In [2]:
```
import pandas
mydataset = {
  'cars': ["BMW", "Volvo", "Ford"],
  'passings': [3, 7, 2]
}

myvar = pandas.DataFrame(mydataset)

print(myvar)
```

```
    cars  passings
0    BMW         3
1  Volvo         7
2   Ford         2
```

In [8]:
```
import pandas


restaurant = {
    'Veg' : ["dosa","idli"],
    'passings': ["60","40"]
}

a = pandas.DataFrame(restaurant)
print(a)
```

```
    Veg passings
0  dosa       60
1  idli       40
```

In [2]:
```
import pandas
rest=pandas.read_csv("E:\\tiffin.csv")
print(rest.to_string())
```

```
    Veg  Price
0  idli     40
1  dosa     60
2  vada     50
```

Introduction pandas

In [3]:
```
import pandas as pd
details = {
    'categ' : ['name','year', 'place'],
    'det' : ['Hemika',2004,'Visakhapatnam']
}
a=pd.DataFrame(details)
print(a)
```

```
   categ           det
0   name         Hemika
1   year           2004
2  place  Visakhapatnam
```

Note: as is a keyword to use duplicate word instead of pandas can use pd or any other variable

In [5]:
```
#To check pandas version:
import pandas as p
print(p.__version__)
```

```
1.3.4
```

Exporting pandas

In [6]:
```
import pandas
details = {
    'categ' : ['name','year', 'place'],
    'det' : ['Hemika',2004,'Visakhapatnam']
}
a=pandas.DataFrame(details)
filename='details.xlsx'
a.to_excel(filename)
print(a)
```

```
   categ            det
0   name         Hemika
1   year           2004
2  place  Visakhapatnam
```

Pandas series

In [9]:
```
# Keyword Series
import pandas as pd
a=[100,200,300]
a=pd.Series(a)
print(a)
```

```
0    100
1    200
2    300
dtype: int64
```

In [10]:
```
#Labels the values are labeled with their index number
import pandas as pd
a=[100,2000,30000]
a=pd.Series(a)
print(a[0])
```

```
100
```

In [13]:
```
#Creating labels
import pandas as pd
a=[100,2000,30000]
a=pd.Series(a,index=["a","b","c"])
print(a)
print(a["a"])
```

```
a      100
b     2000
c    30000
dtype: int64
100
```

In [12]:
```
# key/value object as seies
# you can also use a key/value object, like a dictionary, when creating a Series
import pandas
calories = {"day1": 420, "day2": 380, "day3": 390}
var = pd.Series(calories)
print(var)
```

```
day1    420
day2    380
day3    390
dtype: int64
```

In [21]:
```
#Data types object more prior than int and float
import pandas as p
a={
    "x":[1,2,"a"],
    "y":[1,2,3],
    "z":["a","b","c"]
}
print(pd.Series(a))
```

```
x    [1, 2, a]
y    [1, 2, 3]
z    [a, b, c]
dtype: object
```

Data Frame

Note: Data sets in Pandas are usually multi-dimensional tables, called DataFrames. Series is like a column, a DataFrame is the whole table.

In [17]:
```
import pandas as pd
var = {
  "calories": [420, 380, 390],
  "duration": [50, 40, 45]
}
v = pd.DataFrame(var)
print(v)
```

```
   calories  duration
0       420        50
1       380        40
2       390        45
```

#locate rows

the DataFrame is like a table with rows and columns. Pandas use the loc attribute to return one or more specified row(s)

In [28]:
```
import pandas as p
a={
    "x":[1,2,"a"],
    "y":[1,2,3],
    "z":["a","b","c"]
}
v=p.DataFrame(a)
print(v)
print(v.loc[0])
print(v.loc[[0, 1]])
```

```
   x  y  z
0  1  1  a
1  2  2  b
2  a  3  c
x    1
y    1
z    a
Name: 0, dtype: object
   x  y  z
0  1  1  a
1  2  2  b
```

In [29]:
```
import pandas as pd
data = {
  "calories": [420, 380, 390],
  "duration": [50, 40, 45]
}
df = pd.DataFrame(data, index = ["day1", "day2", "day3"])
print(df)
```

```
      calories  duration
day1       420        50
day2       380        40
day3       390        45
```

In [ ]: