```
#P
import pandas as pd

dummy={'A':[1,2,None,4,5],
     'B':[10,54,None,30,50],
     'C':[100,200,None,534,697]}

df=pd.DataFrame(dummy)
df.describe()
```

	A	B	C
count	4.000000	4.000000	4.000000
mean	3.000000	36.000000	382.750000
std	1.825742	20.264912	279.865891
min	1.000000	10.000000	100.000000
25%	1.750000	25.000000	175.000000
50%	3.000000	40.000000	367.000000
75%	4.250000	51.000000	574.750000
max	5.000000	54.000000	697.000000

```
print("Original Data Frame \n", df)
print("\n Handling Missing VAlues")

dropset = df.dropna()
print("\n AFTER DROPPING VALUES ARE: \n", dropset)

fillset = df.fillna(df.mean())
print("\n AFTER FILLING MISSING WITH MEAN: \n", fillset)

fillset_median = df.fillna(df.median())
print("\n AFTER FILLING MISSING WITH MEDIAN: \n", fillset_median)
```

```
Original Data Frame
      A     B      C
0  1.0  10.0  100.0
1  2.0  54.0  200.0
2  NaN   NaN    NaN
3  4.0  30.0  534.0
4  5.0  50.0  697.0

 Handling Missing VAlues

 AFTER DROPPING VALUES ARE:
      A     B      C
0  1.0  10.0  100.0
1  2.0  54.0  200.0
3  4.0  30.0  534.0
4  5.0  50.0  697.0

 AFTER FILLING MISSING WITH MEAN:
      A     B       C
0  1.0  10.0  100.00
1  2.0  54.0  200.00
2  3.0  36.0  382.75
3  4.0  30.0  534.00
4  5.0  50.0  697.00

 AFTER FILLING MISSING WITH MEDIAN:
      A     B      C
0  1.0  10.0  100.0
1  2.0  54.0  200.0
2  3.0  40.0  367.0
3  4.0  30.0  534.0
4  5.0  50.0  697.0
```

```
import pandas as pd
import numpy as np
data={'A':[1,2,3,4,5],
     'B':[10,20,30,40,50],
     'C':[100,200,300,400,500]}

df_outliers=pd.DataFrame(data)
z_scores = np.abs((df_outliers - df_outliers.mean())/df_outliers.std())
outliers = (z_scores > 3 ).any(axis=1)

df_no_outliers = df_outliers[~outliers]

print("original Dataframe with outliers:")
print(df_outliers)
print("\n Handling Outliers:")
print("DataFrame without Outliers:")
print(df_no_outliers)
```

```
original Dataframe with outliers:
   A   B    C
0  1  10  100
1  2  20  200
2  3  30  300
3  4  40  400
4  5  50  500

 Handling Outliers:
DataFrame without Outliers:
   A   B    C
0  1  10  100
1  2  20  200
2  3  30  300
3  4  40  400
4  5  50  500
```

Start coding or generate with AI.