Pransshu G

21BTRCD019

Data Preprocessing and Feature Engineering

In [1]:
```
import numpy as np
import pandas as pd
```

In [2]:
```
#pandas operations
#conditional selection
#access rows and cols
#add and delete columns
#concatenation merging and joining dataframe
#handling missing values
#access()
```

In [6]:
```
d = {'USN':[100,101,102],
     'Name':['jerry','tom','ramesh'],
     'Mobile':[99,98,77],
     'marks':[30,35,32]
    }
```

In [7]:
```
std = pd.DataFrame(d)
```

In [8]:
```
std
```

	USN	Name	Mobile	marks
0	100	jerry	99	30
1	101	tom	98	35
2	102	ramesh	77	32

In [11]:
```
#dataframe operations
std.head(2)
#head - displaying rows
#display all by default
#parameter passed, display rows as per parameters
```

	USN	Name	Mobile	marks
0	100	jerry	99	30
1	101	tom	98	35

In [13]:
```
std.columns
#list of columns in df
```

```
Index(['USN', 'Name', 'Mobile', 'marks'], dtype='object')
```

In [14]:
```
std.info()
#gives all info about the df
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3 entries, 0 to 2
Data columns (total 4 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   USN     3 non-null      int64 
 1   Name    3 non-null      object
 2   Mobile  3 non-null      int64 
 3   marks   3 non-null      int64 
dtypes: int64(3), object(1)
memory usage: 224.0+ bytes
```

In [17]:
```
std.isnull()
#any missing values
```

	USN	Name	Mobile	marks
0	False	False	False	False
1	False	False	False	False
2	False	False	False	False

In [19]:
```
#access columns
std['USN']
```

```
0    100
1    101
2    102
Name: USN, dtype: int64
```

In [20]:
```
#accessing mulitple columns
std[['USN','marks']]
```

	USN	marks
0	100	30
1	101	35
2	102	32

In [21]:
```
#access rows
std.loc[1]
#returns row
```

```
USN       101
Name      tom
Mobile     98
marks      35
Name: 1, dtype: object
```

In [22]:
```
#create a dataframe cars having the attributes car id, car name.
#should have 5 values
```

In [55]:
```
car = {
    'car_id':[5,6,7,8,9],
    'car_name':['Alto','Wagonr','Fortuner','Ertiga',np.nan]
}
```

In [56]:
```
df = pd.DataFrame(car)
```

In [57]:
```
df.head()
```

	car_id	car_name
0	5	Alto
1	6	Wagonr
2	7	Fortuner
3	8	Ertiga
4	9	NaN

In [58]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5 entries, 0 to 4
Data columns (total 2 columns):
 #   Column    Non-Null Count  Dtype 
---  ------    --------------  ----- 
 0   car_id    5 non-null      int64 
 1   car_name  4 non-null      object
dtypes: int64(1), object(1)
memory usage: 208.0+ bytes
```

In [59]:
```
df.columns
```

```
Index(['car_id', 'car_name'], dtype='object')
```

In [60]:
```
df.loc(1)
```

```
<pandas.core.indexing._LocIndexer at 0x235535956d0>
```

In [61]:
```
df.iloc(1)
```

```
<pandas.core.indexing._iLocIndexer at 0x23553524b80>
```

In [62]:
```
df['car_id']
```

```
0    5
1    6
2    7
3    8
4    9
Name: car_id, dtype: int64
```

In [63]:
```
df.loc[1]
```

```
car_id          6
car_name    Wagonr
Name: 1, dtype: object
```

In [64]:
```
df.iloc[0]
```

```
car_id         5
car_name    Alto
Name: 0, dtype: object
```

In [65]:
```
df[['car_id','car_name']]
```

	car_id	car_name
0	5	Alto
1	6	Wagonr
2	7	Fortuner
3	8	Ertiga
4	9	NaN

In [66]:
```
df.isnull()
```

	car_id	car_name
0	False	False
1	False	False
2	False	False
3	False	False
4	False	True

In [67]:
```
df.iloc[4]
```

```
car_id        9
car_name    NaN
Name: 4, dtype: object
```

In [70]:
```
#conditional selection
std[std['marks']<35]
```

	USN	Name	Mobile	marks
0	100	jerry	99	30
2	102	ramesh	77	32

In [71]:
```
df1 = std['marks']<35
```

In [74]:
```
std[df1]['Name']
```

```
0     jerry
2    ramesh
Name: Name, dtype: object
```

In [79]:
```
std[(std['USN']>100) & (std['marks']<40)]
```

	USN	Name	Mobile	marks
1	101	tom	98	35
2	102	ramesh	77	32

In [81]:
```
#adding new columns
s = "bangalore,delhi,kashmir".split(',')
print(s,type(s))
```

```
['bangalore', 'delhi', 'kashmir'] <class 'list'>
```

In [84]:
```
std['Address'] = s
```

In [85]:
```
std
```

	USN	Name	Mobile	marks	Address
0	100	jerry	99	30	bangalore
1	101	tom	98	35	delhi
2	102	ramesh	77	32	kashmir

In [90]:
```
#deleting
std.drop('Address',axis = 1, inplace = True)
```

In [91]:
```
std
```

	USN	Name	Mobile	marks
0	100	jerry	99	30
1	101	tom	98	35
2	102	ramesh	77	32

In [92]:
```
df1.isnull()
```

```
0          False
1          False
2          False
Address    False
Name: marks, dtype: bool
```

In [93]:
```
std.isnull()
```

	USN	Name	Mobile	marks
0	False	False	False	False
1	False	False	False	False
2	False	False	False	False

In [101]:
```
a = {
    'USN':[None,None,102],
    'Name':[None,None,'ramesh'],
    'Mobile':[99,98,97],
    'marks':[38,32,37]
}
```

In [102]:
```
a1 = pd.DataFrame(a)
```

In [103]:
```
a1
```

	USN	Name	Mobile	marks
0	NaN	None	99	38
1	NaN	None	98	32
2	102.0	ramesh	97	37

In [104]:
```
a1.isnull()
```

	USN	Name	Mobile	marks
0	True	True	False	False
1	True	True	False	False
2	False	False	False	False

In [105]:
```
a1.dropna(thresh = 2)
```

	USN	Name	Mobile	marks
0	NaN	None	99	38
1	NaN	None	98	32
2	102.0	ramesh	97	37

In [ ]: