```
1 import numpy as np
```

```
1 arr1=np.arange(6)
2 arr1
```

```
array([0, 1, 2, 3, 4, 5])
```

```
1 arr1=arr1.reshape(3,2)
2 arr1
```

```
array([[0, 1],
       [2, 3],
       [4, 5]])
```

```
1 arr2=np.arange(6).reshape(3,2)
2 arr2
```

```
array([[0, 1],
       [2, 3],
       [4, 5]])
```

```
1 arr1+arr2
```

```
array([[ 0,  2],
       [ 4,  6],
       [ 8, 10]])
```

```
1 arr4=arr1[0,:] #0,0,: wrong
2 arr4
```

```
array([0, 1])
```

```
1 arr5=arr1*arr4
2 arr5
```

```
array([[0, 1],
       [0, 3],
       [0, 5]])
```

```
1 arr4
```

```
array([0, 1])
```

```
1 arr4[0]
```

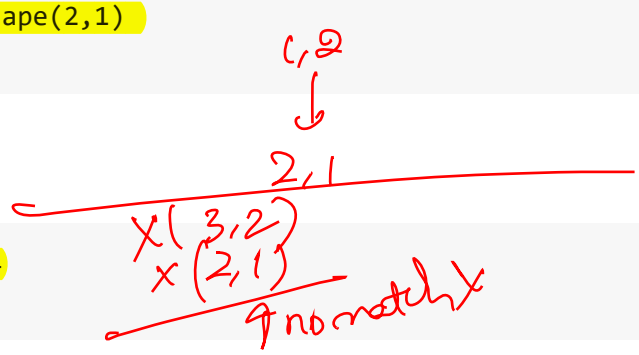
```
0
```

```
1 arr4[1]
```

```
1
```

```
arr4=arr4.reshape(2,1)
arr4
```

```
array([[0],
       [1]])
```



```
arr6=arr1*arr4
arr6
```

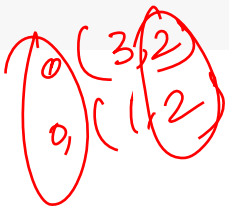
```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-26-523a856a42d3> in <module>()
----> 1 arr6=arr1*arr4
      2 arr6

ValueError: operands could not be broadcast together with shapes (3,2) (2,1)
```

SEARCH STACK OVERFLOW

```
arr6=arr1*(arr4.reshape(1,2))
arr6
```

```
array([[0, 1],
       [0, 3],
       [0, 5]])
```



	MERCURY	VENUS	EARTH	MARS	JUPITER	SATURN	URANUS	NEPTUNE	PLUTO
Mass	0.330	4.87	5.97	0.642	1898	568	86.8	102	0.0146
Diameter	57.9	108.2	149.6	227.9	778.6	1433.5	2872.5	4495.1	5906.4
DayLength	4222.6	2802.0	24.0	24.7	9.9	10.7	17.2	16.1	153.3

```
Planet_Small = np.loadtxt(planets_small.txt)
```

```
1 Planet_Small = np.loadtxt("planets_small.txt",skiprows=1)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-46-7f3fea7baec3> in <module>()
----> 1 Planet_Small = np.loadtxt("planets_small.txt",skiprows=1)

3 frames
/usr/local/lib/python3.7/dist-packages/numpy/lib/npyio.py in floatconv(x)
    734         if '0x' in x:
    735             return float.fromhex(x)
--> 736         return float(x)
    737
    738     typ = dtype.type

ValueError: could not convert string to float: 'Mass'
```

SEARCH STACK OVERFLOW

```
1 Planet_Small = np.loadtxt("planets_small.txt",skiprows=1,usecols=(1,2,3,4,5,6,7,8,9))
2 Planet_Small
3 # can also use (range(1,9)) instead of (1,2,3,4,5,6,7,8,9)
```

```
array([[3.3000e-01, 4.8700e+00, 5.9700e+00, 6.4200e-01, 1.8980e+03,
        5.6800e+02, 8.6800e+01, 1.0200e+02],
       [5.7900e+01, 1.0820e+02, 1.4960e+02, 2.2790e+02, 7.7860e+02,
        1.4335e+03, 2.8725e+03, 4.4951e+03],
       [4.2226e+03, 2.8020e+03, 2.4000e+01, 2.4700e+01, 9.9000e+00,
        1.0700e+01, 1.7200e+01, 1.6100e+01]])
```

```
1 Planet_Small.shape
```

```
(3, 9)
```

```
1 Planet_Small.size
```

```
27
```

```
1 Planet_Small.dtype
```

```
dtype('float64')
```

```
1 Planet_Small.ndim
```

```
2
```

```
1 Planet_Small.ndim
```

```
2
```

MERCURY VENUS EARTH MOON MARS JUPITER SATURN URANUS NEPTUNE PLUTO

Mass(1024kg) 0.330 4.87 5.97 0.073 0.642 1898 568 86.8 102 0.0146

Diameter(km) 4879 12104 12756 3475 6792 142984 120536 51118 49528 2370

Density(kg/m3) 5427 5243 5514 3340 3933 1326 687 1271 1638 2095

Gravity(m/s2) 3.7 8.9 9.8 1.6 3.7 23.1 9.0 8.7 11.0 0.7

EscapeVelocity(km/s) 4.3 10.4 11.2 2.4 5.0 59.5 35.5 21.3 23.5 1.3

RotationPeriod(hours) 1407.6 -5832.5 23.9 655.7 24.6 9.9 10.7 -17.2 16.1 -153.3

LengthofDay(hours) 4222.6 2802.0 24.0 708.7 24.7 9.9 10.7 17.2 16.1 153.3
DistancefromSun(106km) 57.9 108.2 149.6 0.384 227.9 778.6 1433.5 2872.5 4495.1 5906.4
Perihelion(106km) 46.0 107.5 147.1 0.363 206.6 740.5 1352.6 2741.3 4444.5 4436.8
Aphelion(106km) 69.8 108.9 152.1 0.406 249.2 816.6 1514.5 3003.6 4545.7 7375.9
OrbitalPeriod(days) 88.0 224.7 365.2 27.3 687.0 4331 10747 30589 59800 90560
OrbitalVelocity(km/s) 47.4 35.0 29.8 1.0 24.1 13.1 9.7 6.8 5.4 4.7 OrbitalInclination(degrees) 7.0 3.4 0.0 5.1 1.9 1.3 2.5 0.8 1.8 17.2 OrbitalEccentricity 0.205 0.007 0.017 0.055 0.094 0.049 0.057 0.046 0.011 0.244 ObliquitytoOrbit(degrees) 0.034 177.4 23.4 6.7 25.2 3.1 26.7 97.8 28.3 122.5
MeanTemperature(C) 167 464 15 -20 -65 -110 -140 -195 -200 -225 SurfacePressure(bars) 0 92 1 0 0.01 Unknown Unknown Unknown Unknown 0.00001 NumberofMoons 0 0 1 0 2 79 82 27 14 5
RingSystem? 0 0 0 0 0 1 1 1 1 0 GlobalMagneticField? 1 0 1 0 0 1 1 1 1 Unknown

```
1  Planet = np.loadtxt("planets.txt",skiprows=1,usecols=(1,2,3,4,5,6,7,8,9),)
2  Planet
3  #error bcz  data has unknown in colum
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-63-0ace21ffff6d> in <module>()
----> 1 Planet = np.loadtxt("planets.txt",skiprows=1,usecols=(1,2,3,4,5,6,7,8,9))
      2 Planet
```

3 frames

```
1 Planet = np.loadtxt("planets.txt",skiprows=1,usecols=(1,2,3,4,5,6,7,8,9))
2 Planet
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-66-0ace21ffff6d> in <module>()
----> 1 Planet = np.loadtxt("planets.txt",skiprows=1,usecols=(1,2,3,4,5,6,7,8,9))
      2 Planet
```

3 frames

```
/usr/local/lib/python3.7/dist-packages/numpy/lib/npyio.py in floatconv(x)
    734         if '0x' in x:
    735             return float.fromhex(x)
--> 736         return float(x)
    737
    738     typ = dtype.type

ValueError: could not convert string to float: 'Unknown'
```

