

Data Preprocessing on Titanic Dataset

1.Import the Libraries.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

2.Importing the dataset.

```
df=pd.read_csv("Titanic-Dataset.csv")
df
```

	PassengerId	Survived	Pclass	\
0	1	0	3	
1	2	1	1	
2	3	1	3	
3	4	1	1	
4	5	0	3	
..	
886	887	0	2	
887	888	1	1	
888	889	0	3	
889	890	1	1	
890	891	0	3	

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

```

889                                Behr, Mr. Karl Howell    male  26.0
0
890                                Dooley, Mr. Patrick     male  32.0
0

```

```

      Parch      Ticket    Fare Cabin Embarked
0         0         A/5 21171    7.2500   NaN      S
1         0         PC 17599   71.2833   C85      C
2         0  STON/O2. 3101282    7.9250   NaN      S
3         0         113803   53.1000  C123      S
4         0         373450    8.0500   NaN      S
..      ...
886        0         211536   13.0000   NaN      S
887        0         112053   30.0000   B42      S
888        2         W./C. 6607   23.4500   NaN      S
889        0         111369   30.0000  C148      C
890        0         370376    7.7500   NaN      Q

```

```
[891 rows x 12 columns]
```

```
df.head()
```

```

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

```

```

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

```

```

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

```

```
df.describe()
```

	PassengerId	Survived	Pclass	Age	SibSp \
count	891.000000	891.000000	891.000000	714.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008
std	257.353842	0.486592	0.836071	14.526497	1.102743
min	1.000000	0.000000	1.000000	0.420000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000
50%	446.000000	0.000000	3.000000	28.000000	0.000000
75%	668.500000	1.000000	3.000000	38.000000	1.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000

	Parch	Fare
count	891.000000	891.000000
mean	0.381594	32.204208
std	0.806057	49.693429
min	0.000000	0.000000
25%	0.000000	7.910400
50%	0.000000	14.454200
75%	0.000000	31.000000
max	6.000000	512.329200

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
#   Column          Non-Null Count  Dtype
---  -
0   PassengerId     891 non-null   int64
1   Survived        891 non-null   int64
2   Pclass          891 non-null   int64
3   Name            891 non-null   object
4   Sex             891 non-null   object
5   Age             714 non-null   float64
6   SibSp           891 non-null   int64
7   Parch           891 non-null   int64
8   Ticket          891 non-null   object
9   Fare            891 non-null   float64
10  Cabin           204 non-null   object
11  Embarked        889 non-null   object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

```
df.corr()
```

```
C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\1134722465.py:1:
FutureWarning: The default value of numeric_only in DataFrame.corr is
deprecated. In a future version, it will default to False. Select only
valid columns or specify the value of numeric_only to silence this
warning.
df.corr()
```

	PassengerId	Survived	Pclass	Age	SibSp
Parch \ PassengerId	1.000000	-0.005007	-0.035144	0.036847	-0.057527
0.001652					
Survived	-0.005007	1.000000	-0.338481	-0.077221	-0.035322
0.081629					
Pclass	-0.035144	-0.338481	1.000000	-0.369226	0.083081
0.018443					
Age	0.036847	-0.077221	-0.369226	1.000000	-0.308247
0.189119					
SibSp	-0.057527	-0.035322	0.083081	-0.308247	1.000000
0.414838					
Parch	-0.001652	0.081629	0.018443	-0.189119	0.414838
1.000000					
Fare	0.012658	0.257307	-0.549500	0.096067	0.159651
0.216225					

	Fare
PassengerId	0.012658
Survived	0.257307
Pclass	-0.549500
Age	0.096067
SibSp	0.159651
Parch	0.216225
Fare	1.000000

```
df.corr().Survived.sort_values(ascending=False)
```

C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\1717937034.py:1:
FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will default to False. Select only valid columns or specify the value of numeric_only to silence this warning.

```
df.corr().Survived.sort_values(ascending=False)
```

Survived	1.000000
Fare	0.257307
Parch	0.081629
PassengerId	-0.005007
SibSp	-0.035322
Age	-0.077221
Pclass	-0.338481

Name: Survived, dtype: float64

```
df.corr().PassengerId.sort_values(ascending=False)
```

C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\1990250427.py:1:
FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will default to False. Select only valid columns or specify the value of numeric_only to silence this

```
warning.  
df.corr().PassengerId.sort_values(ascending=False)
```

```
PassengerId    1.000000  
Age            0.036847  
Fare           0.012658  
Parch         -0.001652  
Survived       -0.005007  
Pclass        -0.035144  
SibSp         -0.057527  
Name: PassengerId, dtype: float64
```

```
df.corr().Pclass.sort_values(ascending=False)
```

```
C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\2321911039.py:1:  
FutureWarning: The default value of numeric_only in DataFrame.corr is  
deprecated. In a future version, it will default to False. Select only  
valid columns or specify the value of numeric_only to silence this  
warning.
```

```
df.corr().Pclass.sort_values(ascending=False)
```

```
Pclass         1.000000  
SibSp          0.083081  
Parch          0.018443  
PassengerId   -0.035144  
Survived      -0.338481  
Age           -0.369226  
Fare          -0.549500  
Name: Pclass, dtype: float64
```

```
df.corr().Age.sort_values(ascending=False)
```

```
C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\2477542723.py:1:  
FutureWarning: The default value of numeric_only in DataFrame.corr is  
deprecated. In a future version, it will default to False. Select only  
valid columns or specify the value of numeric_only to silence this  
warning.
```

```
df.corr().Age.sort_values(ascending=False)
```

```
Age            1.000000  
Fare           0.096067  
PassengerId    0.036847  
Survived      -0.077221  
Parch         -0.189119  
SibSp         -0.308247  
Pclass        -0.369226  
Name: Age, dtype: float64
```

```
df.corr().SibSp.sort_values(ascending=False)
```

```
C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\410142966.py:1:
FutureWarning: The default value of numeric_only in DataFrame.corr is
deprecated. In a future version, it will default to False. Select only
valid columns or specify the value of numeric_only to silence this
warning.
```

```
df.corr().SibSp.sort_values(ascending=False)
```

```
SibSp      1.000000
Parch      0.414838
Fare       0.159651
Pclass     0.083081
Survived   -0.035322
PassengerId -0.057527
Age        -0.308247
Name: SibSp, dtype: float64
```

```
df.corr().Parch.sort_values(ascending=False)
```

```
C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\1925834096.py:1:
FutureWarning: The default value of numeric_only in DataFrame.corr is
deprecated. In a future version, it will default to False. Select only
valid columns or specify the value of numeric_only to silence this
warning.
```

```
df.corr().Parch.sort_values(ascending=False)
```

```
Parch      1.000000
SibSp      0.414838
Fare       0.216225
Survived    0.081629
Pclass     0.018443
PassengerId -0.001652
Age        -0.189119
Name: Parch, dtype: float64
```

```
df.corr().Fare.sort_values(ascending=False)
```

```
C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\60082530.py:1:
FutureWarning: The default value of numeric_only in DataFrame.corr is
deprecated. In a future version, it will default to False. Select only
valid columns or specify the value of numeric_only to silence this
warning.
```

```
df.corr().Fare.sort_values(ascending=False)
```

```
Fare      1.000000
Survived  0.257307
Parch     0.216225
SibSp     0.159651
Age       0.096067
PassengerId 0.012658
Pclass    -0.549500
Name: Fare, dtype: float64
```

3. Checking for Null Values.

```
df.isnull().any()
```

PassengerId	False
Survived	False
Pclass	False
Name	False
Sex	False
Age	True
SibSp	False
Parch	False
Ticket	False
Fare	False
Cabin	True
Embarked	True

dtype: bool

```
df.isnull().sum()
```

PassengerId	0
Survived	0
Pclass	0
Name	0
Sex	0
Age	177
SibSp	0
Parch	0
Ticket	0
Fare	0
Cabin	687
Embarked	2

dtype: int64

```
df.Age.nunique()
```

88

```
df.PassengerId.nunique()
```

891

```
df.Survived.nunique()
```

2

```
df.Pclass.nunique()
```

3

```
df.Name.nunique()
```

891

```
df.SibSp.nunique()
```

7

```
df.Parch.nunique()
```

7

```
df.Fare.nunique()
```

248

```
df.Cabin.nunique()
```

147

```
df.Embarked.nunique()
```

3

```
df.Sex.nunique()
```

2

```
df.Ticket.nunique()
```

681

```
df.Age.unique()
```

```
array([22. , 38. , 26. , 35. , nan, 54. , 2. , 27. , 14. ,
       4. , 58. , 20. , 39. , 55. , 31. , 34. , 15. , 28. ,
       8. , 19. , 40. , 66. , 42. , 21. , 18. , 3. , 7. ,
       49. , 29. , 65. , 28.5, 5. , 11. , 45. , 17. , 32. ,
       16. , 25. , 0.83, 30. , 33. , 23. , 24. , 46. , 59. ,
       71. , 37. , 47. , 14.5, 70.5, 32.5, 12. , 9. , 36.5 ,
       51. , 55.5, 40.5 , 44. , 1. , 61. , 56. , 50. , 36. ,
       45.5 , 20.5 , 62. , 41. , 52. , 63. , 23.5 , 0.92, 43. ,
       60. , 10. , 64. , 13. , 48. , 0.75, 53. , 57. , 80. ,
       70. , 24.5 , 6. , 0.67, 30.5 , 0.42, 34.5 , 74. ])
```

```
df.PassengerId.unique()
```

```
array([ 1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12,
       13,
       14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25,
       26,
       27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38,
       39,
       40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51,
       52,
```


65,	53,	54,	55,	56,	57,	58,	59,	60,	61,	62,	63,	64,
78,	66,	67,	68,	69,	70,	71,	72,	73,	74,	75,	76,	77,
91,	79,	80,	81,	82,	83,	84,	85,	86,	87,	88,	89,	90,
104,	92,	93,	94,	95,	96,	97,	98,	99,	100,	101,	102,	103,
117,	105,	106,	107,	108,	109,	110,	111,	112,	113,	114,	115,	116,
130,	118,	119,	120,	121,	122,	123,	124,	125,	126,	127,	128,	129,
143,	131,	132,	133,	134,	135,	136,	137,	138,	139,	140,	141,	142,
156,	144,	145,	146,	147,	148,	149,	150,	151,	152,	153,	154,	155,
169,	157,	158,	159,	160,	161,	162,	163,	164,	165,	166,	167,	168,
182,	170,	171,	172,	173,	174,	175,	176,	177,	178,	179,	180,	181,
195,	183,	184,	185,	186,	187,	188,	189,	190,	191,	192,	193,	194,
208,	196,	197,	198,	199,	200,	201,	202,	203,	204,	205,	206,	207,
221,	209,	210,	211,	212,	213,	214,	215,	216,	217,	218,	219,	220,
234,	222,	223,	224,	225,	226,	227,	228,	229,	230,	231,	232,	233,
247,	235,	236,	237,	238,	239,	240,	241,	242,	243,	244,	245,	246,
260,	248,	249,	250,	251,	252,	253,	254,	255,	256,	257,	258,	259,
273,	261,	262,	263,	264,	265,	266,	267,	268,	269,	270,	271,	272,
286,	274,	275,	276,	277,	278,	279,	280,	281,	282,	283,	284,	285,
299,	287,	288,	289,	290,	291,	292,	293,	294,	295,	296,	297,	298,
312,	300,	301,	302,	303,	304,	305,	306,	307,	308,	309,	310,	311,
325,	313,	314,	315,	316,	317,	318,	319,	320,	321,	322,	323,	324,
338,	326,	327,	328,	329,	330,	331,	332,	333,	334,	335,	336,	337,
351,	339,	340,	341,	342,	343,	344,	345,	346,	347,	348,	349,	350,
364,	352,	353,	354,	355,	356,	357,	358,	359,	360,	361,	362,	363,
	365,	366,	367,	368,	369,	370,	371,	372,	373,	374,	375,	376,

377,
378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 388, 389,
390,
391, 392, 393, 394, 395, 396, 397, 398, 399, 400, 401, 402,
403,
404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 414, 415,
416,
417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428,
429,
430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 441,
442,
443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454,
455,
456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467,
468,
469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480,
481,
482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493,
494,
495, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506,
507,
508, 509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519,
520,
521, 522, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532,
533,
534, 535, 536, 537, 538, 539, 540, 541, 542, 543, 544, 545,
546,
547, 548, 549, 550, 551, 552, 553, 554, 555, 556, 557, 558,
559,
560, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 571,
572,
573, 574, 575, 576, 577, 578, 579, 580, 581, 582, 583, 584,
585,
586, 587, 588, 589, 590, 591, 592, 593, 594, 595, 596, 597,
598,
599, 600, 601, 602, 603, 604, 605, 606, 607, 608, 609, 610,
611,
612, 613, 614, 615, 616, 617, 618, 619, 620, 621, 622, 623,
624,
625, 626, 627, 628, 629, 630, 631, 632, 633, 634, 635, 636,
637,
638, 639, 640, 641, 642, 643, 644, 645, 646, 647, 648, 649,
650,
651, 652, 653, 654, 655, 656, 657, 658, 659, 660, 661, 662,
663,
664, 665, 666, 667, 668, 669, 670, 671, 672, 673, 674, 675,
676,
677, 678, 679, 680, 681, 682, 683, 684, 685, 686, 687, 688,
689,

```

690, 691, 692, 693, 694, 695, 696, 697, 698, 699, 700, 701,
702,
703, 704, 705, 706, 707, 708, 709, 710, 711, 712, 713, 714,
715,
716, 717, 718, 719, 720, 721, 722, 723, 724, 725, 726, 727,
728,
729, 730, 731, 732, 733, 734, 735, 736, 737, 738, 739, 740,
741,
742, 743, 744, 745, 746, 747, 748, 749, 750, 751, 752, 753,
754,
755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 765, 766,
767,
768, 769, 770, 771, 772, 773, 774, 775, 776, 777, 778, 779,
780,
781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791, 792,
793,
794, 795, 796, 797, 798, 799, 800, 801, 802, 803, 804, 805,
806,
807, 808, 809, 810, 811, 812, 813, 814, 815, 816, 817, 818,
819,
820, 821, 822, 823, 824, 825, 826, 827, 828, 829, 830, 831,
832,
833, 834, 835, 836, 837, 838, 839, 840, 841, 842, 843, 844,
845,
846, 847, 848, 849, 850, 851, 852, 853, 854, 855, 856, 857,
858,
859, 860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870,
871,
872, 873, 874, 875, 876, 877, 878, 879, 880, 881, 882, 883,
884,
885, 886, 887, 888, 889, 890, 891], dtype=int64)

df.Survived.unique()
array([0, 1], dtype=int64)

df.Pclass.unique()
array([3, 1, 2], dtype=int64)

df.Name.unique()
array(['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',

```

Johan', 'Saundercock, Mr. William Henry', 'Andersson, Mr. Anders',
 '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 Fernandeo',
 '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',
 'Lefebvre, 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',
 'Lindhahl, 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 Pelsmaeker, 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',
 'Pettersson, 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 Lemer cier',

'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. Lallo',
 'Herman, Mrs. Samuel (Jane Laver)', 'Hamalainen, Master.
 Viljo',
 'Carlsson, Mr. August Sigfrid', 'Bailey, Mr. Percy Andrew',
 'Theobald, Mr. Thomas Leonard',
 'Rothes, 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',
        'Roebbing, 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'], dtype=object)

```

```
df.SibSp.unique()
```

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

```
df.Parch.unique()
```

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

```
df.Fare.unique()
```

```
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])

```

```
df.Cabin.unique()
```

```

array([nan, 'C85', 'C123', 'E46', 'G6', 'C103', 'D56', 'A6',
       'C23 C25 C27', 'B78', 'D33', 'B30', 'C52', 'B28', 'C83', 'F33',
       'F G73', 'E31', 'A5', 'D10 D12', 'D26', 'C110', 'B58 B60',
       'E101',
       'F E69', 'D47', 'B86', 'F2', 'C2', 'E33', 'B19', 'A7', 'C49',
       'F4',
       'A32', 'B4', 'B80', 'A31', 'D36', 'D15', 'C93', 'C78', 'D35',

```



```

        'C87', 'B77', 'E67', 'B94', 'C125', 'C99', 'C118', 'D7', 'A19',
        'B49', 'D', 'C22 C26', 'C106', 'C65', 'E36', 'C54',
        'B57 B59 B63 B66', 'C7', 'E34', 'C32', 'B18', 'C124', 'C91',
'E40',
        'T', 'C128', 'D37', 'B35', 'E50', 'C82', 'B96 B98', 'E10',
'E44',
        'A34', 'C104', 'C111', 'C92', 'E38', 'D21', 'E12', 'E63',
'A14',
        'B37', 'C30', 'D20', 'B79', 'E25', 'D46', 'B73', 'C95', 'B38',
        'B39', 'B22', 'C86', 'C70', 'A16', 'C101', 'C68', 'A10', 'E68',
        'B41', 'A20', 'D19', 'D50', 'D9', 'A23', 'B50', 'A26', 'D48',
        'E58', 'C126', 'B71', 'B51 B53 B55', 'D49', 'B5', 'B20', 'F
G63',
        'C62 C64', 'E24', 'C90', 'C45', 'E8', 'B101', 'D45', 'C46',
'D30',
        'E121', 'D11', 'E77', 'F38', 'B3', 'D6', 'B82 B84', 'D17',
'A36',
        'B102', 'B69', 'E49', 'C47', 'D28', 'E17', 'A24', 'C50', 'B42',
        'C148'], dtype=object)

```

```
df.Embarked.unique()
```

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

```
df.Sex.unique()
```

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

```
df.Ticket.unique()
```

```

array(['A/5 21171', 'PC 17599', 'STON/02. 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',
        'S0/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', 'SC0/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'], dtype=object)

df.Age.value_counts()

```

24.00	30
22.00	27
18.00	26
19.00	25
28.00	25

	..
36.50	1
55.50	1
0.92	1
23.50	1
74.00	1

Name: Age, Length: 88, dtype: int64

df.PassengerId.value_counts()

1	1
599	1
588	1
589	1
590	1

	..
301	1
302	1
303	1
304	1
891	1

Name: PassengerId, Length: 891, dtype: int64

df.Survived.value_counts()

0	549
1	342

Name: Survived, dtype: int64

df.Fare.value_counts()

8.0500	43
13.0000	42
7.8958	38
7.7500	34
26.0000	31

	..
35.0000	1
28.5000	1
6.2375	1
14.0000	1
10.5167	1

Name: Fare, Length: 248, dtype: int64

df.Parch.value_counts()

```

0      678
1      118
2       80
5        5
3        5
4        4
6        1
Name: Parch, dtype: int64

df.Pclass.value_counts()

3      491
1      216
2      184
Name: Pclass, dtype: int64

df.Embarked.value_counts()

S      644
C      168
Q       77
Name: Embarked, dtype: int64

df.Ticket.value_counts()

347082      7
CA. 2343     7
1601        7
3101295     6
CA 2144     6
..
9234        1
19988       1
2693        1
PC 17612    1
370376      1
Name: Ticket, Length: 681, dtype: int64

df.Name.value_counts()

Braund, Mr. Owen Harris      1
Boulos, Mr. Hanna           1
Frolicher-Stehli, Mr. Maxmillian  1
Gilinski, Mr. Eliezer        1
Murdlin, Mr. Joseph          1
..
Kelly, Miss. Anna Katherine "Annie Kate"  1
McCoy, Mr. Bernard           1
Johnson, Mr. William Cahoone Jr  1
Keane, Miss. Nora A          1

```

```
Dooley, Mr. Patrick 1
Name: Name, Length: 891, dtype: int64
```

```
df.Sex.value_counts()
```

```
male    577
female  314
Name: Sex, dtype: int64
```

```
df.Cabin.value_counts()
```

```
B96 B98    4
G6         4
C23 C25 C27 4
C22 C26    3
F33        3
..
E34        1
C7         1
C54        1
E36        1
C148       1
Name: Cabin, Length: 147, dtype: int64
```

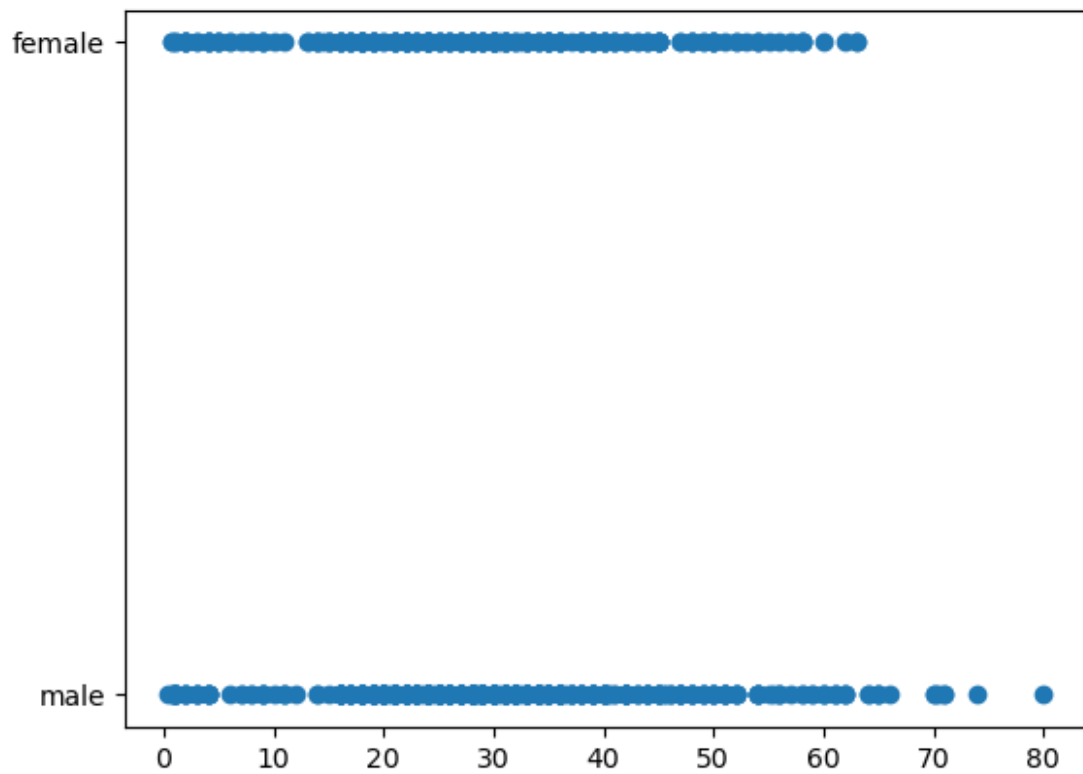
```
df.SibSp.value_counts()
```

```
0    608
1    209
2     28
4     18
3     16
8      7
5      5
Name: SibSp, dtype: int64
```

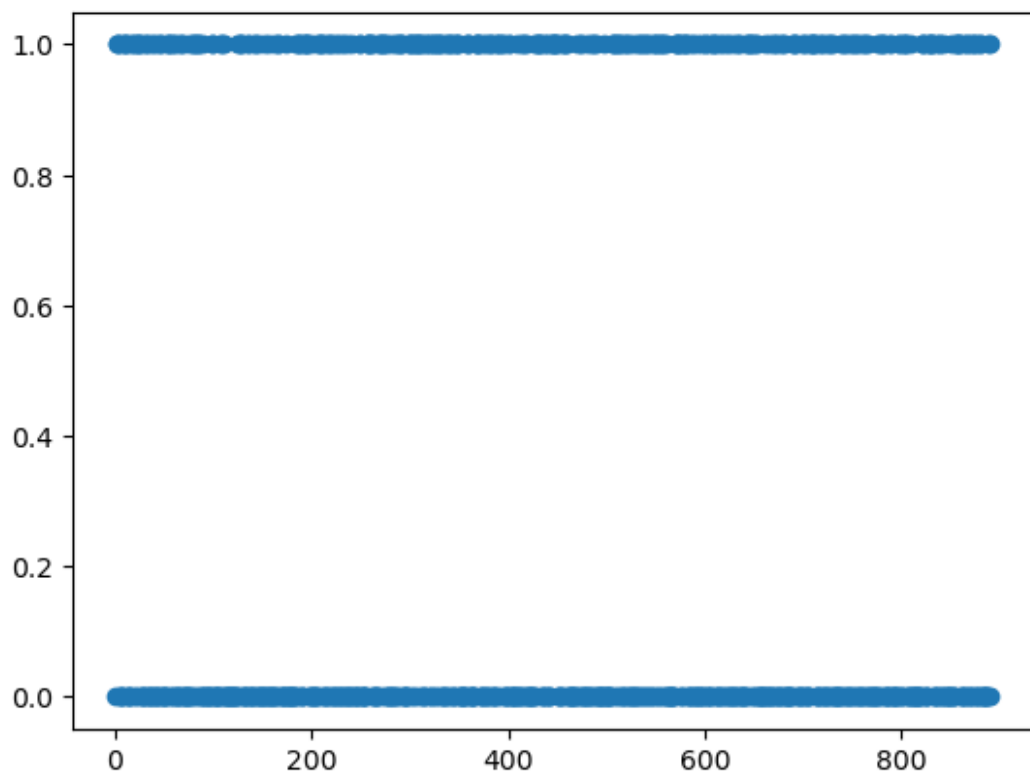
4.Data Visualization.

```
plt.scatter(df["Age"],df["Sex"])
```

```
<matplotlib.collections.PathCollection at 0x1a48ed80a10>
```

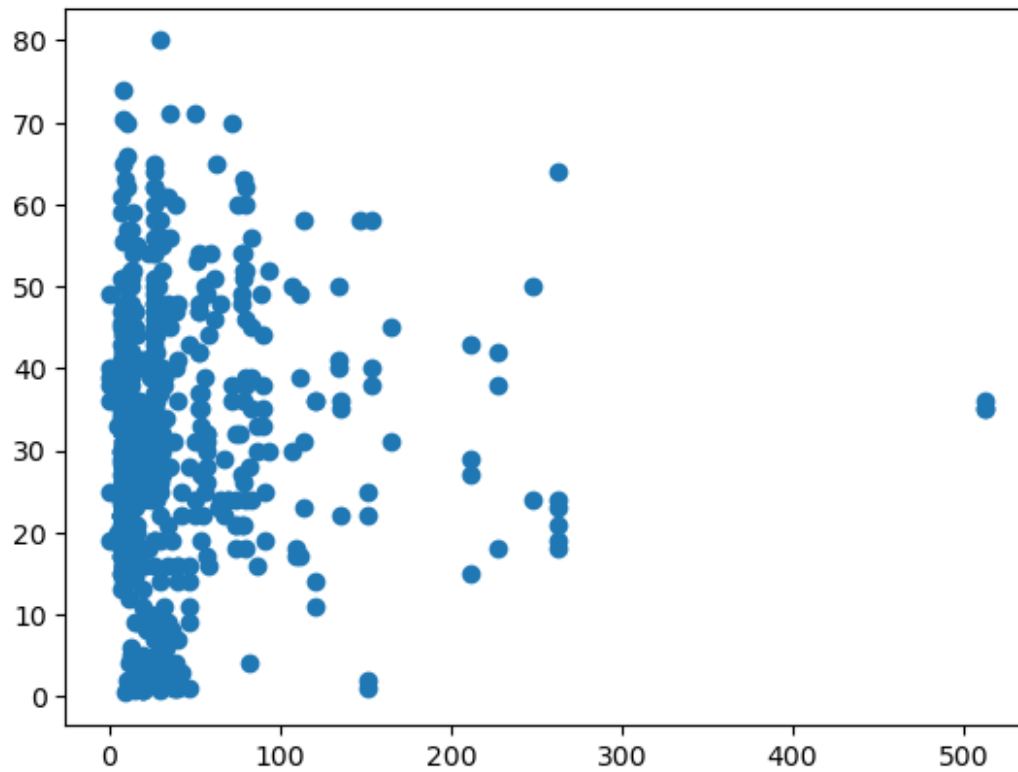


```
plt.scatter(df["PassengerId"],df["Survived"])  
<matplotlib.collections.PathCollection at 0x1a48f694b90>
```

```
plt.scatter(df["Fare"],df["Age"])
```

```
<matplotlib.collections.PathCollection at 0x1a48eba28d0>
```

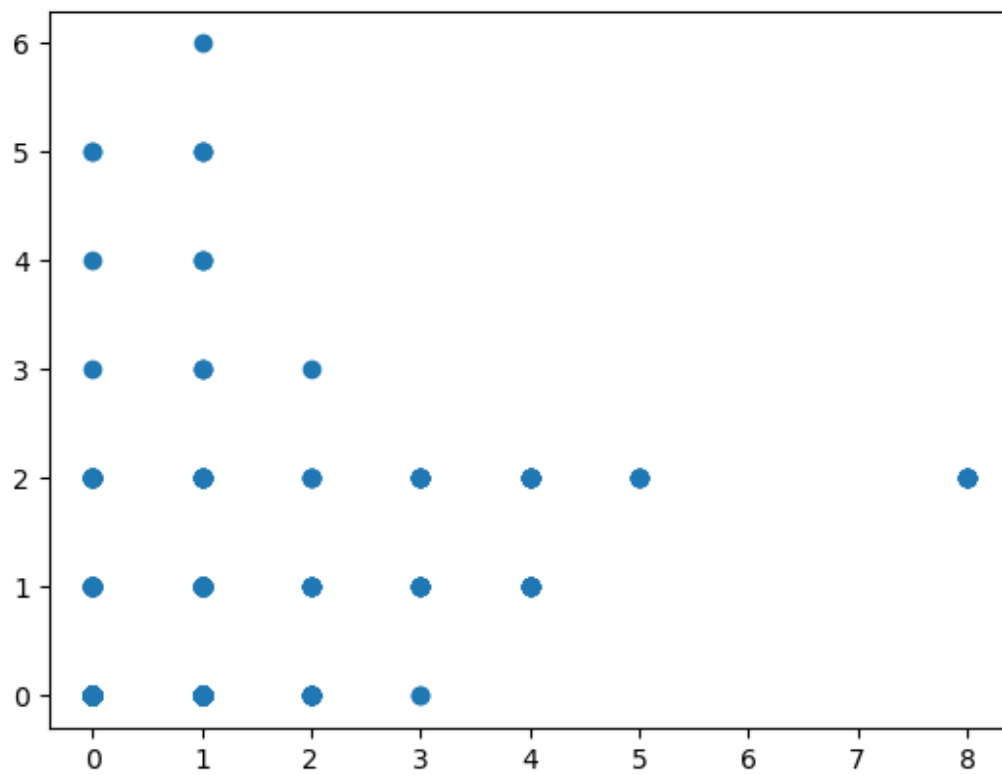


```
plt.scatter(df["Name"],df["Sex"])
<matplotlib.collections.PathCollection at 0x1a48eec7e50>
```



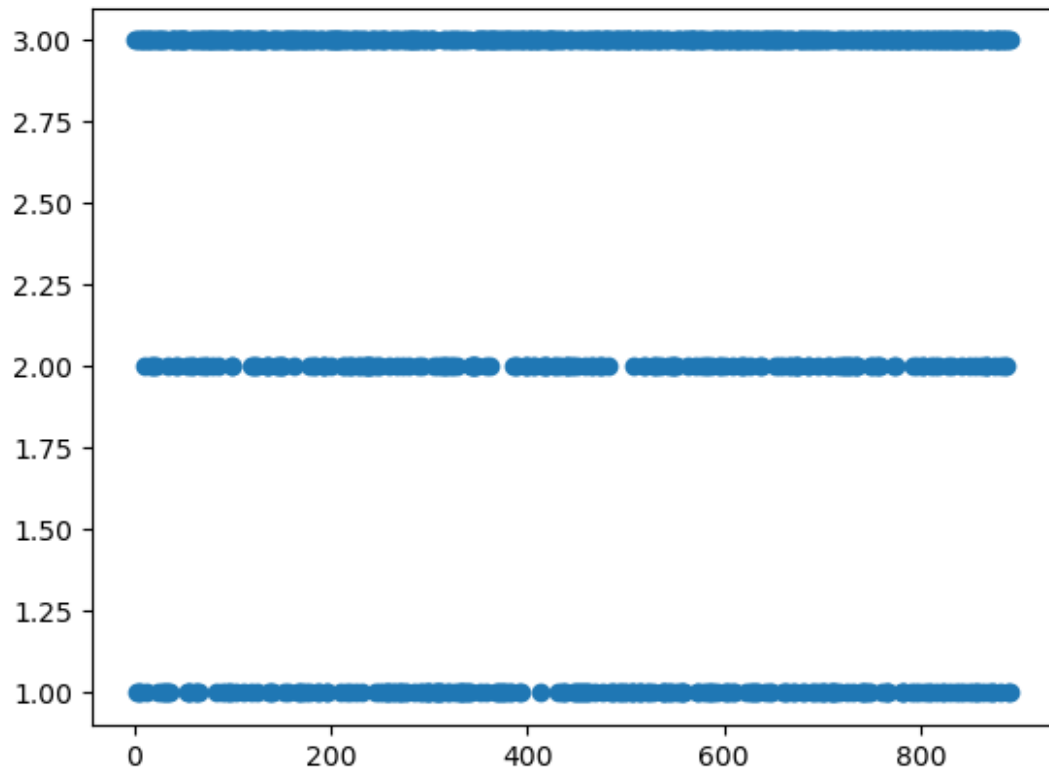
```
plt.scatter(df["SibSp"],df["Parch"])
<matplotlib.collections.PathCollection at 0x1a48eec7e50>
```

```
<matplotlib.collections.PathCollection at 0x1a4914bed10>
```

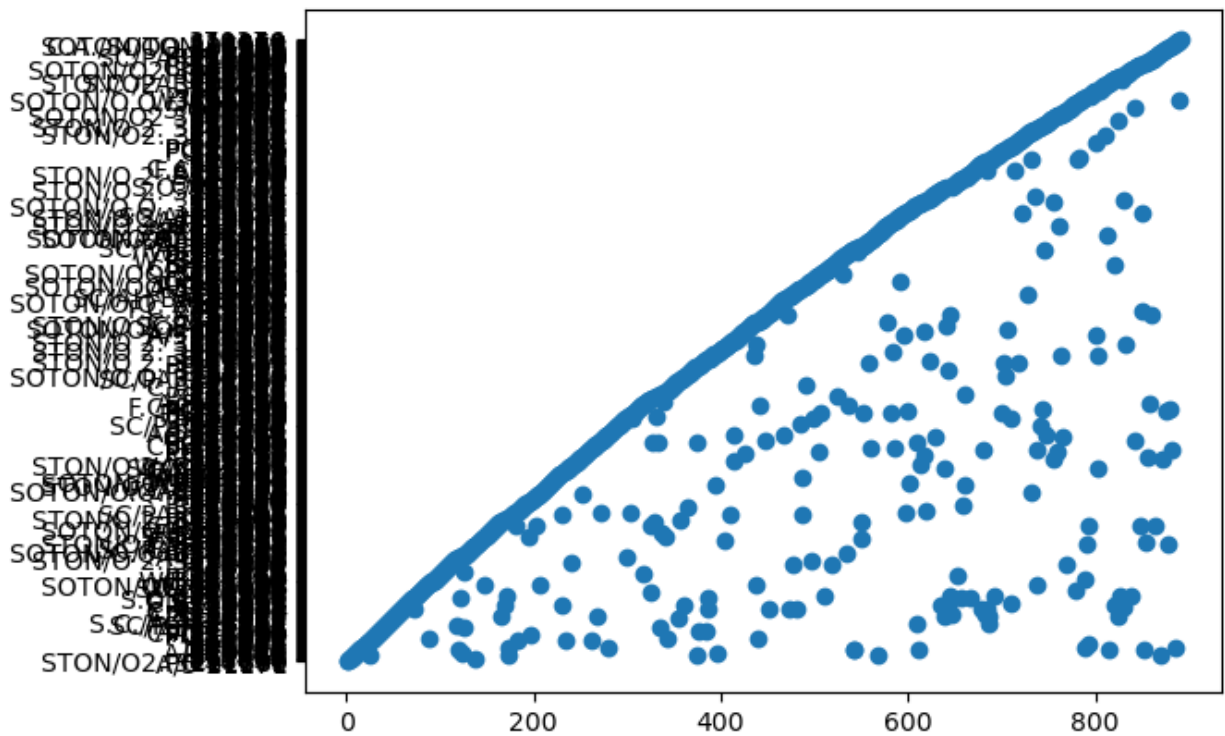


```
plt.scatter(df["PassengerId"],df["Pclass"])
```

```
<matplotlib.collections.PathCollection at 0x1a49159cd10>
```

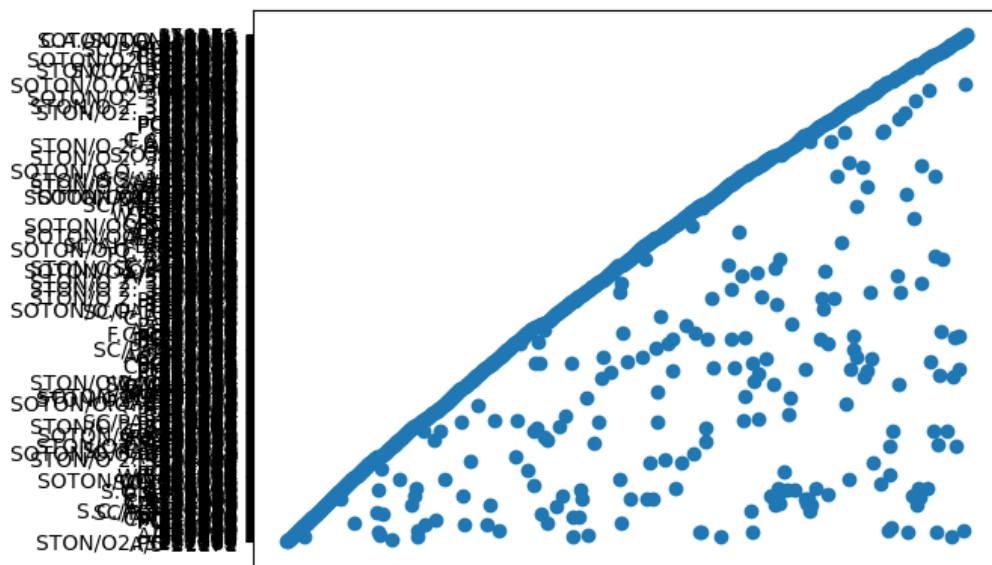


```
plt.scatter(df["PassengerId"],df["Ticket"])  
<matplotlib.collections.PathCollection at 0x1a48f714090>
```



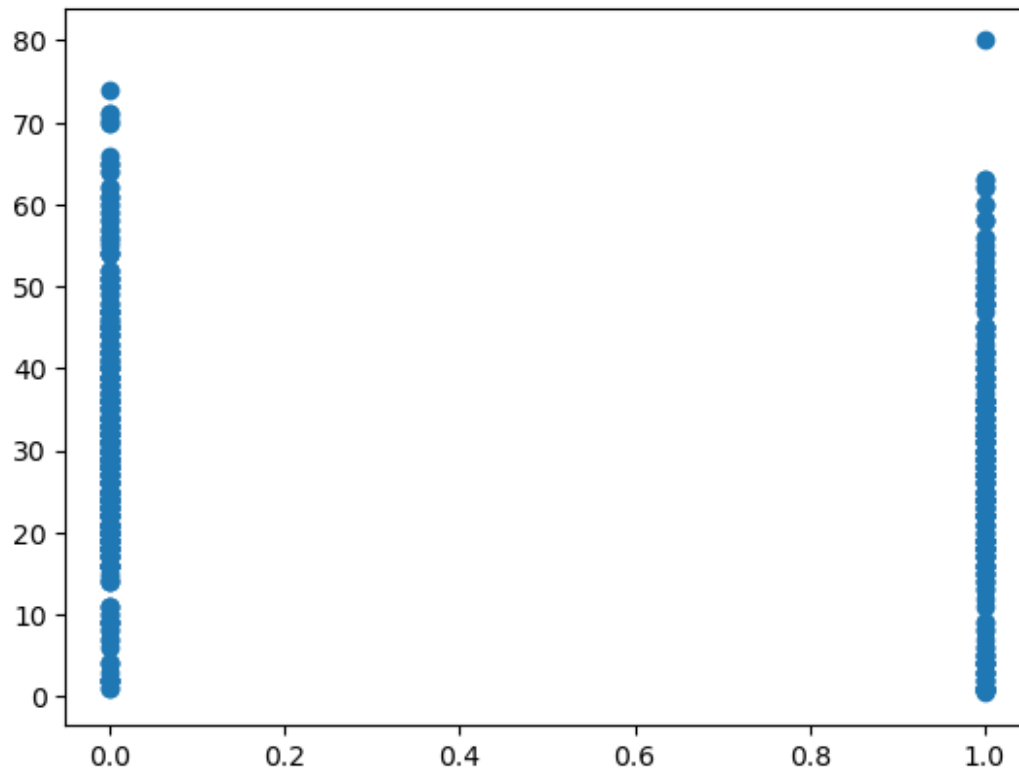
```
plt.scatter(df["Name"],df["Ticket"])
```

```
<matplotlib.collections.PathCollection at 0x1a49205fcd0>
```



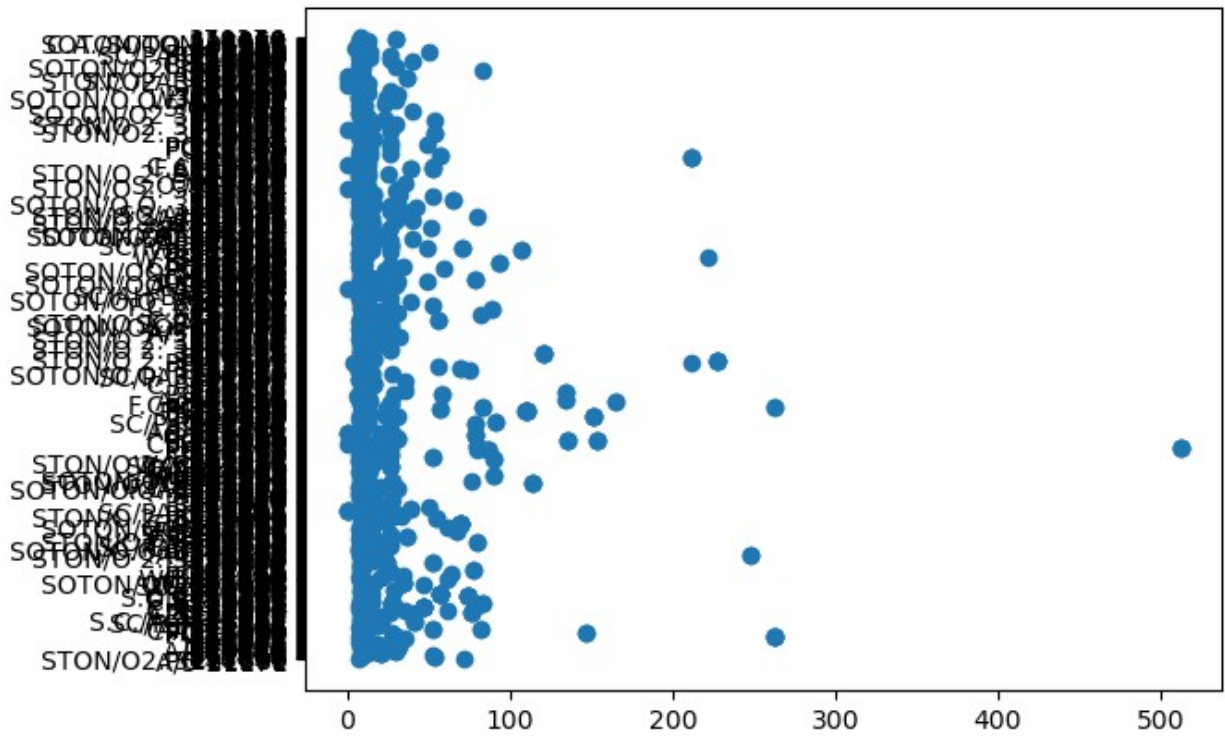
```
plt.scatter(df["Survived"],df["Age"])
```

```
<matplotlib.collections.PathCollection at 0x1a491a5ea90>
```



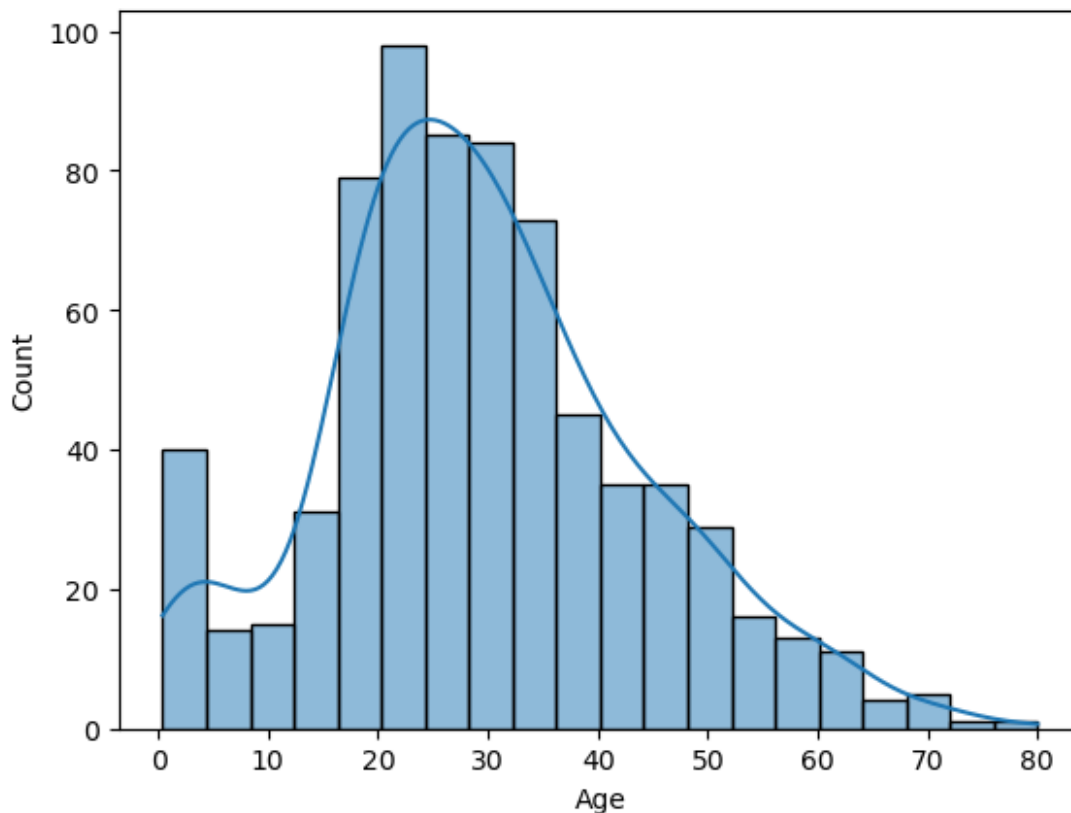
```
plt.scatter(df["Fare"],df["Ticket"])
```

```
<matplotlib.collections.PathCollection at 0x1a4948fda50>
```



```
sns.histplot(data=df,x="Age",bins=20,kde=True)
```

```
<Axes: xlabel='Age', ylabel='Count'>
```



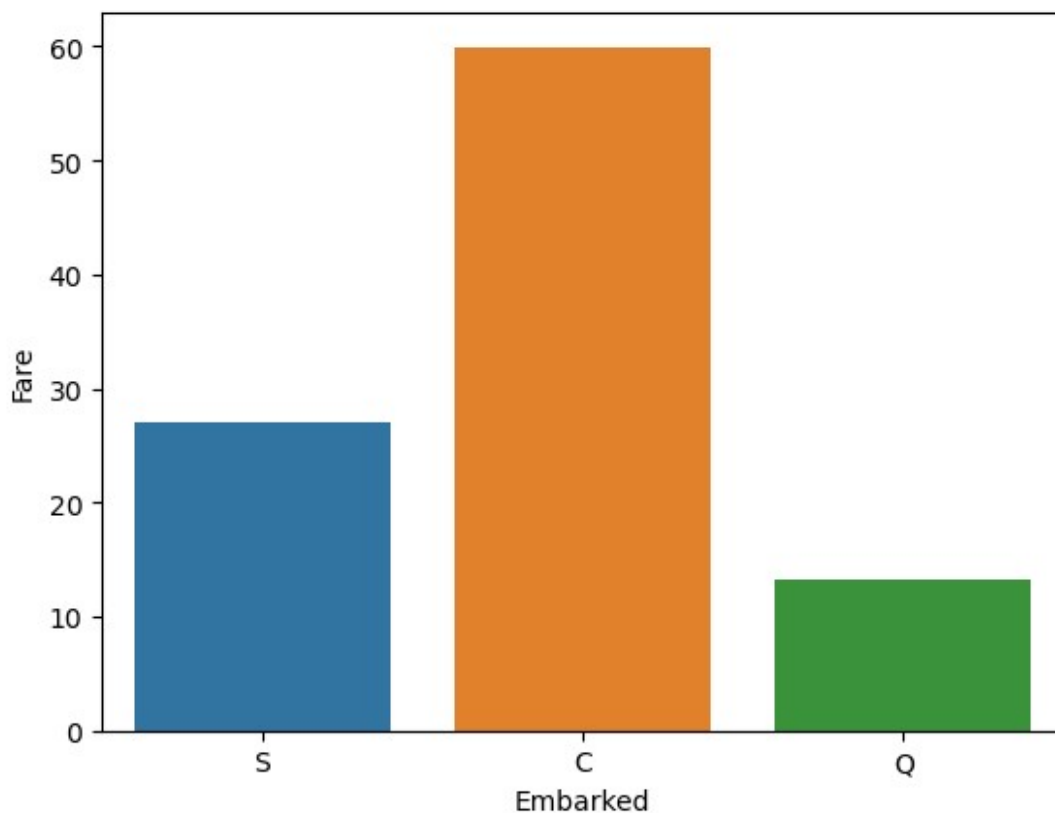
```
sns.barplot(x=df["Embarked"],y=df["Fare"],ci=None)
```

C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\2315957941.py:1:
FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=None` for the same effect.

```
sns.barplot(x=df["Embarked"],y=df["Fare"],ci=None)
```

```
<Axes: xlabel='Embarked', ylabel='Fare'>
```

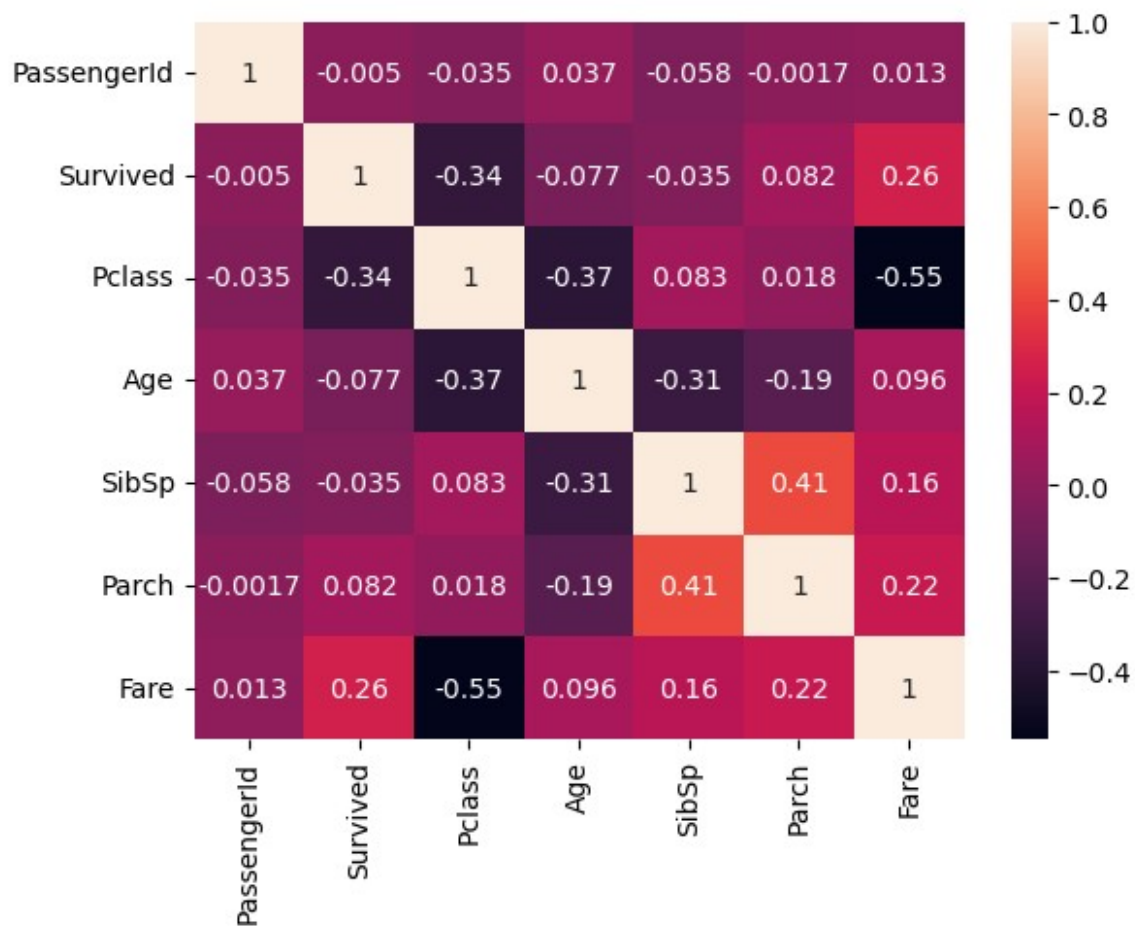



```
sns.heatmap(df.corr(),annot=True)
```

```
C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\4277794465.py:1:  
FutureWarning: The default value of numeric_only in DataFrame.corr is  
deprecated. In a future version, it will default to False. Select only  
valid columns or specify the value of numeric_only to silence this  
warning.
```

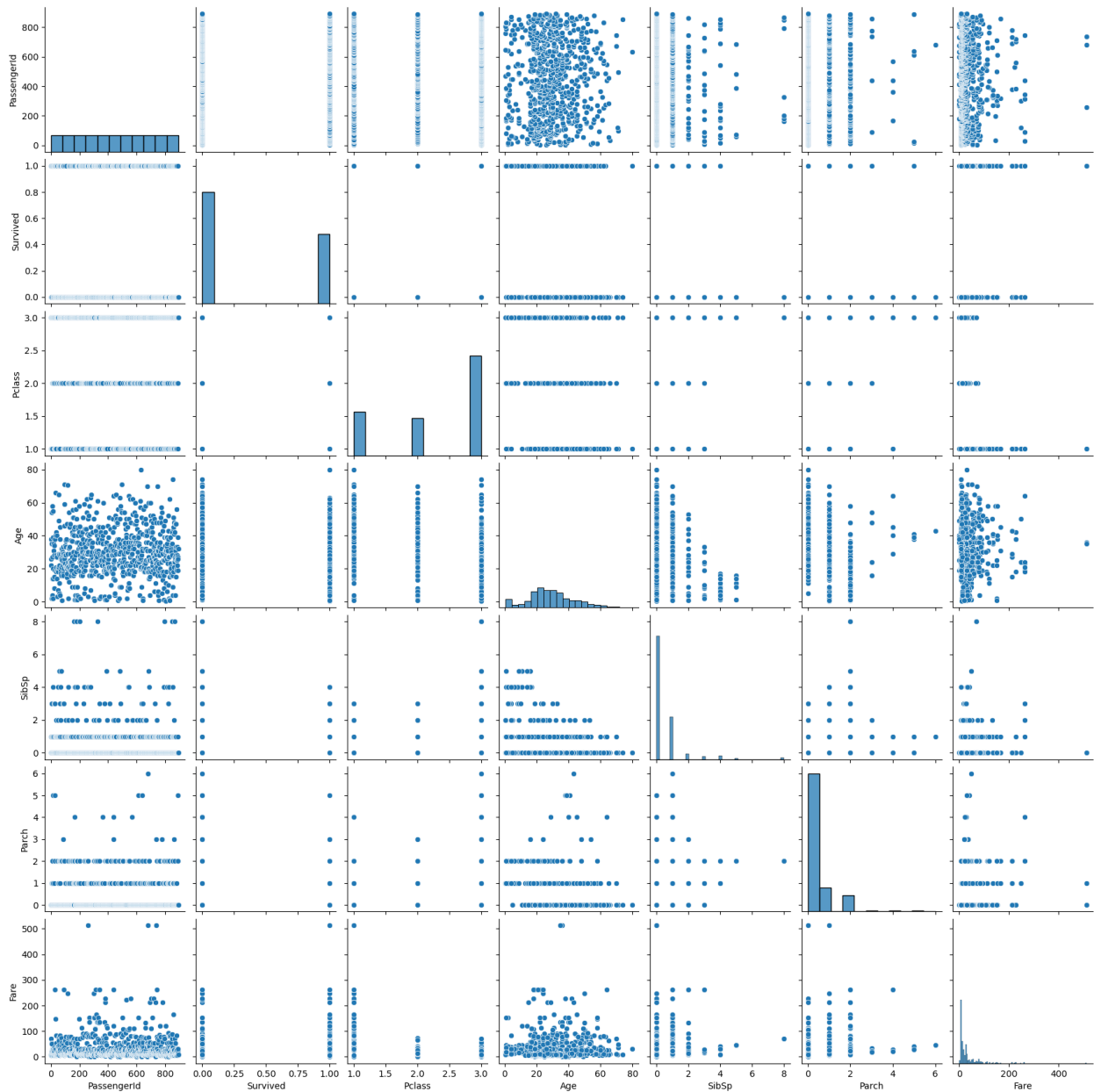
```
sns.heatmap(df.corr(),annot=True)
```

```
<Axes: >
```



```
sns.pairplot(df)
```

```
<seaborn.axisgrid.PairGrid at 0x1a4917f6e90>
```



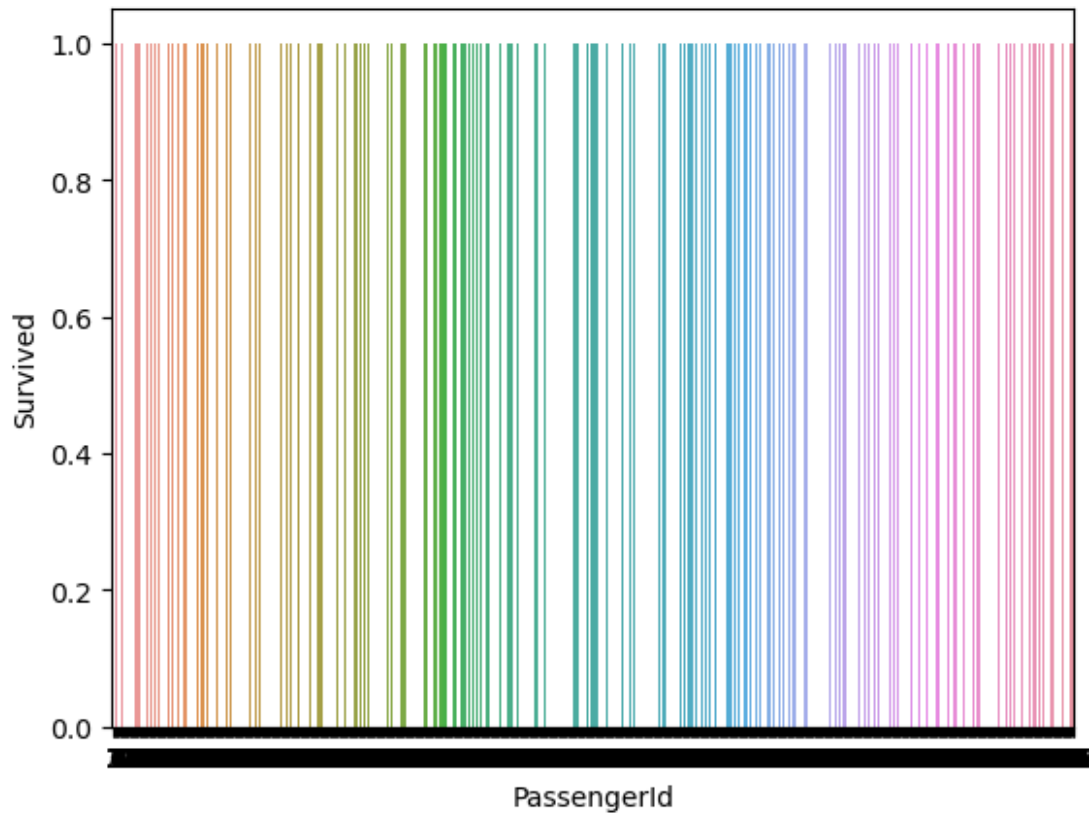
```
sns.barplot(x=df["PassengerId"],y=df["Survived"],ci=0)
```

C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\995913145.py:1:
FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', 0)` for the same effect.

