In [4]:
```
#          pandas:---Reading and Writing Data


# I/O API Tools
# readers and writers
#        Readers           writers
# 1) read_csv              to_csv
#2) read_excel              to_excel
#3) read_hdf                to_hdf
#4) read_sql                to_sql
#5) read_json               to_json
#6) read_html               to_html
#7) read_stata              to_ stata
#8) read_clipboard          to_clipboard
#9) read_pickel             to_pickel
#10)read_msgpack            to_msgpack(experimental)
#11)read_gbq                to_gbq(experimental)
```

In [5]:
```
#  CSV and Textual files
#  CSV----> comma-separated values
#-values in a row are separated by a comma
#----------------------------------------
#(.txt)--->
#  tabular data separeted by spaces contained in text files(.txt) format.
```

In [63]:
```
#  reading data in CSV or Text files

# -create CSV file in excel save as (.csv) format
# - then in jupyter notebook go to documents and upload the csv  file
```

In [25]:
```
#read_csv()

#  1 example
import pandas as pd
a=pd.read_csv("data.csv",sep=',')
a
```

Out[25]:

| | college data | | | |
|---|---|---|---|---|
| 0 | name | college | dob | marks |
| 1 | nikhil | menon | 30/9/1999 | 86 |
| 2 | ketan | menon | 10/2/2000 | 86 |
| 3 | prashant | menon | 4/7/2003 | 67 |
| 4 | jj | 5/3/2011 | NaN | NaN |
| 5 | kk | 7/8/2001 | NaN | NaN |

In [ ]:
```

```

In [51]:
```
# example 2

import pandas as pd
b=pd.read_csv("sampledata.csv",)
b
```

Out[51]:

| | Serial Number | Company Name | Employee Markme | Description | Leave |
|---|---|---|---|---|---|
| **0** | 9.788190e+12 | TALES OF SHIVA | Mark | mark | 0 |
| **1** | 9.780100e+12 | 1Q84 THE COMPLETE TRILOGY | HARUKI MURAKAMI | Mark | 0 |
| **2** | 9.780200e+12 | MY KUMAN | Mark | Mark | 0 |
| **3** | 9.780010e+12 | THE GOD OF SMAAL THINGS | ARUNDHATI ROY | 4TH HARPER COLLINS | 2 |
| **4** | 9.780550e+12 | THE BLACK CIRCLE | Mark | 4TH HARPER COLLINS | 0 |
| **5** | 9.788130e+12 | THE THREE LAWS OF PERFORMANCE | Mark | 4TH HARPER COLLINS | 0 |
| **6** | 9.789380e+12 | CHAMarkKYA MANTRA | Mark | 4TH HARPER COLLINS | 0 |
| **7** | 9.788180e+12 | 59.FLAGS | Mark | 4TH HARPER COLLINS | 0 |
| **8** | 9.780740e+12 | THE POWER OF POSITIVE THINKING FROM | Mark | A & A PUBLISHER | 0 |
| **9** | 9.789380e+12 | YOU CAN IF YO THINK YO CAN | PEALE | A & A PUBLISHER | 0 |
| **10** | 9.788180e+12 | DONGRI SE DUBAI TAK (MPH) | Mark | A & A PUBLISHER | 0 |
| **11** | 9.788190e+12 | MarkLANDA ADYTAN KOSH | Mark | AADISH BOOK DEPOT | 0 |
| **12** | 9.788190e+12 | MarkLANDA VISHAL SHABD SAGAR | - | AADISH BOOK DEPOT | 1 |
| **13** | 8.187776e+09 | MarkLANDA CONCISE DICT(ENG TO HINDI) | Mark | AADISH BOOK DEPOT | 0 |
| **14** | 9.789380e+12 | LIEUTEMarkMarkT GENERAL BHAGAT: A SAGA OF BRAV... | Mark | AAM COMICS | 2 |
| **15** | 9.789380e+12 | LN. MarkIK SUNDER SINGH | N.A | AAN COMICS | 0 |
| **16** | 9.789380e+12 | I AM KRISHMark | DEEP TRIVEDI | AATMAN INNOVATIONS PVT LTD | 1 |
| **17** | 9.789380e+12 | DON'T TEACH ME TOLERANCE INDIA | DEEP TRIVEDI | AATMAN INNOVATIONS PVT LTD | 0 |
| **18** | 9.789380e+12 | MUJHE SAHISHNUTA MAT SIKHAO BHARAT | DEEP TRIVEDI | AATMAN INNOVATIONS PVT LTD | 0 |
| **19** | 9.789380e+12 | SECRETS OF DESTINY | DEEP TRIVEDI | AATMAN INNOVATIONS PVT LTD | 1 |
| **20** | 9.789380e+12 | BHAGYA KE RAHASYA (HINDI) SECRET OF DESTINY | DEEP TRIVEDI | AATMAN INNOVATIONS PVT LTD | 1 |

| | Serial Number | Company Name | Employee Markme | Description | Leave |
|---|---|---|---|---|---|
| **21** | 9.788190e+12 | MEIN MANN HOON | DEEP TRIVEDI | AATMAN INNOVATIONS PVT LTD | 0 |
| **22** | 9.789380e+12 | I AM THE MIND | DEEP TRIVEDI | AATMARAM & SONS | 0 |
| **23** | 9.780350e+12 | THE ART OF CHOOSING | SHEEMark IYENGAR | ABACUS | 0 |
| **24** | 9.780350e+12 | IN SPITE OF THE GODS | EDWARD LUCE | ABACUS | 1 |
| **25** | 9.788190e+12 | QUESTIONS & ANWERS ABOUT THE GREAT BIBLE | Mark | ABC PUBLISHERS DISTRIBUTORS | 4 |
| **26** | 9.789380e+12 | NIBANDH EVAM KAHANI LEKHAN { HINDI } | Mark | ABHI BOOKS | 1 |
| **27** | 9.789330e+12 | INDIAN ECONOMY SINCE INDEPENDENCE 27TH /E | UMA KAPILA | ACADEMIC FOUNDATION | 1 |
| **28** | 9.788170e+12 | ECONOMIC DEVELOPMENT AND POLICY IN INDIA | UMA KAPILA | ACADEMIC FOUNDATION | 1 |
| **29** | 9.789330e+12 | INDIAN ECONOMY PERFORMANCE 18TH/E 2017-2018 | UMA KAPILA | ACADEMIC FOUNDATION | 2 |
| **30** | 9.789330e+12 | INDIAN ECONOMIC DEVELOPMENTSINCE 1947 (NO RETU... | UMA KAPILA | ACADEMIC FOUNDATION | 1 |
| **31** | 9.789380e+12 | PRELIMS SPECIAL READING COMPREHENSION PAPER II... | MarkGENDRA PRATAP | ACCESS PUBLISHING INDIA PVT.LTD | 0 |
| **32** | 9.789380e+12 | THE CONSTITUTION OF INDIA 2ND / E | AR KHAN | ACCESS PUBLISHING INDIA PVT.LTD | 10 |
| **33** | 9.789390e+12 | INDIAN HERITAGE ,ART & CULTURE | MADHUKAR | ACCESS PUBLISHING INDIA PVT.LTD | 10 |
| **34** | 9.789380e+12 | BHARAT KA SAMVIDHAN | AR KHAN | ACCESS PUBLISHING INDIA PVT.LTD | 4 |
| **35** | 9.789380e+12 | ETHICS, INTEGRITY & APTITUDE ( 3RD/E) | P N ROY ,G SUBBA RAO | ACCESS PUBLISHING INDIA PVT.LTD | 10 |
| **36** | 9.789380e+12 | GENERAL STUDIES PAPER -- I (2016) | Mark | ACCESS PUBLISHING INDIA PVT.LTD | 0 |
| **37** | 9.789380e+12 | GENERAL STUDIES PAPER - II (2016) | Mark | ACCESS PUBLISHING INDIA PVT.LTD | 0 |
| **38** | 9.789380e+12 | INDIAN AND WORLD GEOGRAPHY 2E | D R KHULLAR | ACCESS PUBLISHING INDIA PVT.LTD | 10 |
| **39** | 9.789380e+12 | VASTUNISTHA PRASHN SANGRAHA: BHARAT KA ITIHAS | MEEMarkKSHI KANT | ACCESS PUBLISHING INDIA PVT.LTD | 0 |
| **40** | 9.789380e+12 | PHYSICAL, HUMAN AND ECONOMIC GEOGRAPHY | D R KHULLAR | ACCESS PUBLISHING INDIA PVT.LTD | 4 |
| **41** | 9.789380e+12 | WORLD GEOGRAPHY | DR KHULLAR | ACCESS PUBLISHING INDIA PVT.LTD | 5 |

| | Serial Number | Company Name | Employee Markme | Description | Leave |
|---|---|---|---|---|---|
| **42** | 9.789380e+12 | INDIA: MAP ENTRIES IN GEOGRAPHY | MAJID HUSAIN | ACCESS PUBLISHING INDIA PVT.LTD | 5 |
| **43** | 9.789380e+12 | GOOD GOVERMarkNCE IN INDIA 2/ED. | G SUBBA RAO | ACCESS PUBLISHING INDIA PVT.LTD | 1 |
| **44** | 9.789380e+12 | KAMYABI KE SUTRA-CIVIL SEWA PARIKSHA AAP KI MU... | ASHOK KUMAR | ACCESS PUBLISHING INDIA PVT.LTD | 0 |
| **45** | 9.789380e+12 | GENERAL SCIENCE PRELIRY EXAM | Mark | ACCESS PUBLISHING INDIA PVT.LTD | 0 |
| **46** | 9.781740e+12 | SUCCESS AND DYSLEXIA | SUCCESS AND DYSLEXIA | ACER PRESS | 0 |
| **47** | 9.781740e+12 | AN EXTRAORDIMarkRY SCHOOL | SARA JAMES | ACER PRESS | 0 |
| **48** | 9.781740e+12 | POWERFUL PRACTICES FOR READING IMPROVEMENT | GLASSWELL | ACER PRESS | 0 |
| **49** | 9.781740e+12 | EARLY CHILDHOOD PLAY MATTERS | SHOMark BASS | ACER PRESS | 0 |
| **50** | 9.781740e+12 | LEADING LEARNING AND TEACHING | STEPHEN DINHAM | ACER PRESS | 0 |
| **51** | 9.781740e+12 | READING AND LEARNING DIFFICULTIES | PETER WESTWOOD | ACER PRESS | 0 |
| **52** | 9.781740e+12 | NUMERACY AND LEARNING DIFFICULTIES | PETER WOODLAND] | ACER PRESS | 0 |
| **53** | 9.781740e+12 | TEACHING AND LEARNING DIFFICULTIES | PETER WOODLAND | ACER PRESS | 0 |
| **54** | 9.781740e+12 | USING DATA TO IMPROVE LEARNING | ANTHONY SHADDOCK | ACER PRESS | 0 |
| **55** | 9.781740e+12 | PATHWAYS TO SCHOOL SYSTEM IMPROVEMENT | MICHAEL GAFFNEY | ACER PRESS | 0 |
| **56** | 9.781740e+12 | FOR THOSE WHO TEACH | PHIL RIDDEN | ACER PRESS | 0 |
| **57** | 9.781740e+12 | KEYS TO SCHOOL LEADERSHIP | PHIL RIDDEN & JOHN DE NOBILE | ACER PRESS | 0 |
| **58** | 9.781740e+12 | DIVERSE LITERACIES IN EARLY CHILDHOOD | LEONIE ARTHUR | ACER PRESS | 0 |
| **59** | 9.781740e+12 | CREATIVE ARTS IN THE LIVESOF YOUNG CHILDREN | ROBYN EWING | ACER PRESS | 0 |
| **60** | 9.781740e+12 | SOCIAL AND EMOTIOMarkL DEVELOPMENT | ROS LEYDEN AND ERIN SHALE | ACER PRESS | 0 |
| **61** | 9.781740e+12 | DISCUSSIONS IN SCIENCE | TIM SPROD | ACER PRESS | 0 |
| **62** | 9.781740e+12 | YOUNG CHILDREN LEARNING MATHEMATICS | ROBERT HUNTING | ACER PRESS | 0 |
| **63** | 9.781740e+12 | COACHING CHILDREN | KELLY SUMICH | ACER PRESS | 1 |

| | Serial Number | Company Name | Employee Markme | Description | Leave |
|---|---|---|---|---|---|
| **64** | 9.781740e+12 | TEACHING PHYSICAL EDUCATIOMarkL IN PRIMARY SCHOOL | JANET L CURRIE | ACER PRESS | 0 |
| **65** | 9.781740e+12 | ASSESSMENT AND REPORTING | PHIL RIDDEN AND SANDY | ACER PRESS | 0 |
| **66** | 9.781740e+12 | COLLABORATION IN LEARNING | MAL LEE AND LORRAE WARD | ACER PRESS | 0 |
| **67** | 9.780860e+12 | RE-IMAGINING EDUCATIMarkL LEADERSHIP | BRIAN J.CALDWELL | ACER PRESS | 0 |
| **68** | 9.780860e+12 | TOWARDS A MOVING SCHOOL | FLEMING & KLEINHENZ | ACER PRESS | 0 |
| **69** | 9.780860e+12 | DESINGNING A THINKING A CURRICULAM | SUSAN WILKS | ACER PRESS | 0 |
| **70** | 9.780860e+12 | LEADING A DIGITAL SCHOOL | MAL LEE AND MICHEAL GAFFNEY | ACER PRESS | 0 |
| **71** | 9.780860e+12 | NUMERACY | WESTWOOD | ACER PRESS | 0 |
| **72** | 9.780860e+12 | TEACHING ORAL LANGUAGE | JOHN MUNRO | ACER PRESS | 0 |
| **73** | 9.780860e+12 | SPELLING | WESTWOOD | ACER PRESS | 0 |
| **74** | 9.788190e+12 | STORIES OF SHIVA | Mark | ACK | 0 |
| **75** | 9.788190e+12 | JAMSET JI TATA: THE MAN WHO SAW TOMORROW | NaN | ACK | 0 |
| **76** | 9.788180e+12 | HEROES FROM THE MAHABHARTA { 5-IN-1 } | Mark | ACK | 0 |
| **77** | 9.788180e+12 | SURYA | NaN | ACK | 0 |
| **78** | 9.788180e+12 | TALES OF THE MOTHER GODDESS | - | ACK | 0 |
| **79** | 9.788180e+12 | ADVENTURES OF KRISHMark | Mark | ACK | 0 |
| **80** | 9.788180e+12 | MAHATMA GANDHI | Mark | ACK | 1 |
| **81** | 9.788180e+12 | TALES FROM THE PANCHATANTRA 3-IN-1 | - | ACK | 0 |
| **82** | 9.788180e+12 | YET MORE TALES FROM THE JATAKAS { 3-IN-1 } | AMarkNT PAI | ACK | 0 |
| **83** | 9.788180e+12 | LEGENDARY RULERS OF INDIA | - | ACK | 0 |
| **84** | 9.788180e+12 | GREAT INDIAN CLASSIC | Mark | ACK | 0 |
| **85** | 9.788180e+12 | TULSIDAS ' RAMAYAMark | Mark | ACK | 0 |
| **86** | 9.788180e+12 | TALES OF HANUMAN | - | ACK | 0 |
| **87** | 9.788180e+12 | VALMIKI'S RAMAYAMark | A C K | ACK | 1 |
| **88** | 9.788180e+12 | THE BEST OF INIDAN WIT AND WISDOM | Mark | ACK | 0 |

| | Serial Number | Company Name | Employee Markme | Description | Leave |
|---|---|---|---|---|---|
| **89** | 9.788180e+12 | MORE TALES FROM THE PANCHTANTRA | AMarkNT PAL | ACK | 0 |
| **90** | 9.788180e+12 | THE GREAT MUGHALS {5-IN-1} | AMarkNT. | ACK | 0 |
| **91** | 9.788180e+12 | FAMOUS SCIENTISTS | Mark | ACK | 0 |
| **92** | 9.788180e+12 | KOMarkRK | Mark | ACK | 0 |
| **93** | 9.788180e+12 | THE MUGHAL COURT | REEMark | ACK | 0 |
| **94** | 9.788180e+12 | MORE STORIES FROM THE JATAKAS | Mark | ACK | 0 |
| **95** | 9.788180e+12 | MORE TALES OF BIRBAL | - | ACK | 0 |
| **96** | 9.788180e+12 | TALES FROM THE JATAKAS | - | ACK | 0 |
| **97** | 9.788180e+12 | RAMarkS OF MEWAR | - | ACK | 0 |
| **98** | 9.788180e+12 | THE SONS OF THE PANDAVAS | - | ACK | 0 |

In [ ]:

In [ ]:

In [46]:
```
a.sort_index()
```

Out[46]:

| | | | | college data |
|---|---|---|---|---|
| **0** | **name** | **college** | **dob** | marks |
| **1** | **nikhil** | **menon** | **30/9/1999** | 86 |
| **2** | **ketan** | **menon** | **10/2/2000** | 86 |
| **3** | **prashant** | **menon** | **4/7/2003** | 67 |
| **4** | **jj** | **5/3/2011** | **NaN** | NaN |
| **5** | **kk** | **7/8/2001** | **NaN** | NaN |

In [ ]:
```
# skiprows=select the starting line
# nrows= lines to be read after it   # how much raw want then use
# header --> 0 1 2 3

a=pd.read_csv('data.csv',skiprows=[2],nrows=4,header=None)
a
```

In [39]:
```
# writing data in CSV


a.to_csv('data.csv')
```

In [32]:
```
a.to_csv('data.csv',index=False,header=False)
a
```

Out[32]:

| | college data |
|---|---|

| | | | | college data |
|---|---|---|---|---|
| **0** | **name** | **college** | **dob** | marks |
| **1** | **nikhil** | **menon** | **30/9/1999** | 86 |
| **2** | **ketan** | **menon** | **10/2/2000** | 86 |
| **3** | **prashant** | **menon** | **4/7/2003** | 67 |
| **4** | **jj** | **5/3/2011** | **NaN** | NaN |
| **5** | **kk** | **7/8/2001** | **NaN** | NaN |

In [31]:
```
a.to_csv('data.csv',na_rep='NaN')
a
```

Out[31]:

| | | | | college data |
|---|---|---|---|---|
| **0** | **name** | **college** | **dob** | marks |
| **1** | **nikhil** | **menon** | **30/9/1999** | 86 |
| **2** | **ketan** | **menon** | **10/2/2000** | 86 |
| **3** | **prashant** | **menon** | **4/7/2003** | 67 |
| **4** | **jj** | **5/3/2011** | **NaN** | NaN |
| **5** | **kk** | **7/8/2001** | **NaN** | NaN |

In [ ]:
```

```

In [8]:
```
# ----------------------------reading and writing Data in HTML

# read_html()
# to_html()----> directly convert the DataFrame in an HTML file

# html file automatic create in working directory

import numpy as np
import pandas as pd
h=pd.DataFrame(np.arange(4).reshape(2,2))
h
```

Out[8]:

| | **0** | **1** |
|---|---|---|
| **0** | 0 | 1 |
| **1** | 2 | 3 |

In [9]:
```
print(h.to_html())   # pd to html(h.to_html)
```

```
<table border="1" class="dataframe">
  <thead>
    <tr style="text-align: right;">
      <th></th>
      <th>0</th>
      <th>1</th>
    </tr>
```

```
  </thead>
  <tbody>
    <tr>
      <th>0</th>
      <td>0</td>
      <td>1</td>
    </tr>
    <tr>
      <th>1</th>
      <td>2</td>
      <td>3</td>
    </tr>
  </tbody>
</table>
```

In [10]:
```
# simple dataframe

h=pd.DataFrame(np.random.random((4,4)),
               index=['white','black','red','blue'],
               columns=['up','down','right','left'])
h
```

Out[10]:

| | up | down | right | left |
|---|---|---|---|---|
| **white** | 0.937956 | 0.430853 | 0.529881 | 0.347045 |
| **black** | 0.967814 | 0.054729 | 0.946517 | 0.922334 |
| **red** | 0.006358 | 0.263461 | 0.998704 | 0.774403 |
| **blue** | 0.462672 | 0.620645 | 0.171823 | 0.065939 |

In [11]:
```
# html tag
# create new object s1

s1=['<html>']
s1.append('<head><title>my HTML file</title></head>')
s1.append('<body>')
s1.append(h.to_html())
s1.append('</body></html>')
html=''.join(s1)    # join to s1 dataframe
```

In [12]:
```
html_file=open('frame.html','w')    #create new file frame.html automatic
html_file.write(html)
html_file.close()
```

In [13]:
```
# -----------------reading DataFrame fom an HTML page

import pandas as pd
a=pd.read_html('frame.html')
a
```

Out[13]:
```
[  Unnamed: 0        up      down     right      left
 0      white  0.937956  0.430853  0.529881  0.347045
 1      black  0.967814  0.054729  0.946517  0.922334
 2        red  0.006358  0.263461  0.998704  0.774403
 3       blue  0.462672  0.620645  0.171823  0.065939]
```