SEARCH STACK OVERFLOW

To overcom this unknown issue

```
1 planetsnew = np.genfromtxt("planets.txt",skip_header=1,usecols=range(0,9))
2 planetsnew
```

```
        8.68000e+01],
       [        nan,  4.87900e+03,  1.21040e+04,  1.27560e+04,
        3.47500e+03,  6.79200e+03,  1.42984e+05,  1.20536e+05,
        5.11180e+04],
       [        nan,  5.42700e+03,  5.24300e+03,  5.51400e+03,
        3.34000e+03,  3.93300e+03,  1.32600e+03,  6.87000e+02,
        1.27100e+03],
       [        nan,  3.70000e+00,  8.90000e+00,  9.80000e+00,
        1.60000e+00,  3.70000e+00,  2.31000e+01,  9.00000e+00,
        8.70000e+00],
       [        nan,  4.30000e+00,  1.04000e+01,  1.12000e+01,
        2.40000e+00,  5.00000e+00,  5.95000e+01,  3.55000e+01,
        2.13000e+01],
       [        nan,  1.40760e+03, -5.83250e+03,  2.39000e+01,
        6.55700e+02,  2.46000e+01,  9.90000e+00,  1.07000e+01,
       -1.72000e+01],
       [        nan,  4.22260e+03,  2.80200e+03,  2.40000e+01,
        7.08700e+02,  2.47000e+01,  9.90000e+00,  1.07000e+01,
        1.72000e+01],
       [        nan,  5.79000e+01,  1.08200e+02,  1.49600e+02,
        3.84000e-01,  2.27900e+02,  7.78600e+02,  1.43350e+03,
        2.87250e+03],
       [        nan,  4.60000e+01,  1.07500e+02,  1.47100e+02,

        3.63000e-01,  2.06600e+02,  7.40500e+02,  1.35260e+03,
        2.74130e+03],
       [        nan,  6.98000e+01,  1.08000e+02,  1.52100e+02,
```

```
array([[        nan,  6.98000e+01,  1.08900e+02,  1.52100e+02,
         4.06000e-01,  2.49200e+02,  8.16600e+02,  1.51450e+03,
         3.00360e+03],
       [        nan,  8.80000e+01,  2.24700e+02,  3.65200e+02,
         2.73000e+01,  6.87000e+02,  4.33100e+03,  1.07470e+04,
         3.05890e+04],
       [        nan,  4.74000e+01,  3.50000e+01,  2.98000e+01,
         1.00000e+00,  2.41000e+01,  1.31000e+01,  9.70000e+00,
         6.80000e+00],
       [        nan,  7.00000e+00,  3.40000e+00,  0.00000e+00,
         5.10000e+00,  1.90000e+00,  1.30000e+00,  2.50000e+00,
         8.00000e-01],
       [        nan,  2.05000e-01,  7.00000e-03,  1.70000e-02,
         5.50000e-02,  9.40000e-02,  4.90000e-02,  5.70000e-02,
         4.60000e-02],
       [        nan,  3.40000e-02,  1.77400e+02,  2.34000e+01,
         6.70000e+00,  2.52000e+01,  3.10000e+00,  2.67000e+01,
         9.78000e+01],
       [        nan,  1.67000e+02,  4.64000e+02,  1.50000e+01,
        -2.00000e+01, -6.50000e+01, -1.10000e+02, -1.40000e+02,
        -1.95000e+02],
       [        nan,  0.00000e+00,  9.20000e+01,  1.00000e+00,
         0.00000e+00,  1.00000e-02,         nan,         nan,
                nan],
       [        nan,  0.00000e+00,  0.00000e+00,  1.00000e+00,
         0.00000e+00,  2.00000e+00,  7.90000e+01,  8.20000e+01,
         2.70000e+01],
       [        nan,  0.00000e+00,  0.00000e+00,  0.00000e+00,
         0.00000e+00,  0.00000e+00,  1.00000e+00,  1.00000e+00,
         1.00000e+00],