```
sns.barplot(x=df["PassengerId"],y=df["Survived"],ci=0)
```

```
<Axes: xlabel='PassengerId', ylabel='Survived'>
```



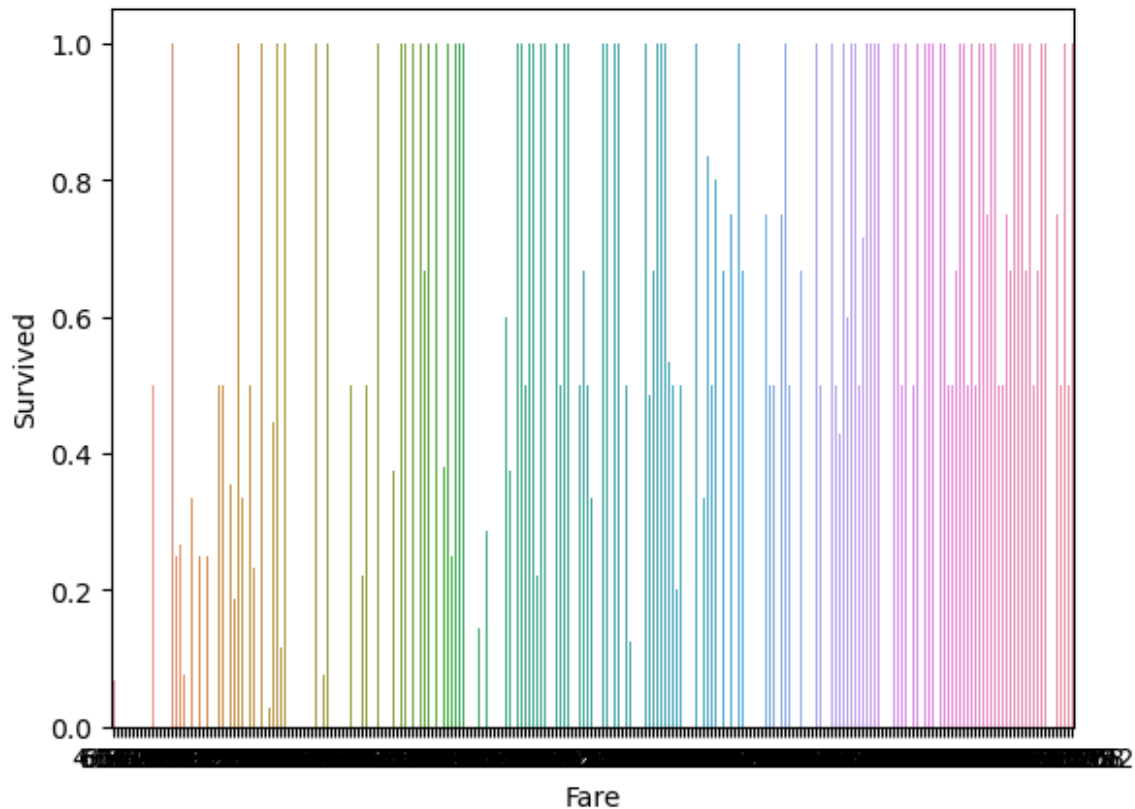
```
sns.barplot(x=df["Fare"],y=df["Survived"],ci=0)
```

C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\2258517547.py:1:
FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', 0)` for the same effect.

```
sns.barplot(x=df["Fare"],y=df["Survived"],ci=0)
```

```
<Axes: xlabel='Fare', ylabel='Survived'>
```



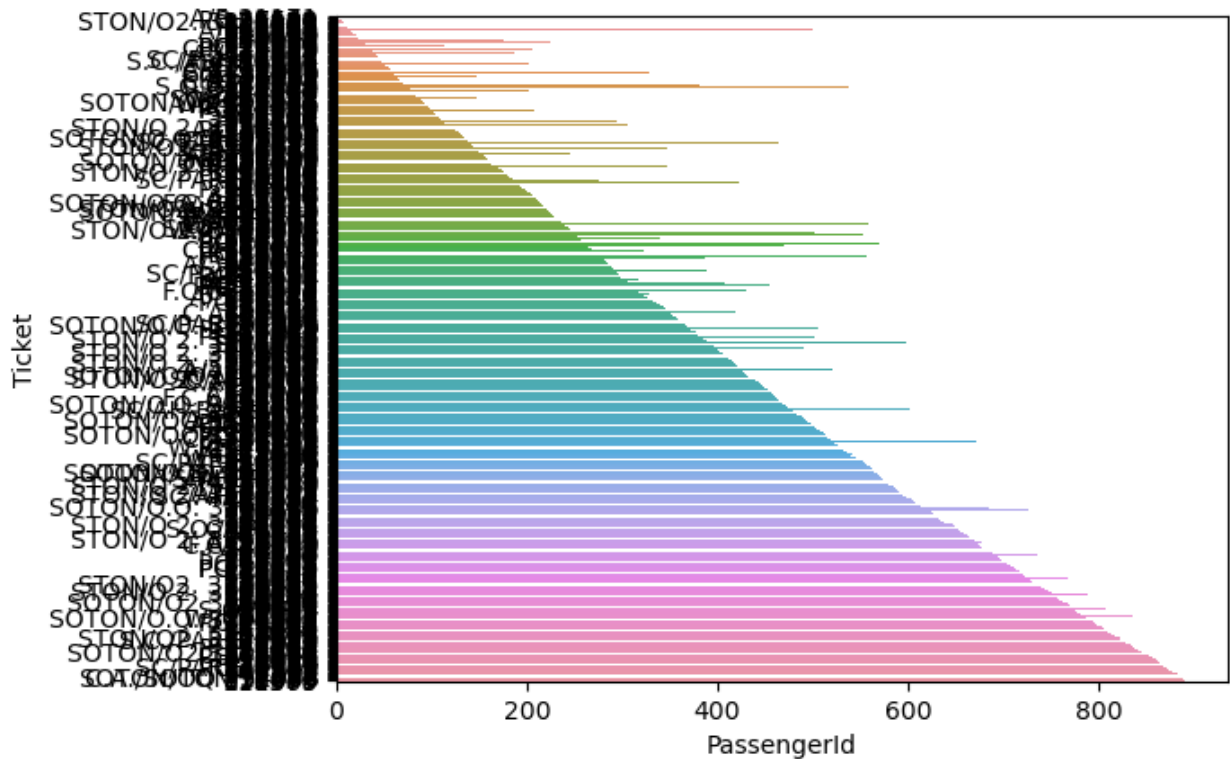
```
sns.barplot(x=df["PassengerId"],y=df["Ticket"],ci=0)
```

C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\1551021374.py:1:
FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', 0)` for the same effect.

```
sns.barplot(x=df["PassengerId"],y=df["Ticket"],ci=0)
```

```
<Axes: xlabel='PassengerId', ylabel='Ticket'>
```



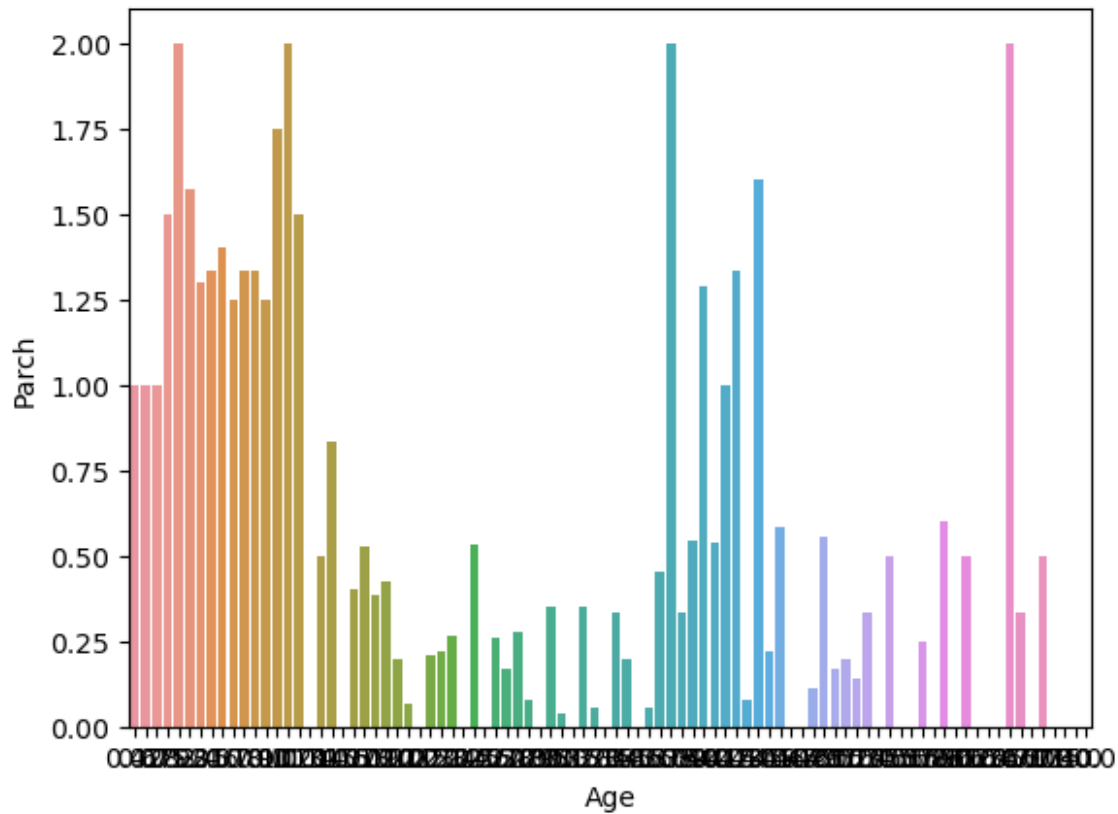
```
sns.barplot(x=df["Age"],y=df["Parch"],ci=0)
```

C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\3471573399.py:1:
FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', 0)` for the same effect.

```
sns.barplot(x=df["Age"],y=df["Parch"],ci=0)
```

```
<Axes: xlabel='Age', ylabel='Parch'>
```



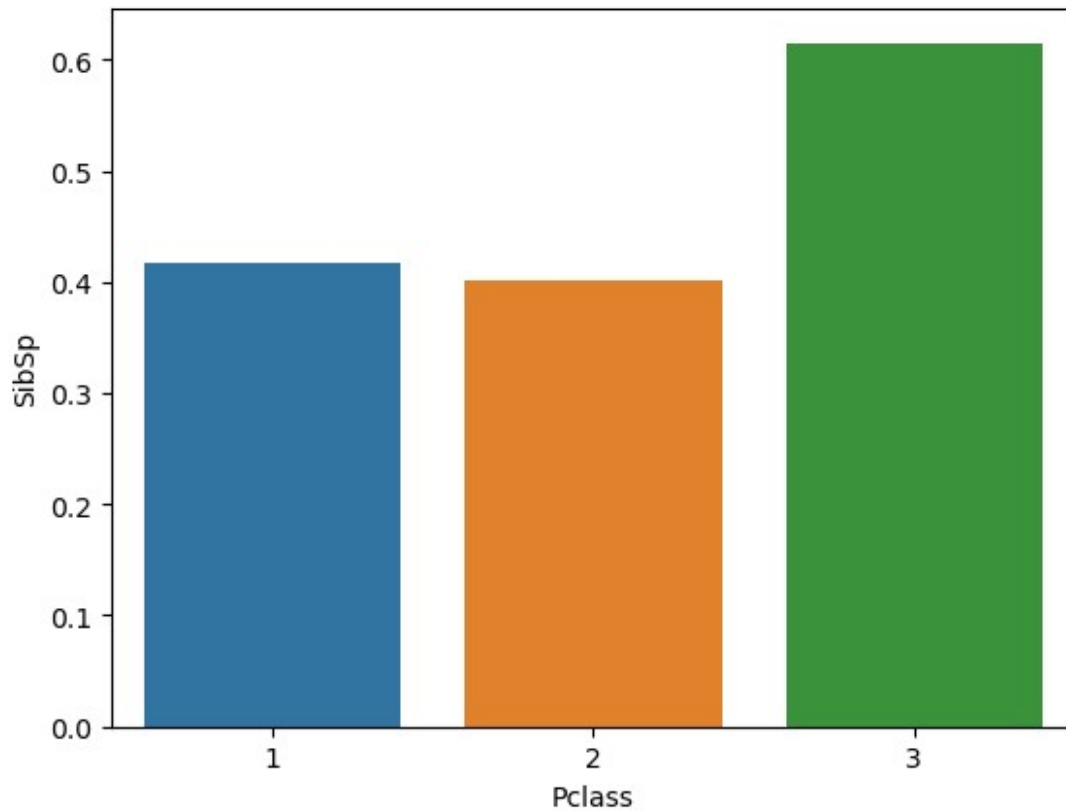
```
sns.barplot(x=df["Pclass"],y=df["SibSp"],ci=0)
```

C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\2568251139.py:1:
FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', 0)` for the same effect.

```
sns.barplot(x=df["Pclass"],y=df["SibSp"],ci=0)
```

```
<Axes: xlabel='Pclass', ylabel='SibSp'>
```



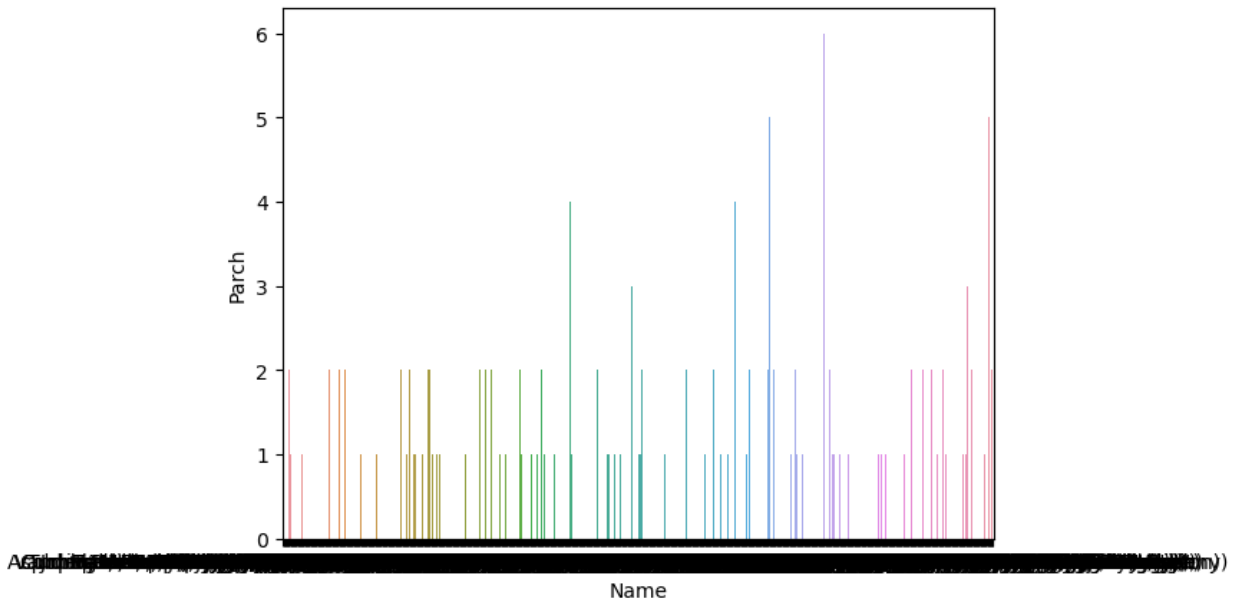
```
sns.barplot(x=df["Name"],y=df["Parch"],ci=0)
```

C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\1639528596.py:1:
FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', 0)` for the same effect.

```
sns.barplot(x=df["Name"],y=df["Parch"],ci=0)
```

```
<Axes: xlabel='Name', ylabel='Parch'>
```

```
sns.barplot(x=df["Age"],y=df["Ticket"],ci=0)
```

C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\3692608565.py:1:
FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', 0)` for the same effect.

```
sns.barplot(x=df["Age"],y=df["Ticket"],ci=0)
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
  boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
  boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
  boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\numpy\lib\
nanfunctions.py:1556: RuntimeWarning: All-NaN slice encountered
  return function_base._ureduce(a,
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
  boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
  boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\numpy\lib\
nanfunctions.py:1556: RuntimeWarning: All-NaN slice encountered
  return function_base._ureduce(a,
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
```

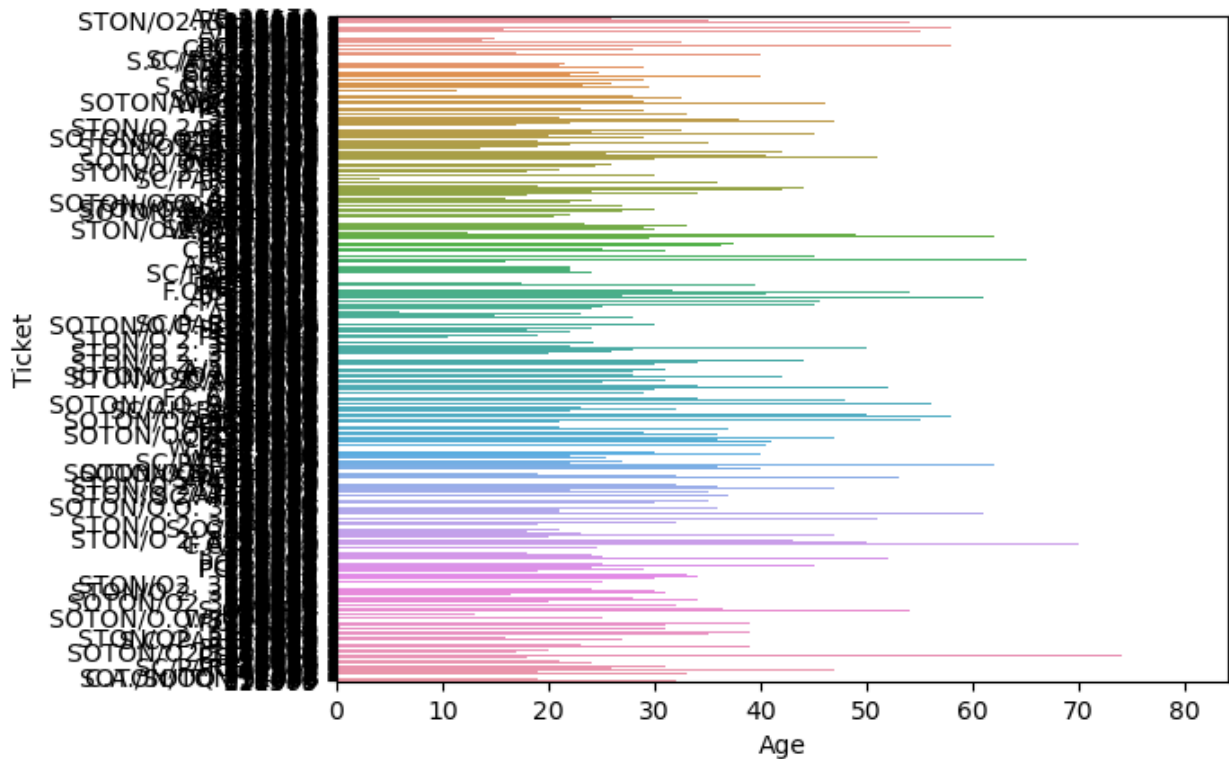
```
RuntimeWarning: Mean of empty slice
  boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
  boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
  boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
  boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\numpy\lib\
nanfunctions.py:1556: RuntimeWarning: All-NaN slice encountered
  return function_base._ureduce(a,
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
  boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
  boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\numpy\lib\
nanfunctions.py:1556: RuntimeWarning: All-NaN slice encountered
  return function_base._ureduce(a,
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
  boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\numpy\lib\
nanfunctions.py:1556: RuntimeWarning: All-NaN slice encountered
  return function_base._ureduce(a,
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
  boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\numpy\lib\
nanfunctions.py:1556: RuntimeWarning: All-NaN slice encountered
  return function_base._ureduce(a,
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
  boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
```

```

    boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\numpy\lib\
nanfunctions.py:1556: RuntimeWarning: All-NaN slice encountered
    return function_base._ureduce(a,
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
    boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\numpy\lib\
nanfunctions.py:1556: RuntimeWarning: All-NaN slice encountered
    return function_base._ureduce(a,
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
    boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
    boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
    boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\numpy\lib\
nanfunctions.py:1556: RuntimeWarning: All-NaN slice encountered
    return function_base._ureduce(a,
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
    boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
    boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
    boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
    boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:
RuntimeWarning: Mean of empty slice
    boot_dist.append(f(*sample, **func_kwargs))
C:\Users\Asus\anaconda3\Lib\site-packages\numpy\lib\
nanfunctions.py:1556: RuntimeWarning: All-NaN slice encountered
    return function_base._ureduce(a,

<Axes: xlabel='Age', ylabel='Ticket'>

```



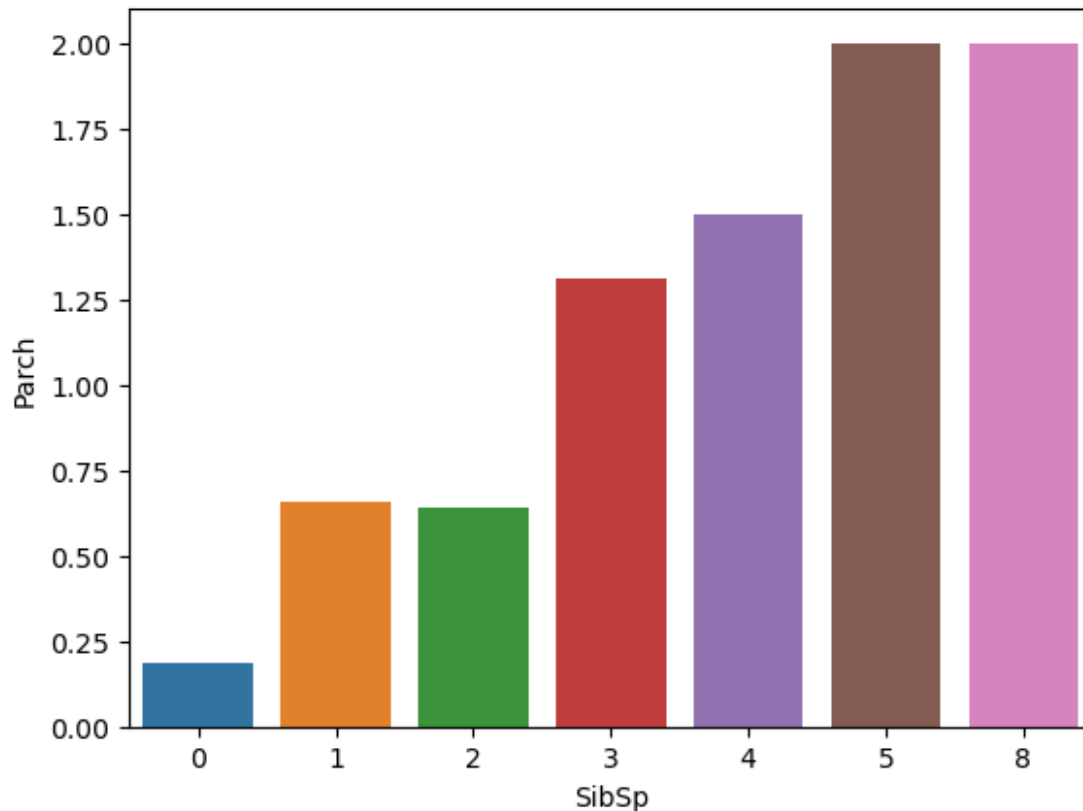
```
sns.barplot(x=df["SibSp"],y=df["Parch"],ci=0)
```

C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\1052202501.py:1:
FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', 0)` for the same effect.

```
sns.barplot(x=df["SibSp"],y=df["Parch"],ci=0)
```

```
<Axes: xlabel='SibSp', ylabel='Parch'>
```



```
sns.barplot(x=df["SibSp"],y=df["Age"],ci=0)
```

C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\1896325314.py:1:
FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', 0)` for the same effect.

```
sns.barplot(x=df["SibSp"],y=df["Age"],ci=0)
```

C:\Users\Asus\anaconda3\Lib\site-packages\seaborn\algorithms.py:98:

RuntimeWarning: Mean of empty slice

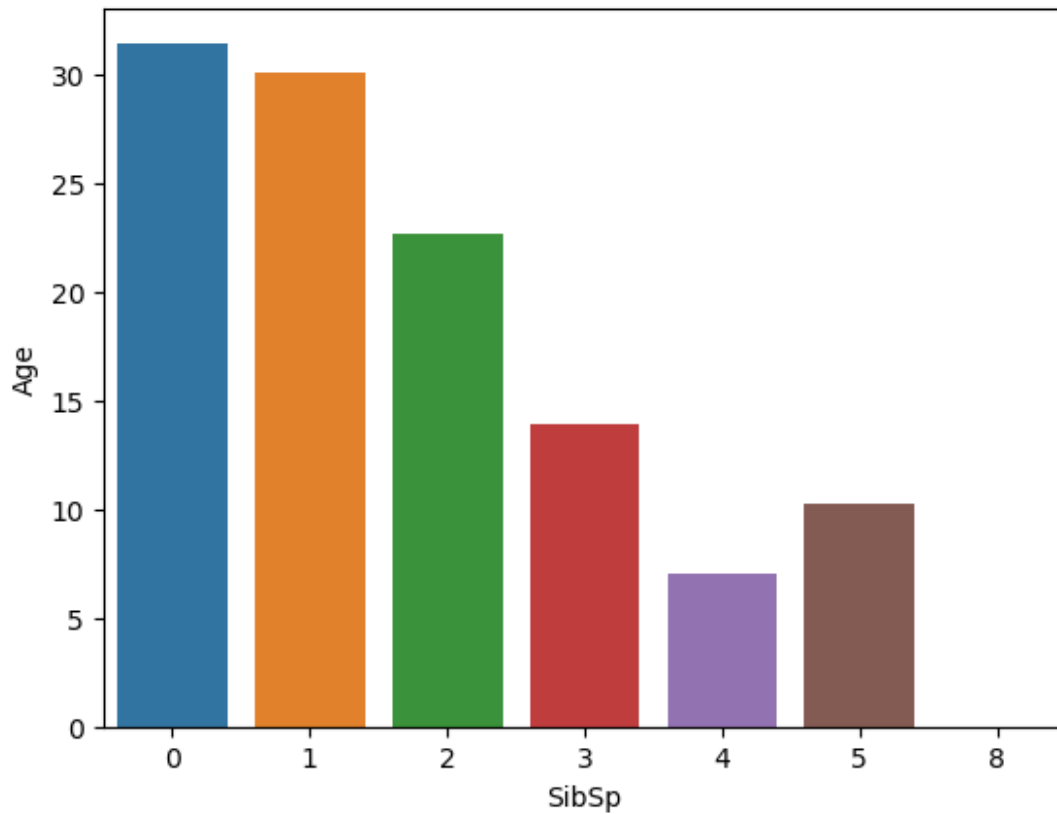
```
boot_dist.append(f(*sample, **func_kwargs))
```

C:\Users\Asus\anaconda3\Lib\site-packages\numpy\lib\

nanfunctions.py:1556: RuntimeWarning: All-NaN slice encountered

```
return function_base._ureduce(a,
```

```
<Axes: xlabel='SibSp', ylabel='Age'>
```



```
sns.barplot(x=df["Cabin"],y=df["Parch"],ci=0)
```

C:\Users\Asus\AppData\Local\Temp\ipykernel_7624\1665784577.py:1:
FutureWarning:

The `ci` parameter is deprecated. Use `errorbar=('ci', 0)` for the same effect.

```
sns.barplot(x=df["Cabin"],y=df["Parch"],ci=0)
```

```
<Axes: xlabel='Cabin', ylabel='Parch'>
```


0	0	A/5	21171	7.2500	NaN	S
1	0	PC	17599	71.2833	C85	C
2	0	STON/02.	3101282	7.9250	NaN	S
3	0		113803	53.1000	C123	S
4	0		373450	8.0500	NaN	S

```
from scipy import stats
z_scores = np.abs(stats.zscore(df["PassengerId"]))
outliers=df["PassengerId"][z_scores>3]
outliers
```

```
Series([], Name: PassengerId, dtype: int64)
```

```
z_scores = np.abs(stats.zscore(df["Fare"]))
outliers=df["Fare"][z_scores>3]
outliers
```

27	263.0000
88	263.0000
118	247.5208
258	512.3292
299	247.5208
311	262.3750
341	263.0000
377	211.5000
380	227.5250
438	263.0000
527	221.7792
557	227.5250
679	512.3292
689	211.3375
700	227.5250
716	227.5250
730	211.3375
737	512.3292
742	262.3750
779	211.3375

```
Name: Fare, dtype: float64
```

```
Q1=df["Fare"].quantile (0.25)
Q3=df["Fare"].quantile (0.75)
```

```
IQR = Q3-Q1
```

```
lower_bound = Q1-1.5* IQR
upper_bound = Q3 + 1.5 * IQR
```

```
df_cleaned = df[(df["Fare"]> lower_bound) & (df["Fare"] <
upper_bound)]
```

```
print (f"Original DataFrame size: {df.shape}")
```



```
print (f"Cleaned DataFrame size: {df_cleaned.shape}")
df_cleaned
```

Original DataFrame size: (891, 12)
Cleaned DataFrame size: (775, 12)

	PassengerId	Survived	Pclass	\
0	1	0	3	
2	3	1	3	
3	4	1	1	
4	5	0	3	
5	6	0	3	
..	
886	887	0	2	
887	888	1	1	
888	889	0	3	
889	890	1	1	
890	891	0	3	

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

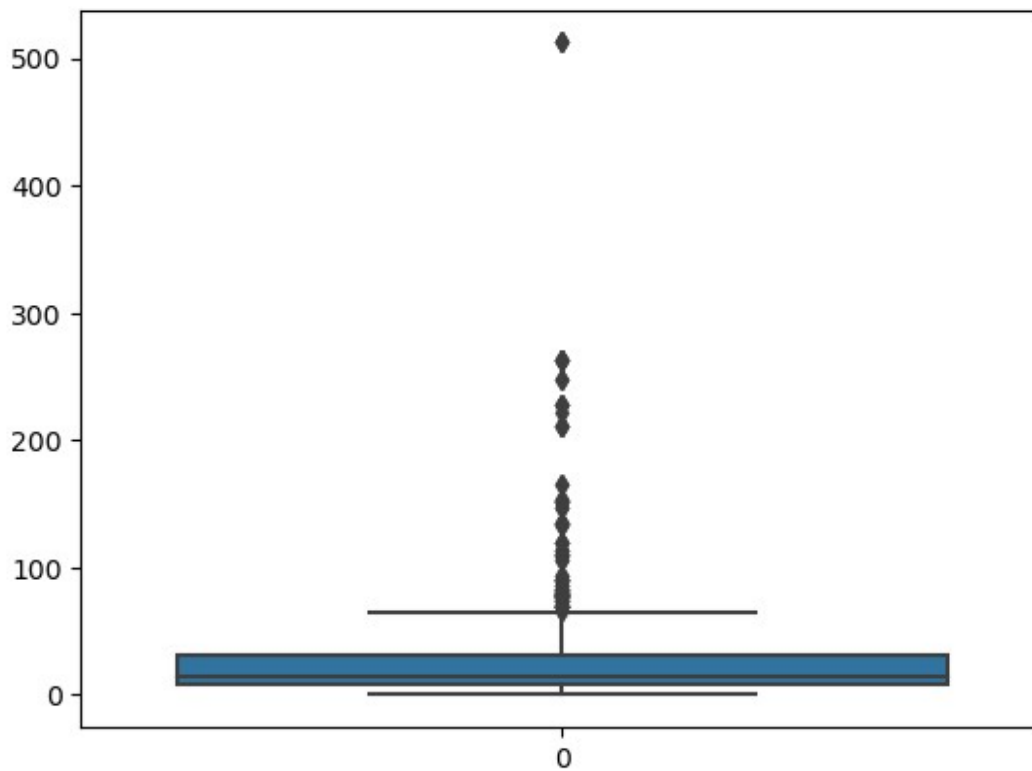
	Ticket	Fare	Cabin	Embarked
0	A/5 21171	7.2500	NaN	S
2	STON/O2. 3101282	7.9250	NaN	S
3	113803	53.1000	C123	S
4	373450	8.0500	NaN	S
5	330877	8.4583	NaN	Q

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

[775 rows x 12 columns]

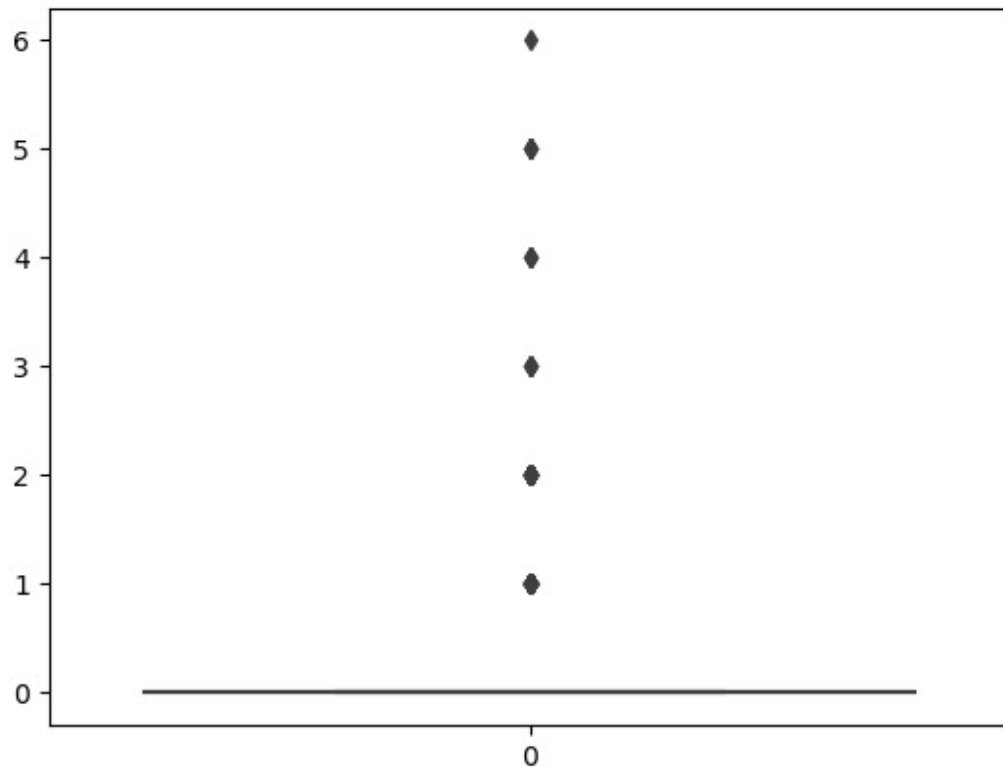
```
sns.boxplot(df["Fare"])
```

<Axes: >



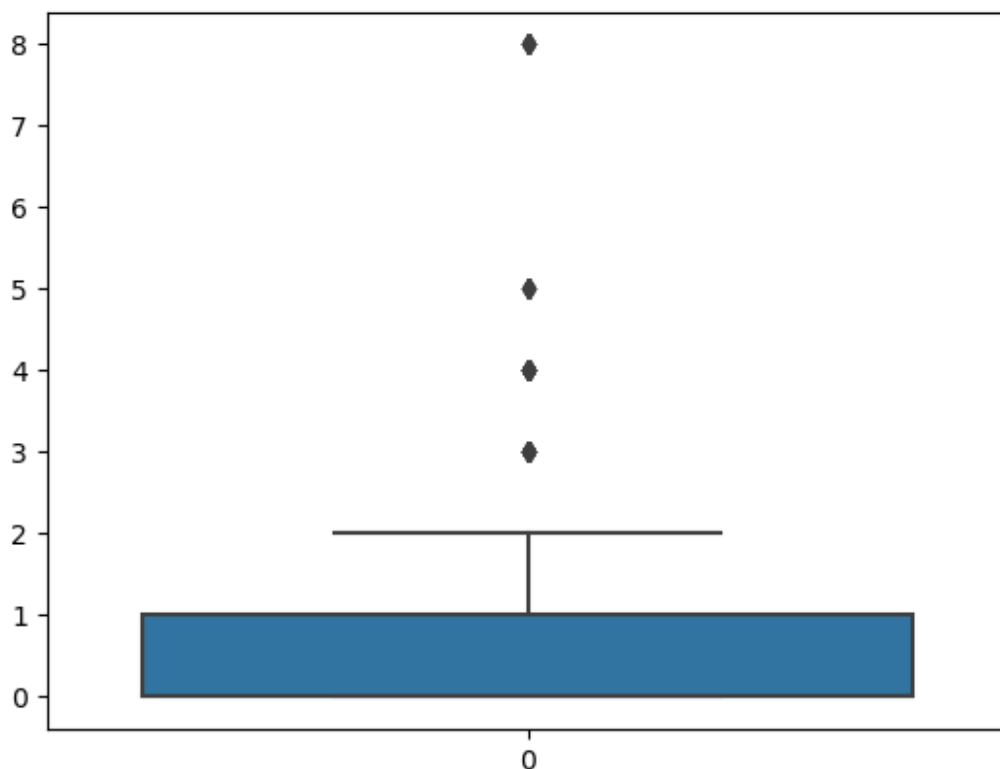
```
sns.boxplot(df["Parch"])
```

<Axes: >



```
sns.boxplot(df["SibSp"])
```

```
<Axes: >
```



6.Splitting Dependent and Independent Variables

df.head()

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

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

	Parch	Ticket	Fare	Cabin	Embarked
0	0	A/5 21171	7.2500	NaN	S

1	0	PC 17599	71.2833	C85	C
2	0	STON/02. 3101282	7.9250	NaN	S
3	0	113803	53.1000	C123	S
4	0	373450	8.0500	NaN	S

```
X=df.drop(columns=["PassengerId"],axis=1)
```

```
X
```

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

	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
Embarked							
0	male	22.0	1	0	A/5 21171	7.2500	NaN
S							
1	female	38.0	1	0	PC 17599	71.2833	C85
C							
2	female	26.0	0	0	STON/02. 3101282	7.9250	NaN
S							
3	female	35.0	1	0	113803	53.1000	C123
S							
4	male	35.0	0	0	373450	8.0500	NaN
S							
..
...							
886	male	27.0	0	0	211536	13.0000	NaN

```

S
887 female 19.0 0 0 112053 30.0000 B42
S
888 female NaN 1 2 W./C. 6607 23.4500 NaN
S
889 male 26.0 0 0 111369 30.0000 C148
C
890 male 32.0 0 0 370376 7.7500 NaN
Q

```

```
[891 rows x 11 columns]
```

```

X=df.drop(columns=["Survived"],axis=1)
X

```

	PassengerId	Pclass	
Name \			
0	1	3	Braund, Mr. Owen
Harris			
1	2	1	Cumings, Mrs. John Bradley (Florence Briggs
Th...			
2	3	3	Heikkinen, Miss.
Laina			
3	4	1	Futrelle, Mrs. Jacques Heath (Lily May
Peel)			
4	5	3	Allen, Mr. William
Henry			
..	
...			
886	887	2	Montvila, Rev.
Juozas			
887	888	1	Graham, Miss. Margaret
Edith			
888	889	3	Johnston, Miss. Catherine Helen
"Carrie"			
889	890	1	Behr, Mr. Karl
Howell			
890	891	3	Dooley, Mr.
Patrick			

	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
Embarked							
0	male	22.0	1	0	A/5 21171	7.2500	NaN
S							
1	female	38.0	1	0	PC 17599	71.2833	C85
C							
2	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN
S							
3	female	35.0	1	0	113803	53.1000	C123
S							

4	male	35.0	0	0	373450	8.0500	NaN
S							
...
...							
886	male	27.0	0	0	211536	13.0000	NaN
S							
887	female	19.0	0	0	112053	30.0000	B42
S							
888	female	NaN	1	2	W./C. 6607	23.4500	NaN
S							
889	male	26.0	0	0	111369	30.0000	C148
C							
890	male	32.0	0	0	370376	7.7500	NaN
Q							

[891 rows x 11 columns]

```
X=df.drop(columns=["Pclass"],axis=1)
X
```

	PassengerId	Survived	
Name \			
0	1	0	Braund, Mr. Owen
Harris			
1	2	1	Cumings, Mrs. John Bradley (Florence
Briggs Th...			
2	3	1	Heikkinen,
Miss. Laina			
3	4	1	Futrelle, Mrs. Jacques Heath (Lily
May Peel)			
4	5	0	Allen, Mr.
William Henry			
..	
...			
886	887	0	Montvila, Rev.
Juozas			
887	888	1	Graham, Miss.
Margaret Edith			
888	889	0	Johnston, Miss. Catherine Helen
"Carrie"			
889	890	1	Behr, Mr. Karl
Howell			
890	891	0	Dooley, Mr.
Patrick			

	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin
Embarked							
0	male	22.0	1	0	A/5 21171	7.2500	NaN
S							
1	female	38.0	1	0	PC 17599	71.2833	C85

```

C
2   female  26.0      0      0  STON/O2. 3101282   7.9250   NaN
S
3   female  35.0      1      0                113803  53.1000  C123
S
4   male    35.0      0      0                373450   8.0500   NaN
S
..      ...      ...      ...      ...      ...      ...
...
886   male   27.0      0      0                211536  13.0000   NaN
S
887   female 19.0      0      0                112053  30.0000  B42
S
888   female  NaN      1      2          W./C. 6607   23.4500   NaN
S
889   male   26.0      0      0                111369  30.0000  C148
C
890   male   32.0      0      0                370376   7.7500   NaN
Q

```

[891 rows x 11 columns]

```
X=df.drop(columns=["Fare"],axis=1)
```

X

```

      PassengerId  Survived  Pclass  \
0                1         0        3
1                2         1        1
2                3         1        3
3                4         1        1
4                5         0        3
..            ...      ...      ...
886            887         0        2
887            888         1        1
888            889         0        3
889            890         1        1
890            891         0        3

```

```

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

```



```

..          ...
...
886          Montvila, Rev. Juozas      male  27.0
0
887          Graham, Miss. Margaret Edith  female  19.0
0
888          Johnston, Miss. Catherine Helen "Carrie"  female  NaN
1
889          Behr, Mr. Karl Howell      male  26.0
0
890          Dooley, Mr. Patrick      male  32.0
0

```

```

Parch      Ticket Cabin Embarked
0          0      A/5 21171   NaN      S
1          0      PC 17599   C85      C
2          0  STON/O2. 3101282   NaN      S
3          0      113803   C123      S
4          0      373450   NaN      S
..          ...
886          0      211536   NaN      S
887          0      112053   B42      S
888          2      W./C. 6607   NaN      S
889          0      111369   C148      C
890          0      370376   NaN      Q

```

[891 rows x 11 columns]

```

X=df.drop(columns=["Age"],axis=1)
X

```

```

PassengerId  Survived  Pclass  \
0             1         0       3
1             2         1       1
2             3         1       3
3             4         1       1
4             5         0       3
..          ...
886          887         0       2
887          888         1       1
888          889         0       3
889          890         1       1
890          891         0       3

```

```

Name      Sex  SibSp
Parch  \
0      Braund, Mr. Owen Harris      male      1
0
1  Cumings, Mrs. John Bradley (Florence Briggs Th...  female      1
0

```

```

2           Heikkinen, Miss. Laina  female  0
0
3  Futrelle, Mrs. Jacques Heath (Lily May Peel)  female  1
0
4           Allen, Mr. William Henry  male  0
0
..           ...  ...  ...
...
886           Montvila, Rev. Juozas  male  0
0
887           Graham, Miss. Margaret Edith  female  0
0
888  Johnston, Miss. Catherine Helen "Carrie"  female  1
2
889           Behr, Mr. Karl Howell  male  0
0
890           Dooley, Mr. Patrick  male  0
0

```

```

      Ticket      Fare Cabin Embarked
0      A/5 21171   7.2500   NaN        S
1      PC 17599  71.2833   C85        C
2  STON/O2. 3101282   7.9250   NaN        S
3      113803  53.1000  C123        S
4      373450   8.0500   NaN        S
..      ...      ...      ...      ...
886      211536  13.0000   NaN        S
887      112053  30.0000  B42        S
888  W./C. 6607  23.4500   NaN        S
889      111369  30.0000  C148        C
890      370376   7.7500   NaN        Q

```

[891 rows x 11 columns]

```
X=df.drop(columns=["Sex"],axis=1)
```

X

```

      PassengerId  Survived  Pclass  \
0                1         0        3
1                2         1        1
2                3         1        3
3                4         1        1
4                5         0        3
..            ...      ...      ...
886            887         0        2
887            888         1        1
888            889         0        3
889            890         1        1
890            891         0        3

```

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

	Ticket	Fare	Cabin	Embarked
0	A/5 21171	7.2500	NaN	S
1	PC 17599	71.2833	C85	C
2	STON/O2. 3101282	7.9250	NaN	S
3	113803	53.1000	C123	S
4	373450	8.0500	NaN	S
..
886	211536	13.0000	NaN	S
887	112053	30.0000	B42	S
888	W./C. 6607	23.4500	NaN	S
889	111369	30.0000	C148	C
890	370376	7.7500	NaN	Q

[891 rows x 11 columns]

```
X=df.drop(columns=["SibSp"],axis=1)
X
```

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

886	887	0	2
887	888	1	1
888	889	0	3
889	890	1	1
890	891	0	3

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

	Ticket	Fare	Cabin	Embarked
0	A/5 21171	7.2500	NaN	S
1	PC 17599	71.2833	C85	C
2	STON/O2. 3101282	7.9250	NaN	S
3	113803	53.1000	C123	S
4	373450	8.0500	NaN	S
..
886	211536	13.0000	NaN	S
887	112053	30.0000	B42	S
888	W./C. 6607	23.4500	NaN	S
889	111369	30.0000	C148	C
890	370376	7.7500	NaN	Q

[891 rows x 11 columns]

```
X=df.drop(columns=["Parch"],axis=1)
X
```

	PassengerId	Survived	Pclass	\
0	1	0	3	
1	2	1	1	
2	3	1	3	
3	4	1	1	
4	5	0	3	
..	
886	887	0	2	
887	888	1	1	
888	889	0	3	
889	890	1	1	
890	891	0	3	

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

	Ticket	Fare	Cabin	Embarked
0	A/5 21171	7.2500	NaN	S
1	PC 17599	71.2833	C85	C
2	STON/O2. 3101282	7.9250	NaN	S
3	113803	53.1000	C123	S
4	373450	8.0500	NaN	S
..
886	211536	13.0000	NaN	S
887	112053	30.0000	B42	S
888	W./C. 6607	23.4500	NaN	S
889	111369	30.0000	C148	C
890	370376	7.7500	NaN	Q

```
[891 rows x 11 columns]
```

```
X=df.drop(columns=["Ticket"],axis=1)
```

```
X
```

	PassengerId	Survived	Pclass	\
0	1	0	3	
1	2	1	1	
2	3	1	3	
3	4	1	1	
4	5	0	3	
..	
886	887	0	2	
887	888	1	1	
888	889	0	3	
889	890	1	1	
890	891	0	3	

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

	Parch	Fare	Cabin	Embarked
0	0	7.2500	NaN	S
1	0	71.2833	C85	C
2	0	7.9250	NaN	S
3	0	53.1000	C123	S
4	0	8.0500	NaN	S

```

..      ...      ...      ...      ...
886      0      13.0000      NaN      S
887      0      30.0000      B42      S
888      2      23.4500      NaN      S
889      0      30.0000      C148      C
890      0      7.7500      NaN      Q

```

```
[891 rows x 11 columns]
```

```

X=df.drop(columns=["Cabin"],axis=1)
X

```

```

      PassengerId  Survived  Pclass  \
0                1         0        3
1                2         1        1
2                3         1        3
3                4         1        1
4                5         0        3
..            ...      ...      ...
886            887         0        2
887            888         1        1
888            889         0        3
889            890         1        1
890            891         0        3

```

```

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

```

	Parch	Ticket	Fare	Embarked
0	0	A/5 21171	7.2500	S
1	0	PC 17599	71.2833	C
2	0	STON/O2. 3101282	7.9250	S
3	0	113803	53.1000	S
4	0	373450	8.0500	S
..
886	0	211536	13.0000	S
887	0	112053	30.0000	S
888	2	W./C. 6607	23.4500	S
889	0	111369	30.0000	C
890	0	370376	7.7500	Q

[891 rows x 11 columns]

```
X=df.drop(columns=["Embarked"],axis=1)
X
```

	PassengerId	Survived	Pclass	\
0	1	0	3	
1	2	1	1	
2	3	1	3	
3	4	1	1	
4	5	0	3	
..	
886	887	0	2	
887	888	1	1	
888	889	0	3	
889	890	1	1	
890	891	0	3	

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


```

1
889                                Behr, Mr. Karl Howell    male  26.0
0
890                                Dooley, Mr. Patrick      male  32.0
0

```

```

      Parch      Ticket    Fare Cabin
0         0      A/5 21171    7.2500   NaN
1         0      PC 17599   71.2833   C85
2         0  STON/O2. 3101282    7.9250   NaN
3         0      113803   53.1000  C123
4         0      373450    8.0500   NaN
..      ...
886        0      211536   13.0000   NaN
887        0      112053   30.0000   B42
888        2      W./C. 6607   23.4500   NaN
889        0      111369   30.0000  C148
890        0      370376    7.7500   NaN

```

```
[891 rows x 11 columns]
```

```
X.shape
```

```
(891, 11)
```

```
type(X)
```

```
pandas.core.frame.DataFrame
```

```
y=df['PassengerId']
y.head()
```

```

0    1
1    2
2    3
3    4
4    5
Name: PassengerId, dtype: int64

```

```
y=df['Survived']
y.head()
```

```

0    0
1    1
2    1
3    1
4    0
Name: Survived, dtype: int64

```

```
y=df['Pclass']
y.head()
```

```
0    3
1    1
2    3
3    1
4    3
Name: Pclass, dtype: int64
```

```
y=df['Fare']
y.head()
```

```
0    7.2500
1   71.2833
2    7.9250
3   53.1000
4    8.0500
Name: Fare, dtype: float64
```

```
y=df['Age']
y.head()
```

```
0    22.0
1    38.0
2    26.0
3    35.0
4    35.0
Name: Age, dtype: float64
```

```
y=df['Sex']
y.head()
```

```
0    male
1   female
2   female
3   female
4     male
Name: Sex, dtype: object
```

```
y=df['SibSp']
y.head()
```

```
0    1
1    1
2    0
3    1
4    0
Name: SibSp, dtype: int64
```

```
y=df['Cabin']
y.head()
```

```

0      NaN
1      C85
2      NaN
3      C123
4      NaN
Name: Cabin, dtype: object

y=df['Ticket']
y.head()

0      A/5 21171
1      PC 17599
2      STON/O2. 3101282
3      113803
4      373450
Name: Ticket, dtype: object

y=df['Name']
y.head()

0      Braund, Mr. Owen Harris
1      Cumings, Mrs. John Bradley (Florence Briggs Th...
2      Heikkinen, Miss. Laina
3      Futrelle, Mrs. Jacques Heath (Lily May Peel)
4      Allen, Mr. William Henry
Name: Name, dtype: object

```

7.Encoding

```

X.head()

```

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

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

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

```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
```

```
X["PassengerId"]=le.fit_transform(X["PassengerId"])
```

```
X.head()
```

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

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

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

```
X["Survived"]=le.fit_transform(X["Survived"])
X.head()
```

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

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

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

```
print(le.classes_)
```

```
[0 1]
```

```
mapping=dict(zip(le.classes_, range(len(le.classes_))))
```

```
mapping
```

```
{0: 0, 1: 1}
```

8.Feature Scaling

```
from sklearn.preprocessing import MinMaxScaler
```

```
ms=MinMaxScaler()
```

```
import pandas as pd
```

```
from sklearn.preprocessing import StandardScaler
```

```
# Assuming 'X' is your original DataFrame
```

```
# Separate numeric and non-numeric columns
```

```
numeric_cols = X.select_dtypes(include=['number'])
```

```
non_numeric_cols = X.select_dtypes(exclude=['number'])
```

```
# Preprocess non-numeric columns if needed
```

```
# For example, you can use one-hot encoding for categorical columns
```

```
non_numeric_cols_encoded = pd.get_dummies(non_numeric_cols)
```

```
# Concatenate numeric and processed non-numeric columns
```

```
X_processed = pd.concat([numeric_cols, non_numeric_cols_encoded],  
axis=1)
```

```
# Apply StandardScaler to the entire DataFrame
```

```
ms = StandardScaler()
```

```
X_scaled = pd.DataFrame(ms.fit_transform(X_processed),  
columns=X_processed.columns)
```

```
X_scaled
```

	PassengerId	Survived	Pclass	Age	SibSp	Parch
Fare \						
0	-1.730108	-0.789272	0.827377	-0.530377	0.432793	-0.473674
0.502445						
1	-1.726220	1.266990	-1.566107	0.571831	0.432793	-0.473674
0.786845						
2	-1.722332	1.266990	0.827377	-0.254825	-0.474545	-0.473674
0.488854						
3	-1.718444	1.266990	-1.566107	0.365167	0.432793	-0.473674
0.420730						
4	-1.714556	-0.789272	0.827377	0.365167	-0.474545	-0.473674
0.486337						
..
...						
886	1.714556	-0.789272	-0.369365	-0.185937	-0.474545	-0.473674
0.386671						
887	1.718444	1.266990	-1.566107	-0.737041	-0.474545	-0.473674
0.044381						
888	1.722332	-0.789272	0.827377	NaN	0.432793	2.008933
0.176263						
889	1.726220	1.266990	-1.566107	-0.254825	-0.474545	-0.473674
0.044381						
890	1.730108	-0.789272	0.827377	0.158503	-0.474545	-0.473674
0.492378						
	Name_Abbing, Mr. Anthony		Name_Abbott, Mr. Rossmore Edward			
0		-0.03352			-0.03352	
1		-0.03352			-0.03352	
2		-0.03352			-0.03352	
3		-0.03352			-0.03352	
4		-0.03352			-0.03352	
..		
886		-0.03352			-0.03352	
887		-0.03352			-0.03352	
888		-0.03352			-0.03352	
889		-0.03352			-0.03352	
890		-0.03352			-0.03352	
	Name_Abbott, Mrs. Stanton (Rosa Hunt)		...	Cabin_E8	Cabin_F E69	
\						
0		-0.03352	...	-0.047431		-0.03352

1			-0.03352	...	-0.047431	-0.03352
2			-0.03352	...	-0.047431	-0.03352
3			-0.03352	...	-0.047431	-0.03352
4			-0.03352	...	-0.047431	-0.03352
..		
886			-0.03352	...	-0.047431	-0.03352
887			-0.03352	...	-0.047431	-0.03352
888			-0.03352	...	-0.047431	-0.03352
889			-0.03352	...	-0.047431	-0.03352
890			-0.03352	...	-0.047431	-0.03352
	Cabin_F G63	Cabin_F G73	Cabin_F2	Cabin_F33	Cabin_F38	
Cabin_F4 \						
0	-0.03352	-0.047431	-0.058124	-0.058124	-0.03352	-
0.047431						
1	-0.03352	-0.047431	-0.058124	-0.058124	-0.03352	-
0.047431						
2	-0.03352	-0.047431	-0.058124	-0.058124	-0.03352	-
0.047431						
3	-0.03352	-0.047431	-0.058124	-0.058124	-0.03352	-
0.047431						
4	-0.03352	-0.047431	-0.058124	-0.058124	-0.03352	-
0.047431						
..
.						
886	-0.03352	-0.047431	-0.058124	-0.058124	-0.03352	-
0.047431						
887	-0.03352	-0.047431	-0.058124	-0.058124	-0.03352	-
0.047431						
888	-0.03352	-0.047431	-0.058124	-0.058124	-0.03352	-
0.047431						
889	-0.03352	-0.047431	-0.058124	-0.058124	-0.03352	-
0.047431						
890	-0.03352	-0.047431	-0.058124	-0.058124	-0.03352	-
0.047431						
	Cabin_G6	Cabin_T				
0	-0.067153	-0.03352				
1	-0.067153	-0.03352				
2	-0.067153	-0.03352				

```
3    -0.067153 -0.03352
4    -0.067153 -0.03352
..    ...
886  -0.067153 -0.03352
887  -0.067153 -0.03352
888  -0.067153 -0.03352
889  -0.067153 -0.03352
890  -0.067153 -0.03352
```

```
[891 rows x 1728 columns]
```

9. Train test split

```
from sklearn.model_selection import train_test_split
X_train,X_test,y_train,y_test=train_test_split(X,y,test_size=0.2,random_state=0)

print(X_train.shape,X_test.shape,y_train.shape,y_test.shape)

(712, 11) (179, 11) (712,) (179,)
```