.0	0	0	7.9250	0	0	1
4	1	1	35.0	1	0	53.1000	0	0	1
5	0	3	35.0	0	0	8.0500	1	0	1

```
[118]: df['Family'] = df['SibSp'] + df['Parch']
```

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

```
[119]:
```

	Survived	Pclass	Age	SibSp	Parch	Fare	male	Q	S	Family
PassengerId										
1	0	3	22.0	1	0	7.2500	1	0	1	1
2	1	1	38.0	1	0	71.2833	0	0	0	1
3	1	3	26.0	0	0	7.9250	0	0	1	0
4	1	1	35.0	1	0	53.1000	0	0	1	1
5	0	3	35.0	0	0	8.0500	1	0	1	0

```
[120]: df.drop(['SibSp', "Parch"], axis = 1, inplace = True)
```

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

```
[121]:
```

	Survived	Pclass	Age	Fare	male	Q	S	Family
PassengerId								
1	0	3	22.0	7.2500	1	0	1	1
2	1	1	38.0	71.2833	0	0	0	1
3	1	3	26.0	7.9250	0	0	1	0
4	1	1	35.0	53.1000	0	0	1	1
5	0	3	35.0	8.0500	1	0	1	0

```
[122]: df.shape
```

```
[122]: (891, 8)
```

```
[123]: df.drop_duplicates(inplace = True) # Pandas drop_duplicates() method helps
↳ in removing duplicates from the data frame.
```

```
[124]: df.shape
```

```
[124]: (776, 8)
```