       [        nan,  1.00000e+00,  0.00000e+00,  1.00000e+00,
         0.00000e+00,  0.00000e+00,  1.00000e+00,  1.00000e+00,
         1.00000e+00]])
```

```
1
```

```
1 planetsnew = np.genfromtxt("planets.txt",skip_header=1,usecols=range(1,9))
2 planetsnew
```

```
array([[ 3.30000e-01,  4.87000e+00,  5.97000e+00,  7.30000e-02,
         6.42000e-01,  1.89800e+03,  5.68000e+02,  8.68000e+01],
       [ 4.87900e+03,  1.21040e+04,  1.27560e+04,  3.47500e+03,
         6.79200e+03,  1.42984e+05,  1.20536e+05,  5.11180e+04],
       [ 5.42700e+03,  5.24300e+03,  5.51400e+03,  3.34000e+03,
         3.93300e+03,  1.32600e+03,  6.87000e+02,  1.27100e+03],
       [ 3.70000e+00,  8.90000e+00,  9.80000e+00,  1.60000e+00,
         3.70000e+00,  2.31000e+01,  9.00000e+00,  8.70000e+00],
       [ 4.30000e+00,  1.04000e+01,  1.12000e+01,  2.40000e+00,
         5.00000e+00,  5.95000e+01,  3.55000e+01,  2.13000e+01],
       [ 1.40760e+03, -5.83250e+03,  2.39000e+01,  6.55700e+02,
         2.46000e+01,  9.90000e+00,  1.07000e+01, -1.72000e+01],
       [ 4.22260e+03,  2.80200e+03,  2.40000e+01,  7.08700e+02,
         2.47000e+01,  9.90000e+00,  1.07000e+01,  1.72000e+01],
       [ 5.79000e+01,  1.08200e+02,  1.49600e+02,  3.84000e-01,
         2.27900e+02,  7.78600e+02,  1.43350e+03,  2.87250e+03],
       [ 4.60000e+01,  1.07500e+02,  1.47100e+02,  3.63000e-01,
         2.06600e+02,  7.40500e+02,  1.35260e+03,  2.74130e+03],
       [ 6.98000e+01,  1.08900e+02,  1.52100e+02,  4.06000e-01,
         2.49200e+02,  8.16600e+02,  1.51450e+03,  3.00360e+03],
```

```
       [ 8.80000e+01,  2.24700e+02,  3.65200e+02,  2.73000e+01,
         6.87000e+02,  4.33100e+03,  1.07470e+04,  3.05890e+04],
       [ 4.74000e+01,  3.50000e+01,  2.98000e+01,  1.00000e+00,
         2.41000e+01,  1.31000e+01,  9.70000e+00,  6.80000e+00],
       [ 7.00000e+00,  3.40000e+00,  0.00000e+00,  5.10000e+00,
         1.90000e+00,  1.30000e+00,  2.50000e+00,  8.00000e-01],
       [ 2.05000e-01,  7.00000e-03,  1.70000e-02,  5.50000e-02,
         9.40000e-02,  4.90000e-02,  5.70000e-02,  4.60000e-02],
       [ 3.40000e-02,  1.77400e+02,  2.34000e+01,  6.70000e+00,
         2.52000e+01,  3.10000e+00,  2.67000e+01,  9.78000e+01],
       [ 1.67000e+02,  4.64000e+02,  1.50000e+01, -2.00000e+01,
        -6.50000e+01, -1.10000e+02, -1.40000e+02, -1.95000e+02],
       [ 0.00000e+00,  9.20000e+01,  1.00000e+00,  0.00000e+00,
         1.00000e-02,           nan,           nan,           nan],
       [ 0.00000e+00,  0.00000e+00,  1.00000e+00,  0.00000e+00,
         2.00000e+00,  7.90000e+01,  8.20000e+01,  2.70000e+01],
       [ 0.00000e+00,  0.00000e+00,  0.00000e+00,  0.00000e+00,
         0.00000e+00,  1.00000e+00,  1.00000e+00,  1.00000e+00],
       [ 1.00000e+00,  0.00000e+00,  1.00000e+00,  0.00000e+00,
         0.00000e+00,  1.00000e+00,  1.00000e+00,  1.00000e+00]])
```

```
1 np.isnan(planetsnew)
```

```
array([[False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False,  True,  True,  True],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False],
       [False, False, False, False, False, False, False, False]])
```

```
1 planetsnew=np.nan_to_num(planetsnew,nan=1)
2 planetsnew
```

```
array([[ 3.30000e-01,  4.87000e+00,  5.97000e+00,  7.30000e-02,
         6.42000e-01,  1.89800e+03,  5.68000e+02,  8.68000e+01],
       [ 4.87900e+03,  1.21040e+04,  1.27560e+04,  3.47500e+03,
         6.79200e+03,  1.42984e+05,  1.20536e+05,  5.11180e+04],
       [ 5.42700e+03,  5.24300e+03,  5.51400e+03,  3.34000e+03,
         3.93300e+03,  1.32600e+03,  6.87000e+02,  1.27100e+03],
       [ 3.70000e+00,  8.90000e+00,  9.80000e+00,  1.60000e+00,
         3.70000e+00,  2.31000e+01,  9.00000e+00,  8.70000e+00],
       [ 4.30000e+00,  1.04000e+01,  1.12000e+01,  2.40000e+00,
         5.00000e+00,  5.95000e+01,  3.55000e+01,  2.13000e+01],
       [ 1.40760e+03, -5.83250e+03,  2.39000e+01,  6.55700e+02,
```

```
  2.46000e+01,  9.90000e+00,  1.07000e+01, -1.72000e+01],
[ 4.22260e+03,  2.80200e+03,  2.40000e+01,  7.08700e+02,
  2.47000e+01,  9.90000e+00,  1.07000e+01,  1.72000e+01],
[ 5.79000e+01,  1.08200e+02,  1.49600e+02,  3.84000e-01,
  2.27900e+02,  7.78600e+02,  1.43350e+03,  2.87250e+03],
[ 4.60000e+01,  1.07500e+02,  1.47100e+02,  3.63000e-01,
  2.06600e+02,  7.40500e+02,  1.35260e+03,  2.74130e+03],
[ 6.98000e+01,  1.08900e+02,  1.52100e+02,  4.06000e-01,
  2.49200e+02,  8.16600e+02,  1.51450e+03,  3.00360e+03],
[ 8.80000e+01,  2.24700e+02,  3.65200e+02,  2.73000e+01,
  6.87000e+02,  4.33100e+03,  1.07470e+04,  3.05890e+04],
[ 4.74000e+01,  3.50000e+01,  2.98000e+01,  1.00000e+00,
  2.41000e+01,  1.31000e+01,  9.70000e+00,  6.80000e+00],
[ 7.00000e+00,  3.40000e+00,  0.00000e+00,  5.10000e+00,
  1.90000e+00,  1.30000e+00,  2.50000e+00,  8.00000e-01],
[ 2.05000e-01,  7.00000e-03,  1.70000e-02,  5.50000e-02,
  9.40000e-02,  4.90000e-02,  5.70000e-02,  4.60000e-02],
[ 3.40000e-02,  1.77400e+02,  2.34000e+01,  6.70000e+00,
  2.52000e+01,  3.10000e+00,  2.67000e+01,  9.78000e+01],
[ 1.67000e+02,  4.64000e+02,  1.50000e+01, -2.00000e+01,
 -6.50000e+01, -1.10000e+02, -1.40000e+02, -1.95000e+02],
[ 0.00000e+00,  9.20000e+01,  1.00000e+00,  0.00000e+00,
  1.00000e-02,  1.00000e+00,  1.00000e+00,  1.00000e+00],
[ 0.00000e+00,  0.00000e+00,  1.00000e+00,  0.00000e+00,
  2.00000e+00,  7.90000e+01,  8.20000e+01,  2.70000e+01],
[ 0.00000e+00,  0.00000e+00,  0.00000e+00,  0.00000e+00,
  0.00000e+00,  1.00000e+00,  1.00000e+00,  1.00000e+00],
[ 1.00000e+00,  0.00000e+00,  1.00000e+00,  0.00000e+00,
  0.00000e+00,  1.00000e+00,  1.00000e+00,  1.00000e+00]])
```

```
np.savetxt("frmtplnt.txt",planetsnew,delimiter=",")
```

```
#unix commanda using python
!ls #list directory
```

```
frmtplnt.txt  planets_small.txt  planets.txt  sample_data
```

```
!ls-lh
```

```
/bin/bash: ls-lh: command not found
```

```
!ls-ltr
```

```
/bin/bash: ls-ltr: command not found
```

✓ 0s completed at 1:28 PM