```
1
    import numpy as np
2
    import pandas as pd
    Mass = pd.Series([0.3,4,5,0.6,1000,560,80,102,0.01],index=['merc','ven','eart','mar',
2
    Mass
    merc
               0.30
               4.00
    ven
    eart
               5.00
    mar
               0.60
    jup
            1000.00
             560.00
    sat
    ura
              80.00
             102.00
    nept
               0.01
    plu
    dtype: float64
    Mass+Mass
               0.60
C→
    merc
               8.00
    ven
              10.00
    eart
    mar
               1.20
    jup
            2000.00
            1120.00
    sat
             160.00
    ura
    nept
             204.00
    plu
               0.02
    dtype: float64
1 Mass*Mass
                  0.0900
    merc
    ven
                 16.0000
                 25.0000
    eart
                  0.3600
    mar
    jup
            1000000.0000
             313600.0000
    sat
               6400.0000
    ura
    nept
              10404.0000
                  0.0001
    plu
    dtype: float64
1 BigMass = Mass[Mass>100]
2 BigMass
            1000.0
    jup
             560.0
    sat
    nept
             102.0
    dtype: float64
1 NewMass = Mass+BigMass
```

2 NewMass #nan bcz bigmass had nan

```
NaN
   eart
            2000.0
   jup
   mar
               NaN
   merc
               NaN
             204.0
   nept
   plu
               NaN
            1120.0
   sat
               NaN
   ura
               NaN
   ven
   dtype: float64
1 pd
    <module 'pandas' from '/usr/local/lib/python3.7/dist-packages/pandas/__init__.py'>
1 print(pd)
    <module 'pandas' from '/usr/local/lib/python3.7/dist-packages/pandas/__init__.py'>
1 pd.isnull(NewMass)
   eart
             True
            False
   jup
             True
   mar
   merc
             True
   nept
            False
            True
   plu
   sat
            False
             True
   ura
   ven
             True
   dtype: bool
1 NewMass[pd.isnull(NewMass)]
   eart
           NaN
           NaN
   mar
           NaN
   merc
   plu
           NaN
           NaN
   ura
   ven
           NaN
   dtype: float64
1 NewMass[~pd.isnull(NewMass)]
            2000.0
   jup
             204.0
   nept
            1120.0
   sat
   dtype: float64
1 Mass['eris']=0.00029
2 Mass
               0.30000
   merc
```

```
4.00000
   ven
   eart
               5.00000
   mar
               0.60000
            1000.00000
   jup
             560.00000
   sat
   ura
              80.00000
             102.00000
   nept
   plu
               0.01000
   eris
               0.00029
   dtype: float64
1 Mass['moon']= 0.7
```

2 Mass 0.30000 merc

ven 4.00000 5.00000 eart mar 0.60000 1000.00000 jup 560.00000 sat 80.00000 ura 102.00000 nept 0.01000 plu eris 0.00029 moon 0.70000 dtype: float64

```
1 Mass.drop['moon'] #braces error
```

2 Mass

```
Traceback (most recent call last)
TypeError
<ipython-input-15-483e0ea56531> in <module>()
----> 1 Mass.drop['moon'] #braces error
```

TypeError: 'method' object is not subscriptable

SEARCH STACK OVERFLOW

```
1 Mass=Mass.drop(['moon']) #just Mass.drop wont drop , i have to save
```

## 1 Mass

```
0.30000
merc
ven
           4.00000
           5.00000
eart
           0.60000
mar
        1000.00000
jup
         560.00000
sat
          80.00000
ura
         102.00000
nept
plu
           0.01000
eris
           0.00029
dtype: float64
```

```
1 diameter=pd.Series((4000,12000,12000,3000,6000,142000,120000,51000,49000,2000),index=[
2 diameter
              4000
   merc
             12000
   ven
             12000
   eart
              3000
   mar
   jup
              6000
   sat
            142000
            120000
   ura
   nept
             51000
             49000
   plu
   eris
              2000
   dtype: int64
1 #density = mass / volume
1 density=Mass/((np.pi*(diameter**3))/6)
2 density
            8.952466e-12
   merc
   ven
            4.420971e-12
            5.526213e-12
   eart
            4.244132e-11
   mar
   jup
            8.841941e-09
            3.735290e-13
   sat
            8.841941e-14
   ura
            1.468558e-12
   nept
   plu
            1.623354e-16
   eris
            6.923240e-14
   dtype: float64
1 Mass
   merc
               0.30000
               4.00000
   ven
   eart
               5.00000
               0.60000
   mar
            1000.00000
   jup
             560.00000
   sat
   ura
              80.00000
             102.00000
   nept
   plu
               0.01000
   eris
               0.00029
   dtype: float64
1 Mass["unidentified"]=6
1 Mass
                       0.30000
   merc
                       4.00000
   ven
   eart
                       5.00000
```

```
0.60000
   mar
   jup
                    1000.00000
    sat
                     560.00000
   ura
                      80.00000
                     102.00000
   nept
   plu
                       0.01000
   eris
                       0.00029
   unidentified
                       6.00000
   dtype: float64
1 density
            8.952466e-12
   merc
   ven
            4.420971e-12
            5.526213e-12
   eart
            4.244132e-11
   mar
            8.841941e-09
   jup
   sat
            3.735290e-13
   ura
            8.841941e-14
   nept
            1.468558e-12
   plu
            1.623354e-16
            6.923240e-14
   eris
   dtype: float64
1 density=Mass/((np.pi*(diameter**3))/6)
2 density
                    5.526213e-12
   eart
   eris
                    6.923240e-14
                    8.841941e-09
   jup
   mar
                    4.244132e-11
   merc
                    8.952466e-12
                    1.468558e-12
   nept
                    1.623354e-16
   plu
                    3.735290e-13
   sat
   unidentified
                             NaN
                    8.841941e-14
   ura
                    4.420971e-12
   ven
   dtype: float64
1 #replacing nan values
1 Density_mean=np.mean(density)
2 Density_mean
   8.90528215144313e-10
1 density.index
    Index(['eart', 'eris', 'jup', 'mar', 'merc', 'nept', 'plu', 'sat',
           'unidentified', 'ura', 'ven'],
          dtype='object')
```

1 for i in density.index :

```
4/20/22, 1:57 PM
                                                  1904-2004pandas - Colaboratory
     2
            if pd.isnull(density[i]) :
                density[i]= Density_mean
     1 density
         eart
                            5.526213e-12
         eris
                           6.923240e-14
                            8.841941e-09
         jup
                           4.244132e-11
         mar
```

1.623354e-16 plu sat 3.735290e-13 unidentified 8.905282e-10

8.952466e-12

1.468558e-12

8.841941e-14 ura 4.420971e-12 ven

dtype: float64

## 1 Mass

merc

nept

merc 0.30000 4.00000 ven eart 5.00000 mar 0.60000 1000.00000 jup 560.00000 sat ura 80.00000 nept 102.00000 plu 0.01000 eris 0.00029 unidentified 6.00000

1 Mass["unidentified2"]=6

dtype: float64

## 1 Mass

0.30000 merc 4.00000 ven 5.00000 eart mar 0.60000 jup 1000.00000 sat 560.00000 ura 80.00000 nept 102.00000 plu 0.01000 eris 0.00029 unidentified 6.00000 unidentified2 6.00000 dtype: float64

1 density=Mass/((np.pi\*(diameter\*\*3))/6)

2 density

eart 5.526213e-12

6.923240e-14 eris jup 8.841941e-09 4.244132e-11 mar 8.952466e-12 merc 1.468558e-12 nept plu 1.623354e-16 sat 3.735290e-13 unidentified NaN unidentified2 NaN ura 8.841941e-14 4.420971e-12 ven

dtype: float64

## 1 pd.isnull(density)

eart False eris False jup False False mar merc False False nept plu False False unidentified True unidentified2 True ura False False ven

dtype: bool

- density[pd.isnull(density)] = np.mean(density)
- 2 density

eart 5.526213e-12 eris 6.923240e-14 8.841941e-09 jup 4.244132e-11 mar merc 8.952466e-12 nept 1.468558e-12 plu 1.623354e-16 sat 3.735290e-13 unidentified 8.905282e-10 unidentified2 8.905282e-10 ura 8.841941e-14 ven 4.420971e-12

dtype: float64

 $https://colab.research.google.com/drive/1vQ7gBQKkTb3tQWP\_MUM5cW0N4fc0JH7m\#scrollTo=QEtkGNomEUsP\&printMode=true$