FOR THE TITANIC DATASET PERFORM THE FOLLOWING

In [10]:
```
# Package imports
import pandas as pd
#import matplotlib.pyplot as plt
import missingno as msno
%matplotlib inline
```

In [5]:
```
#Importing the required dataset

titanic_df = pd.read_csv("titanic.csv")
titanic_df
```

Out[5]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True
...	...	...	...	...	...	...	...	...	...	...	...
886	0	2	male	27.0	0	0	13.0000	S	Second	man	True
887	1	1	female	19.0	0	0	30.0000	S	First	woman	False
888	0	3	female	NaN	1	2	23.4500	S	Third	woman	False
889	1	1	male	26.0	0	0	30.0000	C	First	man	True
890	0	3	male	32.0	0	0	7.7500	Q	Third	man	True

891 rows × 15 columns

1.Display the number of missing values for each feature in the dataset

In [6]:
```
titanic_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   survived     891 non-null    int64
 1   pclass       891 non-null    int64
 2   sex          891 non-null    object
 3   age          714 non-null    float64
 4   sibsp        891 non-null    int64
 5   parch        891 non-null    int64
 6   fare         891 non-null    float64
 7   embarked     889 non-null    object
 8   class        891 non-null    object
 9   who          891 non-null    object
 10  adult_male   891 non-null    bool
 11  deck         203 non-null    object
 12  embark_town  889 non-null    object
 13  alive        891 non-null    object
 14  alone        891 non-null    bool
dtypes: bool(2), float64(2), int64(4), object(7)
memory usage: 92.4+ KB
```

In [7]:
```
titanic_df.isnull()
```

Out[7]:

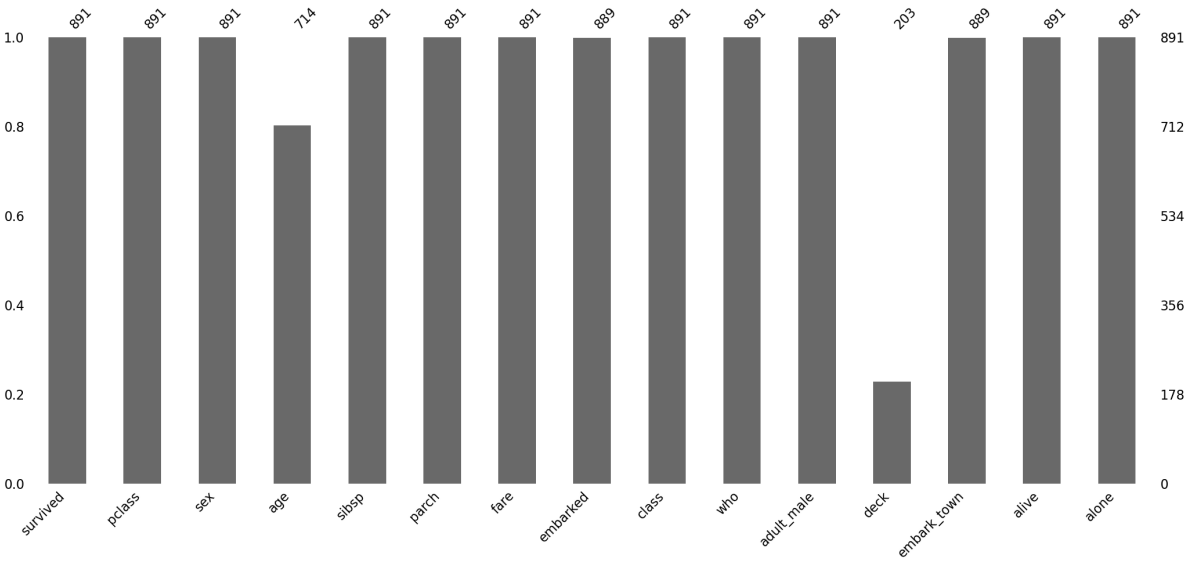
	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck
0	False	False	False	False	False	False	False	False	False	False	False	True
1	False	False	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	False	True
3	False	False	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	False	True
...	...	...	...	...	...	...	...	...	...	...	...	...
886	False	False	False	False	False	False	False	False	False	False	False	True
887	False	False	False	False	False	False	False	False	False	False	False	False
888	False	False	False	True	False	False	False	False	False	False	False	True
889	False	False	False	False	False	False	False	False	False	False	False	False
890	False	False	False	False	False	False	False	False	False	False	False	True

