In [1]:

```
##Import packages needed for data loading and processing

import pandas as pd
import numpy as np
```

In [2]:

```
##Load the Titanic Dataset to use.

titanicdb = pd.read_csv('/home/ANA522/Titanic.csv', sep = ',')
```

In [3]:

```
##Q01: Use a pandas function to overview the Titanic dataset.
titanicdb
```

Out[3]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

891 rows × 12 columns

In [4]:

```
##Q02: Find a pandas attribute to display data type of each column ( all pokemon dataset properties. )

titanicdb.dtypes
```

Out[4]:

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

In [5]:

```
##Q03: Display all entries whose Age attribute value is missing (NA).

AgeNA = titanicdb[titanicdb['Age'].isna()]
AgeNA
```

Out[5]:

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

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
859	860	0	3	Razi, Mr. Raihed	male	NaN	0	0	2629	7.2292	NaN	C
863	864	0	3	Sage, Miss. Dorothy Edith "Dolly"	female	NaN	8	2	CA. 2343	69.5500	NaN	S
868	869	0	3	van Melkebeke, Mr. Philemon	male	NaN	0	0	345777	9.5000	NaN	S
878	879	0	3	Laleff, Mr. Kristo	male	NaN	0	0	349217	7.8958	NaN	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S

177 rows × 12 columns

In [6]:

```
##Q04: Show how many entries whose Embarked attribute value is missing (NA).

count_NA = titanicdb['Embarked'].isna().sum()

print("There are", count_NA, "entries whose Embarked values is NA")
```

```
There are 2 entries whose Embarked values is NA
```

In [7]:

```
##Q05: Display all entries whose Age attribute value or Cabin attribute value is missing (NA).

selected_rows = titanicdb[titanicdb['Age'].isnull() | titanicdb['Cabin'].isnull()]
selected_rows
```

Out[7]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
5	6	0	3	Moran, Mr. James	male	NaN	0	0	330877	8.4583	NaN	Q
7	8	0	3	Palsson, Master. Gosta Leonard	male	2.0	3	1	349909	21.0750	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...
884	885	0	3	Sutehall, Mr. Henry Jr	male	25.0	0	0	SOTON/OQ 392076	7.0500	NaN	S
885	886	0	3	Rice, Mrs. William (Margaret Norton)	female	39.0	0	5	382652	29.1250	NaN	Q
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