In [2]:
```

```

In [2]:
```
# ------------reading and writing data on microsoft Excel files

# to_excel()
# read_excel()
#install module--->pip install xlrd
# xlsx support 2007 excel
#xls support 2003 excel

import pandas as pd
pd.read_excel('excel1.xlsx')
```

Out[2]:

| | Unnamed: 0 | white | red | green | black |
|---|---|---|---|---|---|
| **0** | a | 12 | 23 | 17 | 18 |
| **1** | b | 22 | 16 | 19 | 18 |
| **2** | c | 14 | 23 | 22 | 21 |

In [18]:
```
pd.read_excel('excel1.xlsx','Sheet2')   # S should be capital
```

Out[18]:

| | Unnamed: 0 | YELLOW | PURPLE | BLUE | ORANGE |
|---|---|---|---|---|---|
| **0** | A | 11 | 16 | 44 | 22 |
| **1** | B | 20 | 22 | 23 | 44 |
| **2** | C | 30 | 31 | 37 | 32 |

In [3]:
```
# dataframe to excel

import pandas as pd
import numpy as np

e=pd.DataFrame(np.random.random((4,4)),
               index=['exp1','exp2','exp3','exp4'],
               columns=['jan2015','feb2015','mar2015','april2005'])
e
```

Out[3]:

| | jan2015 | feb2015 | mar2015 | april2005 |
|---|---|---|---|---|
| **exp1** | 0.077299 | 0.434007 | 0.892895 | 0.599720 |
| **exp2** | 0.973440 | 0.404022 | 0.634587 | 0.977898 |
| **exp3** | 0.609296 | 0.529174 | 0.586733 | 0.338496 |
| **exp4** | 0.033061 | 0.274083 | 0.340606 | 0.250281 |

In [12]:
```
# save excel file in working directory
# save dataframe to excel(e-->excel)


e.to_excel('excel3.xlsx')
```

In [13]:
```
pd.read_excel('excel3.xlsx')  # reading/show excel data
```

Out[13]:

| | Unnamed: 0 | jan2015 | feb2015 | mar2015 | april2005 |
|---|---|---|---|---|---|

| | Unnamed: 0 | jan2015 | feb2015 | mar2015 | april2005 |
|---|---|---|---|---|---|
| **0** | exp1 | 0.077299 | 0.434007 | 0.892895 | 0.599720 |
| **1** | exp2 | 0.973440 | 0.404022 | 0.634587 | 0.977898 |
| **2** | exp3 | 0.609296 | 0.529174 | 0.586733 | 0.338496 |
| **3** | exp4 | 0.033061 | 0.274083 | 0.340606 | 0.250281 |

In [8]:
```
# -------------------------------- JSON data
# javaScript object notation
# use for transmission of data through the web
# tabular format
# read_json  --  read
# to_json   ---save

#-------------- create simple dataframe
# example 1

j=pd.DataFrame(np.arange(16).reshape(4,4),
               index=['white','black','red','blue'],
               columns=['up','down','right','left'])
j
```

Out[8]:

| | up | down | right | left |
|---|---|---|---|---|
| **white** | 0 | 1 | 2 | 3 |
| **black** | 4 | 5 | 6 | 7 |
| **red** | 8 | 9 | 10 | 11 |
| **blue** | 12 | 13 | 14 | 15 |

In [9]:
```
# save DataFrame to json


j.to_json('jsondata.json')
```

In [12]:
```
# reading json file

pd.read_json('jsondata.json')
```

Out[12]:

| | up | down | right | left |
|---|---|---|---|---|
| **white** | 0 | 1 | 2 | 3 |
| **black** | 4 | 5 | 6 | 7 |
| **red** | 8 | 9 | 10 | 11 |
| **blue** | 12 | 13 | 14 | 15 |

In [55]:
```
# example 2
# create json data

data1={"sub_ID":["1","2","3","4"],
       "name":["nikhil","rahul","prashant","kk"],
```

```
      "clg name":["menon","KJ","jondhale","HM"],
      "fav subject":["python","share market","maths","pS"],
      "city":["thane","mumbai","mulund","thane"]}
data1
```

Out[55]:
```
{'sub_ID': ['1', '2', '3', '4'],
 'name': ['nikhil', 'rahul', 'prashant', 'kk'],
 'clg name': ['menon', 'KJ', 'jondhale', 'HM'],
 'fav subject': ['python', 'share market', 'maths', 'pS'],
 'city': ['thane', 'mumbai', 'mulund', 'thane']}
```

In [ ]:

In [48]:
```
 # save json file


import json
import pandas as pd
with open('data1.json','w') as outfile:
    json.dump(data1,outfile)
```

In [49]:
```
# reading json file

df=pd.read_json('data1.json')
df
```

Out[49]:

| | sub_ID | name | clg name | fav subject | city |
|---|---|---|---|---|---|
| **0** | 1 | nikhil | menon | python | thane |
| **1** | 2 | rahul | KJ | share market | mumbai |
| **2** | 3 | prashant | jondhale | maths | mulund |
| **3** | 4 | kk | HM | pS | thane |

In [ ]:

In [51]:
```
#---------------- read URL data using json

url="https://api.exchangerate-api.com/v4/latest/USD"
u=pd.read_json(url)
u.head()
```

Out[51]:

| | base | date | time_last_updated | rates |
|---|---|---|---|---|
| **AED** | USD | 2020-10-21 | 1603238649 | 3.672008 |
| **ARS** | USD | 2020-10-21 | 1603238649 | 77.585630 |
| **AUD** | USD | 2020-10-21 | 1603238649 | 1.419170 |
| **BGN** | USD | 2020-10-21 | 1603238649 | 1.657360 |
| **BRL** | USD | 2020-10-21 | 1603238649 | 5.594937 |

In [52]:
```
url
```

Out[52]:
```
'https://api.exchangerate-api.com/v4/latest/USD'
```

In [59]:
```
# read all containtent data

ur=pd.read_json("https://api.exchangerate-api.com/v4/latest/USD")
ur
```

Out[59]:

| | base | date | time_last_updated | rates |
|---|---|---|---|---|
| **AED** | USD | 2020-10-21 | 1603238649 | 3.672008 |
| **ARS** | USD | 2020-10-21 | 1603238649 | 77.585630 |
| **AUD** | USD | 2020-10-21 | 1603238649 | 1.419170 |
| **BGN** | USD | 2020-10-21 | 1603238649 | 1.657360 |
| **BRL** | USD | 2020-10-21 | 1603238649 | 5.594937 |
| **BSD** | USD | 2020-10-21 | 1603238649 | 1.000000 |
| **CAD** | USD | 2020-10-21 | 1603238649 | 1.316241 |
| **CHF** | USD | 2020-10-21 | 1603238649 | 0.908074 |
| **CLP** | USD | 2020-10-21 | 1603238649 | 787.856810 |
| **CNY** | USD | 2020-10-21 | 1603238649 | 6.679677 |
| **COP** | USD | 2020-10-21 | 1603238649 | 3753.666667 |
| **CZK** | USD | 2020-10-21 | 1603238649 | 23.098172 |
| **DKK** | USD | 2020-10-21 | 1603238649 | 6.308761 |
| **DOP** | USD | 2020-10-21 | 1603238649 | 58.089167 |
| **EGP** | USD | 2020-10-21 | 1603238649 | 15.662031 |
| **EUR** | USD | 2020-10-21 | 1603238649 | 0.846949 |
| **FJD** | USD | 2020-10-21 | 1603238649 | 2.140875 |
| **GBP** | USD | 2020-10-21 | 1603238649 | 0.772320 |
| **GTQ** | USD | 2020-10-21 | 1603238649 | 7.734962 |
| **HKD** | USD | 2020-10-21 | 1603238649 | 7.750137 |
| **HRK** | USD | 2020-10-21 | 1603238649 | 6.426060 |
| **HUF** | USD | 2020-10-21 | 1603238649 | 309.639248 |
| **IDR** | USD | 2020-10-21 | 1603238649 | 14864.469422 |
| **ILS** | USD | 2020-10-21 | 1603238649 | 3.381185 |
| **INR** | USD | 2020-10-21 | 1603238649 | 73.468920 |
| **ISK** | USD | 2020-10-21 | 1603238649 | 138.841645 |
| **JPY** | USD | 2020-10-21 | 1603238649 | 105.556721 |
| **KRW** | USD | 2020-10-21 | 1603238649 | 1139.889142 |
| **KZT** | USD | 2020-10-21 | 1603238649 | 426.091892 |
| **MVR** | USD | 2020-10-21 | 1603238649 | 15.400000 |

| | base | date | time_last_updated | rates |
|---|---|---|---|---|
| **MXN** | USD | 2020-10-21 | 1603238649 | 21.165014 |
| **MYR** | USD | 2020-10-21 | 1603238649 | 4.147970 |
| **NOK** | USD | 2020-10-21 | 1603238649 | 9.290145 |
| **NZD** | USD | 2020-10-21 | 1603238649 | 1.520471 |
| **PAB** | USD | 2020-10-21 | 1603238649 | 1.000000 |
| **PEN** | USD | 2020-10-21 | 1603238649 | 3.589138 |
| **PHP** | USD | 2020-10-21 | 1603238649 | 48.555344 |
| **PKR** | USD | 2020-10-21 | 1603238649 | 161.862423 |
| **PLN** | USD | 2020-10-21 | 1603238649 | 3.883343 |
| **PYG** | USD | 2020-10-21 | 1603238649 | 7166.090909 |
| **RON** | USD | 2020-10-21 | 1603238649 | 4.133058 |
| **RUB** | USD | 2020-10-21 | 1603238649 | 77.762882 |
| **SAR** | USD | 2020-10-21 | 1603238649 | 3.750227 |
| **SEK** | USD | 2020-10-21 | 1603238649 | 8.783408 |
| **SGD** | USD | 2020-10-21 | 1603238649 | 1.357321 |
| **THB** | USD | 2020-10-21 | 1603238649 | 31.267255 |
| **TRY** | USD | 2020-10-21 | 1603238649 | 7.880041 |
| **TWD** | USD | 2020-10-21 | 1603238649 | 28.713424 |
| **UAH** | USD | 2020-10-21 | 1603238649 | 28.340281 |
| **USD** | USD | 2020-10-21 | 1603238649 | 1.000000 |
| **UYU** | USD | 2020-10-21 | 1603238649 | 42.724661 |
| **ZAR** | USD | 2020-10-21 | 1603238649 | 16.499809 |

In [1]:
```
#  ------------------The  format HDF5

#--hierarchical Data format
# -- save huge amount of data
# --HDF5 supports compression in real time,thereby taking advantage of repetitive patte
# the data sturcture to compress the file size
# python choice-->
#  1)pyTables-->HDF5 to provide flexible data containers,indexed tables,querying capabi

#  2) h5py-->direct interface with hight-level APIs HDF5

# HDFStore--> pandas class, pytables to store pandas objects
```

In [3]:
```
from pandas.io.pytables import HDFStore
import pandas as pd
import numpy as np
h=pd.DataFrame(np.arange(16).reshape(4,4),
               index=['white','black','red','blue'],
```

```
                columns=['up','down','right','left'])
h
```

Out[3]:

| | up | down | right | left |
|---|---|---|---|---|
| **white** | 0 | 1 | 2 | 3 |
| **black** | 4 | 5 | 6 | 7 |
| **red** | 8 | 9 | 10 | 11 |
| **blue** | 12 | 13 | 14 | 15 |

In [6]:
```
# save HDFStore file in working directory
# we can store  multiple data structure within the same HDF5 file,specifying for each o

store=HDFStore('mydata.h5')
store['obj1']=h           # 1st data structure
```

In [8]:
```
h1=pd.DataFrame(np.arange(16).reshape(4,4),
                index=['white','black','red','blue'],
                columns=['up','down','right','left'])
h1
```

Out[8]:

| | up | down | right | left |
|---|---|---|---|---|
| **white** | 0 | 1 | 2 | 3 |
| **black** | 4 | 5 | 6 | 7 |
| **red** | 8 | 9 | 10 | 11 |
| **blue** | 12 | 13 | 14 | 15 |

In [9]:
```
store['obj2']=h1      #--2nd data structure
```

In [19]:
```
store
```

Out[19]:
```
<class 'pandas.io.pytables.HDFStore'>
File path: mydata.h5
```

In [10]:
```
# show store data
store['obj2']
```

Out[10]:

| | up | down | right | left |
|---|---|---|---|---|
| **white** | 0 | 1 | 2 | 3 |
| **black** | 4 | 5 | 6 | 7 |
| **red** | 8 | 9 | 10 | 11 |
| **blue** | 12 | 13 | 14 | 15 |