891 rows × 15 columns

2. Visualize the missing values as bar plot and matrix plot using missingno

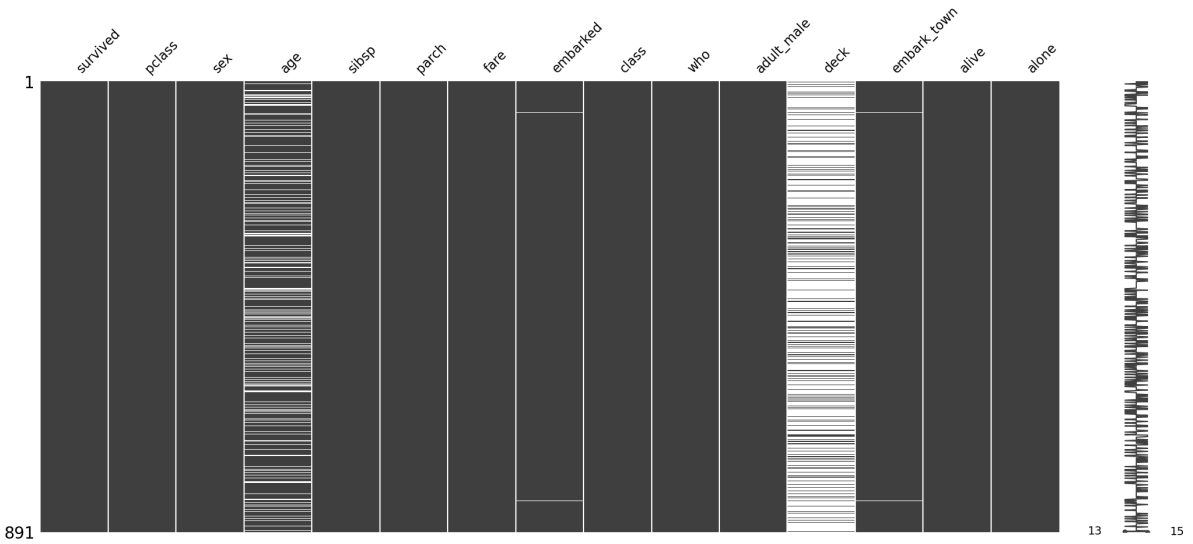
In [11]:
```
msno.bar(titanic_df)
```

Out[11]:
```
<AxesSubplot:>
```



In [12]:
```
msno.matrix(titanic_df)
```

Out[12]:
```
<AxesSubplot:>
```



3.Handle the missing values by deleting data objects.

In [13]:
```
titanic_df.isnull().sum()
```

Out[13]:
```
survived         0
pclass           0
sex              0
age            177
sibsp            0
parch            0
fare             0
embarked         2
class            0
who              0
adult_male       0
deck           688
embark_town      2
alive            0
alone            0
dtype: int64
```

In [14]:
```
df = titanic_df.dropna(axis=0)
df.isnull().sum()
```

Out[14]:
```
survived       0
pclass         0
sex            0
age            0
sibsp          0
parch          0
fare           0
embarked       0
class          0
who            0
adult_male     0
deck           0
embark_town    0
alive          0
alone          0
dtype: int64
```

In [15]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 182 entries, 1 to 889
Data columns (total 15 columns):
 #   Column       Non-Null Count  Dtype
---  ------       --------------  -----
 0   survived     182 non-null    int64
 1   pclass       182 non-null    int64
 2   sex          182 non-null    object
 3   age          182 non-null    float64
 4   sibsp        182 non-null    int64
 5   parch        182 non-null    int64
 6   fare         182 non-null    float64
 7   embarked     182 non-null    object
 8   class        182 non-null    object
 9   who          182 non-null    object
 10  adult_male   182 non-null    bool
 11  deck         182 non-null    object
 12  embark_town  182 non-null    object
 13  alive        182 non-null    object
 14  alone        182 non-null    bool
dtypes: bool(2), float64(2), int64(4), object(7)
memory usage: 20.3+ KB
```

4.Handle the missing values by deleting attributes

In [16]:
```
titanic_df.columns
```

Out[16]:
```
Index(['survived', 'pclass', 'sex', 'age', 'sibsp', 'parch', 'fare',
       'embarked', 'class', 'who', 'adult_male', 'deck', 'embark_town',
       'alive', 'alone'],
      dtype='object')
```

In [17]:
```
df = titanic_df.drop(['deck'],axis=1)
df.isnull().sum()
```

Out[17]:
```
survived         0
pclass           0
sex              0
age            177
sibsp            0
parch            0
fare             0
embarked         2
class            0
who              0
adult_male       0
embark_town      2
alive            0
alone            0
dtype: int64
```

5.Handle the missing value by imputing the missing values with arbitrary value

In [18]:
```
titanic_df['deck'].unique()
```

Out[18]:
```
array([nan, 'C', 'E', 'G', 'D', 'A', 'B', 'F'], dtype=object)
```

In [19]:
```
titanic_df['deck'].isnull().sum()
```

Out[19]:
```
688
```

In [20]:
```
titanic_df['deck'] = titanic_df['deck'].fillna('C')
```

In [21]:
```
titanic_df['deck'].isnull().sum()
```

Out[21]:
```
0
```

In [22]:
```
titanic_df
```

Out[22]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True
...	...	...	...	...	...	...	...	...	...	...	...
886	0	2	male	27.0	0	0	13.0000	S	Second	man	True
887	1	1	female	19.0	0	0	30.0000	S	First	woman	False
888	0	3	female	NaN	1	2	23.4500	S	Third	woman	False
889	1	1	male	26.0	0	0	30.0000	C	First	man	True
890	0	3	male	32.0	0	0	7.7500	Q	Third	man	True

891 rows × 15 columns

Mean

In [23]:
```
mean = titanic_df['age'].mean()
print(mean)
#Replace the missing values for numerical columns with mean
titanic_df['age'] = titanic_df['age'].fillna(mean)
titanic_df['age']
```

Out[23]:
```
29.69911764705882
0      22.000000
1      38.000000
2      26.000000
3      35.000000
4      35.000000
         ...
886    27.000000
887    19.000000
888    29.699118
889    26.000000
890    32.000000
Name: age, Length: 891, dtype: float64
```

Mode

In [27]:
```
#Replace the missing values for categorical columns with mode
mode = titanic_df['deck'].mode()[0]
print(mode)
titanic_df['deck'] = titanic_df['deck'].fillna(mode)
```

