Answers to Question Set 7

Date: 28/04/2020 Name: D.Saravanan

1. Print the total missing values in the dataset.

   Program:

```
import numpy as np
import pandas as pd

df = pd.read_csv("DataSet/Building_Permits.csv")
print("The total missing values in the dataset: {}".format(df.isnull().sum().sum()))
```

   Output:

```
The total missing values in the dataset: 2245941
```

2. Print the percentage of missing values in the dataset column wise.

   Program:

```
import numpy as np
import pandas as pd

df = pd.read_csv("DataSet/Building_Permits.csv")
print(df.isnull().sum()*100/len(df))
```

   Output:

```
Permit Number                          0.000000
Permit Type                            0.000000
Permit Type Definition                 0.000000
Permit Creation Date                   0.000000
Block                                  0.000000
Lot                                    0.000000
Street Number                          0.000000
Street Number Suffix                  98.885872
Street Name                            0.000000
Street Suffix                          1.391654
Unit                                  85.178984
Unit Suffix                           99.014077
Description                            0.145802
Current Status                         0.000000
Current Status Date                    0.000000
Filed Date                             0.000000
Issued Date                            7.511312
Completed Date                        51.135747
First Construction Document Date       7.514329
Structural Notification               96.519859
Number of Existing Stories            21.510307
Number of Proposed Stories            21.552539
Voluntary Soft-Story Retrofit         99.982403
Fire Only Permit                      90.534439
Permit Expiration Date                26.083459
Estimated Cost                        19.138260
Revised Cost                           3.049774
Existing Use                          20.670689
Existing Units                        25.911513
Proposed Use                          21.336853
Proposed Units                        25.596280
Plansets                              18.757667
```

```
TIDF Compliance                            99.998994
Existing Construction Type                 21.802916
Existing Construction Type Description     21.802916
Proposed Construction Type                 21.700352
Proposed Construction Type Description     21.700352
Site Permit                                97.305681
Supervisor District                         0.863248
Neighborhoods - Analysis Boundaries         0.867270
Zipcode                                     0.862745
Location                                    0.854701
Record ID                                   0.000000
dtype: float64
```

3. Print the percentage of total unit missing values.

   Program:

```
import numpy as np
import pandas as pd

df = pd.read_csv("DataSet/Building_Permits.csv")
percent = (df.isnull().sum().sum() * 100)/(len(df.index)*len(df.columns))
print("Percentage of total missing values: {:.2f}".format(percent))
```

   Output:

```
Percentage of total missing values: 26.26
```

4. Remove all rows which are having at the least one null values in it, what is the total number of rows after dropping and what is the percentage of rows lost due to this operation.

   Program:

```
import numpy as np
import pandas as pd

df = pd.read_csv("DataSet/Building_Permits.csv")
rdf = df.dropna(axis=0)
print("The total number of rows after dropping: {}".format(len(rdf.index)))
print("The percentage of rows lost due to this operation: {:.1f}".format((len(df.index)-
    len(rdf.index))*100/len(df.index)))
```

   Output:

```
The total number of rows after dropping: 0
The percentage of rows lost due to this operation: 100.0
```

5. Remove all columns which are having at the least one null values in it. What is the total number of columns after dropping and what is the percentage of columns lost due to this operation.

   Program:

```
import numpy as np
import pandas as pd

df = pd.read_csv("DataSet/Building_Permits.csv")
cdf = df.dropna(axis=1)
print("The total number of columns after dropping: {}".format(len(cdf.columns)))
print("The percentage of columns lost due to this operation: {:.1f}".format((len(df.
    columns)-len(cdf.columns))*100/len(df.columns)))
```

Output:

```
The total number of columns after dropping: 12
The percentage of columns lost due to this operation: 72.1
```

6. Print all the columns name separated by a line.

Program:

```
import numpy as np
import pandas as pd

df = pd.read_csv("DataSet/Building_Permits.csv")
for name in df.columns: print(name)
```

Output:

```
Permit Number
Permit Type
Permit Type Definition
Permit Creation Date
Block
Lot
Street Number
Street Number Suffix
Street Name
Street Suffix
Unit
Unit Suffix
Description
Current Status
Current Status Date
Filed Date
Issued Date
Completed Date
First Construction Document Date
Structural Notification
Number of Existing Stories
Number of Proposed Stories
Voluntary Soft-Story Retrofit
Fire Only Permit
Permit Expiration Date
Estimated Cost
Revised Cost
Existing Use
Existing Units
Proposed Use
Proposed Units
Plansets
TIDF Compliance
Existing Construction Type
Existing Construction Type Description
Proposed Construction Type
Proposed Construction Type Description
Site Permit
Supervisor District
Neighborhoods - Analysis Boundaries
Zipcode
Location
Record ID
```

7. Extract a sample of 100 rows at random from the dataset.

Program:

```
import numpy as np
import pandas as pd

df = pd.read_csv("DataSet/Building_Permits.csv")
randf = df.sample(n = 100)
print("Number of (rows, columns): {}".format(randf.shape))
```

Output:

```
Number of (rows, columns): (100, 43)
```

8. Replace all the NA/Null values in the previous ouput (sample dataset) with the values that came directly after it in the same column.

Program:

```
import numpy as np
import pandas as pd

df = pd.read_csv("DataSet/Building_Permits.csv")
randf = df.sample(n=100)
fildf = randf.fillna(method='bfill')

print("Number of NA values before replacing with the next value: {}".format(randf.isnull
    ().sum().sum()))
print("Number of NA values after replacing with the next value: {}".format(fildf.isnull()
    .sum().sum()))
```

Output:

```
Number of NA values before replacing with the next value: 1164
Number of NA values after replacing with the next value: 408
```

9. In continuation with the previous step fill the remaining missing values with zeros.

Program:

```
import numpy as np
import pandas as pd

df = pd.read_csv("DataSet/Building_Permits.csv")
randf = df.sample(n = 100)
fildf = randf.fillna(method = 'bfill')
findf = fildf.replace(to_replace = np.nan, value = 0)

print("Number of NA values before replacing with the next value: {}".format(randf.isnull
    ().sum().sum()))
print("Number of NA values after replacing with the next value: {}".format(fildf.isnull()
    .sum().sum()))
print("Number of missing values after replacing NA with value 0: {}".format(findf.isnull
    ().sum().sum()))
```

Output:

```
Number of NA values before replacing with the next value: 1164
Number of NA values after replacing with the next value: 408
Number of missing values after replacing NA with value 0: 0
```

10. Print the missing values count of the cleaned sample dataset.

Program:

```
import numpy as np
import pandas as pd

df = pd.read_csv("DataSet/Building_Permits.csv")
randf = df.sample(n = 100)
fildf = randf.fillna(method = 'bfill')
findf = fildf.replace(to_replace = np.nan, value = 0)

print("Number of missing values of the cleaned sample dataset: {}".format(findf.isnull().
    sum().sum()))
```

Output:

```
Number of missing values of the cleaned sample dataset: 0
```