706 rows × 12 columns

In [8]:

```
##Q06: List the indices of all entries whose Age attribute value or Cabin attribute value is missing (NA).

missIndx = titanicdb[titanicdb['Age'].isnull() | titanicdb['Cabin'].isnull()].index.tolist()
print(missIndx)
```

```
[0, 2, 4, 5, 7, 8, 9, 12, 13, 14, 15, 16, 17, 18, 19, 20, 22, 24, 25, 26, 28, 29, 30, 31, 32, 33, 34, 35, 36, 3
7, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 53, 55, 56, 57, 58, 59, 60, 63, 64, 65, 67, 68, 69, 7
0, 71, 72, 73, 74, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87, 89, 90, 91, 93, 94, 95, 98, 99, 100, 101, 10
3, 104, 105, 106, 107, 108, 109, 111, 112, 113, 114, 115, 116, 117, 119, 120, 121, 122, 125, 126, 127, 128, 129,
130, 131, 132, 133, 134, 135, 138, 140, 141, 142, 143, 144, 145, 146, 147, 149, 150, 152, 153, 154, 155, 156, 15
7, 158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 171, 172, 173, 175, 176, 178, 179, 180, 181, 182,
184, 185, 186, 187, 188, 189, 190, 191, 192, 196, 197, 198, 199, 200, 201, 202, 203, 204, 206, 207, 208, 210, 21
1, 212, 213, 214, 216, 217, 219, 220, 221, 222, 223, 225, 226, 227, 228, 229, 231, 232, 233, 234, 235, 236, 237,
238, 239, 240, 241, 242, 243, 244, 246, 247, 249, 250, 253, 254, 255, 256, 258, 259, 260, 261, 264, 265, 266, 26
7, 270, 271, 272, 274, 276, 277, 278, 279, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 293, 294, 295,
296, 298, 300, 301, 302, 303, 304, 306, 308, 312, 313, 314, 315, 316, 317, 320, 321, 322, 323, 324, 326, 328, 33
0, 333, 334, 335, 338, 342, 343, 344, 346, 347, 348, 349, 350, 351, 352, 353, 354, 355, 357, 358, 359, 360, 361,
362, 363, 364, 365, 367, 368, 371, 372, 373, 374, 375, 376, 378, 379, 380, 381, 382, 383, 384, 385, 386, 387, 38
8, 389, 391, 392, 395, 396, 397, 398, 399, 400, 401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 413, 414,
415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427, 428, 431, 432, 433, 436, 437, 439, 440, 441, 44
2, 443, 444, 446, 447, 448, 450, 451, 454, 455, 457, 458, 459, 461, 463, 464, 465, 466, 467, 468, 469, 470, 471,
472, 474, 475, 476, 477, 478, 479, 480, 481, 482, 483, 485, 488, 489, 490, 491, 493, 494, 495, 497, 499, 500, 50
1, 502, 503, 506, 507, 508, 509, 510, 511, 513, 514, 517, 518, 519, 521, 522, 524, 525, 526, 527, 528, 529, 530,
531, 532, 533, 534, 535, 537, 538, 541, 542, 543, 545, 546, 547, 548, 549, 551, 552, 553, 554, 555, 557, 559, 56
0, 561, 562, 563, 564, 565, 566, 567, 568, 569, 570, 573, 574, 575, 576, 578, 579, 580, 582, 584, 586, 588, 589,
590, 592, 593, 594, 595, 596, 597, 598, 600, 601, 602, 603, 604, 605, 606, 607, 608, 610, 611, 612, 613, 614, 61
5, 616, 617, 619, 620, 622, 623, 624, 626, 628, 629, 631, 633, 634, 635, 636, 637, 638, 639, 640, 642, 643, 644,
646, 648, 649, 650, 651, 652, 653, 654, 655, 656, 657, 658, 660, 661, 663, 664, 665, 666, 667, 668, 669, 670, 67
2, 673, 674, 675, 676, 677, 678, 680, 682, 683, 684, 685, 686, 687, 688, 691, 692, 693, 694, 695, 696, 697, 702,
703, 704, 705, 706, 708, 709, 711, 713, 714, 718, 719, 720, 721, 722, 723, 725, 726, 727, 728, 729, 731, 732, 73
3, 734, 735, 736, 738, 739, 740, 743, 744, 746, 747, 749, 750, 752, 753, 754, 755, 756, 757, 758, 760, 761, 762,
764, 766, 767, 768, 769, 770, 771, 773, 774, 775, 776, 777, 778, 780, 783, 784, 785, 786, 787, 788, 790, 791, 79
2, 793, 794, 795, 797, 798, 799, 800, 801, 803, 804, 805, 807, 808, 810, 811, 812, 813, 814, 815, 816, 817, 818,
819, 821, 822, 824, 825, 826, 827, 828, 830, 831, 832, 833, 834, 836, 837, 838, 839, 840, 841, 842, 843, 844, 84
5, 846, 847, 848, 849, 850, 851, 852, 854, 855, 856, 858, 859, 860, 861, 863, 864, 865, 866, 868, 869, 870, 873,
874, 875, 876, 877, 878, 880, 881, 882, 883, 884, 885, 886, 888, 890]
```

In [9]:

```
##Q07: Show how many entries are there both Age and Cabin attribute values are missing (NA).

count1 = len(titanicdb[(titanicdb['Age'].isnull()) & (titanicdb['Cabin'].isnull())])
#count1
print("There are", count1 ,"entries with NA in both Age and Cabin.")
```

```
There are 158 entries with NA in both Age and Cabin.
```

In [10]:

```
##Q08: Sample any 15 entries to be sorted by PassengerId that both Age and Cabin attributes are missing (NA).

missAgeCab = titanicdb[titanicdb['Age'].isnull() & titanicdb['Cabin'].isnull()]
#missAgeCab

missAgeCab15 = missAgeCab.sample(n = 15).sort_values(by='PassengerId')
missAgeCab15
```

Out[10]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
32	33	1	3	Glynn, Miss. Mary Agatha	female	NaN	0	0	335677	7.7500	NaN	Q
36	37	1	3	Mamee, Mr. Hanna	male	NaN	0	0	2677	7.2292	NaN	C
159	160	0	3	Sage, Master. Thomas Henry	male	NaN	8	2	CA. 2343	69.5500	NaN	S
168	169	0	1	Baumann, Mr. John D	male	NaN	0	0	PC 17318	25.9250	NaN	S
198	199	1	3	Madigan, Miss. Margaret "Maggie"	female	NaN	0	0	370370	7.7500	NaN	Q
260	261	0	3	Smith, Mr. Thomas	male	NaN	0	0	384461	7.7500	NaN	Q
425	426	0	3	Wiseman, Mr. Phillippe	male	NaN	0	0	A/4. 34244	7.2500	NaN	S
468	469	0	3	Scanlan, Mr. James	male	NaN	0	0	36209	7.7250	NaN	Q
490	491	0	3	Hagland, Mr. Konrad Mathias Reiersen	male	NaN	1	0	65304	19.9667	NaN	S
502	503	0	3	O'Sullivan, Miss. Bridget Mary	female	NaN	0	0	330909	7.6292	NaN	Q
517	518	0	3	Ryan, Mr. Patrick	male	NaN	0	0	371110	24.1500	NaN	Q
573	574	1	3	Kelly, Miss. Mary	female	NaN	0	0	14312	7.7500	NaN	Q
629	630	0	3	O'Connell, Mr. Patrick D	male	NaN	0	0	334912	7.7333	NaN	Q
760	761	0	3	Garfirth, Mr. John	male	NaN	0	0	358585	14.5000	NaN	S
825	826	0	3	Flynn, Mr. John	male	NaN	0	0	368323	6.9500	NaN	Q

In [11]:

```
###Q09: Display and show how many entries in the dataset have no missing value (NA) in all Age,Cabin, and Embark

newdata = titanicdb.dropna(subset=['Age','Cabin', 'Embarked'])

newdata1 = len(newdata)
print("There are", newdata1, "entries has no NA values in any column.")

newdata
```

```
There are 183 entries has no NA values in any column.
```

Out[11]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
6	7	0	1	McCarthy, Mr. Timothy J	male	54.0	0	0	17463	51.8625	E46	S
10	11	1	3	Sandstrom, Miss. Marguerite Rut	female	4.0	1	1	PP 9549	16.7000	G6	S
11	12	1	1	Bonnell, Miss. Elizabeth	female	58.0	0	0	113783	26.5500	C103	S
...	...	...	...	...	...	...	...	...	...	...	...	...
871	872	1	1	Beckwith, Mrs. Richard Leonard (Sallie Monypeny)	female	47.0	1	1	11751	52.5542	D35	S
872	873	0	1	Carlsson, Mr. Frans Olof	male	33.0	0	0	695	5.0000	B51 B53 B55	S
879	880	1	1	Potter, Mrs. Thomas Jr (Lily Alexenia Wilson)	female	56.0	0	1	11767	83.1583	C50	C
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C

183 rows × 12 columns

In [12]:

```
##Q10: Create a new DataFrame, without modify the original, to hold all entries withtout any NA value
##of any attribute from the original Titanic dataset.

titanicdbNoNA = titanicdb.dropna()
titanicdbNoNA
```

Out[12]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
6	7	0	1	McCarthy, Mr. Timothy J	male	54.0	0	0	17463	51.8625	E46	S
10	11	1	3	Sandstrom, Miss. Marguerite Rut	female	4.0	1	1	PP 9549	16.7000	G6	S
11	12	1	1	Bonnell, Miss. Elizabeth	female	58.0	0	0	113783	26.5500	C103	S
...	...	...	...	...	...	...	...	...	...	...	...	...
871	872	1	1	Beckwith, Mrs. Richard Leonard (Sallie Monypeny)	female	47.0	1	1	11751	52.5542	D35	S
872	873	0	1	Carlsson, Mr. Frans Olof	male	33.0	0	0	695	5.0000	B51 B53 B55	S
879	880	1	1	Potter, Mrs. Thomas Jr (Lily Alexenia Wilson)	female	56.0	0	1	11767	83.1583	C50	C
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C

183 rows × 12 columns

In [13]:

```
###Q11: Replace all missing values (NA) with 0 without overwriting the original dataset by
##createing/saving in a new DataFrame. Display the overview of the new dataset.

titanicdb1 = titanicdb.copy()

#print("creted copy of original dataset")
titanicdb2 = titanicdb1.fillna(0)
#print("replaced all NA with zero")
titanicdb2
```

Out[13]:

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

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	0	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	0	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	0	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	0.0	1	2	W./C. 6607	23.4500	0	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	0	Q

891 rows × 12 columns

In [14]:

```
###Q12: Replace all missing values (NA) in Age column with 0.0, and in Cabin with 'AXX' without
##overwriting the original dataset by createing/saving in a new DataFrame. Display the overview of the new datas

titanicdb3 = titanicdb.copy()

#data_new2['x1'] = data_new2['x1'].fillna(0)
titanicdb3['Age'] = titanicdb3['Age'].fillna(0)

titanicdb3["Cabin"].fillna("AXX", inplace = True)
titanicdb3
```

Out[14]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	AXX	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	AXX	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	AXX	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	AXX	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	0.0	1	2	W./C. 6607	23.4500	AXX	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	AXX	Q

891 rows × 12 columns

In [15]:

```
##Q13: Replace all missing values (NA) using forward filling method without overwriting the original
##dataset, by createing/saving in a new DataFrame. Display the overview of the new dataset.

titanicdb4 = titanicdb.copy()
#titanicdb5 = titanicdb4.ffill(axis = 0)
titanicdb5 = titanicdb4.fillna(method='ffill')
titanicdb5
```

Out[15]:

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

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	C123	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	C50	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	19.0	1	2	W./C. 6607	23.4500	B42	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	C148	Q

891 rows × 12 columns

In [16]:

```
###Q14: Exame the previous new DataFrame resulted from forward filling method, and see if there are still missin

titanicdb5.isna().any()
```

Out[16]:

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

In [17]:

```
##Q15: Port of Embarkation is abbreviated noted in the Embarked column with either 'C','Q',or 'S' if the
##value is not missing. Please write an execution statement to verify the value categorical contents in column E

listunq = (titanicdb['Embarked'].unique().tolist())
print("Total unique values from Embarked column: \n", *listunq)
```

```
Total unique values from Embarked column: 
 S C Q nan
```

In [18]:

```
###Q16: Use the dictionary mapping to create a new column named PortName, to accommodate the full names of the E

port_to_fullname = {'C': 'Cherbourg', 'Q': 'Queenstown', 'S':'Southampton'}
titanicdb['Portname'] = titanicdb['Embarked'].map(port_to_fullname)
titanicdb
```

Out[18]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	Portname
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S	Southampton
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C	Cherbourg
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S	Southampton
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S	Southampton
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S	Southampton
...	...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S	Southampton
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S	Southampton

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	Portname
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S	Southampton
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C	Cherbourg
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q	Queenstown

891 rows × 13 columns

In [19]:

```
###Q17: Print out the index and PassengerId of the first five entries from the original Titanic dataset.

subset = titanicdb.loc[ [0,1,2,3,4] ,['PassengerId'] ]
subset
```

Out[19]:

	PassengerId
0	1
1	2
2	3
3	4
4	5

In [20]:

```
###Q18: Rename row indices to be aligned with PassengerId in place.

titanicdb.index = np.arange(1, len(titanicdb) +1)
titanicdb
```

Out[20]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	Portname
1	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S	Southampton
2	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C	Cherbourg
3	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S	Southampton
4	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S	Southampton
5	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S	Southampton
...	...	...	...	...	...	...	...	...	...	...	...	...	...
887	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S	Southampton
888	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S	Southampton
889	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S	Southampton
890	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C	Cherbourg
891	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q	Queenstown

891 rows × 13 columns

In [21]:

```
###Q19: Display the age distributions of all passengers in the dataset Using the age bins in the list.

ages = [12, 18, 25, 35, 60, 80]
```

```
cats = pd.cut(titanicdb['Age'], ages)
cats.value_counts()
```

Out[21]:

```
(25, 35]    196
(35, 60]    195
(18, 25]    162
(12, 18]     70
(60, 80]     22
Name: Age, dtype: int64
```

In [22]:

```
###Q20: Display the age ranks of all passengers in the dataset using the categories where the whole
##age range of all passengers is divided into equal-length.

ages = [12, 18, 25, 35, 60, 80]
group_names = ['Youth', 'YoungAdult', 'MiddleAged', 'Senior']
grp_names = pd.cut(titanicdb['Age'], 4, labels=group_names)
grp_names
```

Out[22]:

```
1      YoungAdult
2      YoungAdult
3      YoungAdult
4      YoungAdult
5      YoungAdult
          ...
887    YoungAdult
888         Youth
889           NaN
890    YoungAdult
891    YoungAdult
Name: Age, Length: 891, dtype: category
Categories (4, object): ['Youth' < 'YoungAdult' < 'MiddleAged' < 'Senior']
```

In [23]:

```
###Q21: Display the age categories in quartiles of all passengers in the Titanic dataset.

cat_quart = pd.qcut(titanicdb['Age'], 4)
pd.value_counts(cat_quart)
```

Out[23]:

```
(20.125, 28.0]     183
(0.419, 20.125]    179
(38.0, 80.0]       177
(28.0, 38.0]       175
Name: Age, dtype: int64
```

In [24]:

```
##Q22: Detect Fare outliners, assuming there shouldn't be free ($0) ticket.

col = titanicdb['Fare']
col1= col[np.abs(col) < 1]
print("There are" , len(col1), "zero dollar tickets.")
```

```
There are 15 zero dollar tickets.
```

In [ ]:

```
###Q23: Convert categorical data on Embarked column into a dummy mattrix with C, Q, S as indicators

#pd.get_dummies(df['key'])
pd.get_dummies(titanicdb['Embarked'])
```

In [ ]:

```
###Q24: Convert age distributions of all passengers in the Titanic dataset into a dummy mattrix with the
##age bins in the list.

ages = [12, 18, 25, 35, 60, 80]

pd.get_dummies(pd.cut(titanicdb['Age'], ages))
```

In [ ]:

```

```