```
C
```

In [28]:
```
titanic_df['deck']
```

Out[28]:
```
0      C
1      C
2      C
3      C
4      C
      ..
886    C
887    B
888    C
889    C
890    C
Name: deck, Length: 891, dtype: object
```

Median

In [29]:
```
titanic_df['age']= titanic_df['age'].fillna(titanic_df['age'].median())
titanic_df['age']
```

Out[29]:
```
0      22.0
1      38.0
2      26.0
3      35.0
4      35.0
       ...
886    27.0
887    19.0
888    28.0
889    26.0
890    32.0
Name: age, Length: 891, dtype: float64
```

6.Handle the missing value by imputing the missing values using forward fill and backward fill

In [31]:
```
titanic_df = pd.read_csv("titanic2.csv")
titanic_df
```

Out[31]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male
0	0.0	3	male	22.0	1	0	7.2500	S	Third	man	True
1	1.0	1	female	38.0	1	0	71.2833	C	First	woman	False
2	NaN	3	female	NaN	0	0	7.9250	S	Third	woman	False
3	NaN	1	female	NaN	1	0	53.1000	S	First	woman	False
4	0.0	3	male	35.0	0	0	8.0500	S	Third	man	True
...	...	...	...	...	...	...	...	...	...	...	...
886	0.0	2	male	27.0	0	0	13.0000	S	Second	man	True
887	1.0	1	female	19.0	0	0	30.0000	S	First	woman	False
888	0.0	3	female	NaN	1	2	23.4500	S	Third	woman	False
889	1.0	1	male	26.0	0	0	30.0000	C	First	man	True
890	0.0	3	male	32.0	0	0	7.7500	Q	Third	man	True

891 rows × 15 columns

In [32]:
```
new_df = titanic_df.fillna(method="ffill")
new_df
```

Out[32]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male
0	0.0	3	male	22.0	1	0	7.2500	S	Third	man	True
1	1.0	1	female	38.0	1	0	71.2833	C	First	woman	False
2	1.0	3	female	38.0	0	0	7.9250	S	Third	woman	False
3	1.0	1	female	38.0	1	0	53.1000	S	First	woman	False
4	0.0	3	male	35.0	0	0	8.0500	S	Third	man	True
...	...	...	...	...	...	...	...	...	...	...	...
886	0.0	2	male	27.0	0	0	13.0000	S	Second	man	True
887	1.0	1	female	19.0	0	0	30.0000	S	First	woman	False
888	0.0	3	female	19.0	1	2	23.4500	S	Third	woman	False
889	1.0	1	male	26.0	0	0	30.0000	C	First	man	True
890	0.0	3	male	32.0	0	0	7.7500	Q	Third	man	True

891 rows × 15 columns

In [33]:
```
new_df = titanic_df.fillna(method="bfill")
new_df
```

Out[33]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male
0	0.0	3	male	22.0	1	0	7.2500	S	Third	man	True
1	1.0	1	female	38.0	1	0	71.2833	C	First	woman	False
2	0.0	3	female	35.0	0	0	7.9250	S	Third	woman	False
3	0.0	1	female	35.0	1	0	53.1000	S	First	woman	False
4	0.0	3	male	35.0	0	0	8.0500	S	Third	man	True
...	...	...	...	...	...	...	...	...	...	...	...
886	0.0	2	male	27.0	0	0	13.0000	S	Second	man	True
887	1.0	1	female	19.0	0	0	30.0000	S	First	woman	False
888	0.0	3	female	26.0	1	2	23.4500	S	Third	woman	False
889	1.0	1	male	26.0	0	0	30.0000	C	First	man	True
890	0.0	3	male	32.0	0	0	7.7500	Q	Third	man	True

891 rows × 15 columns

In [ ]: