DictReader

csv.DictReader(*f*, *fieldnames=None)*

Create an object that operates like a regular reader but maps the information in each row to a dict whose keys are given by the optional *fieldnames* parameter.

The *fieldnames* parameter is a sequence. If *fieldnames* is omitted, the values in the first row of file *f* will be used as the fieldnames.

```
import csv
def main():
    f=open("employees.csv","r")
    dictr=csv.DictReader(f)
    total=0
    for row in dictr:
        print(row['empno'],row['ename'],row['salary'])
        total=total+float(row['salary'])
    print("Total Salary :",total)

main()
```

Output:

```
1 aaa 50000
2 bbb 45000
3 ccc 45001
4 ddd 45002
5 eee 65000
6 fff 55000
7 ddd 45000
8 sss 25000
9 ddd 15000
10 ddd 12000
11 ccc 11000
12 xyz 10000
13 sss 19000
14 ggg 22000
15 eee 45000
16 yy 65000
17 ttt 90000
18 hhh 34000
19 iii 24000
```

```
20 kkk 23000
21 ppp 43000
22 ooo 32000
23 ttt 34000
24 rrrr 33000
Total Salary : 887003.0
```

Creating csv file and writing data

csv.writer(*csvfile)*
Return a writer object responsible for converting the user's data into delimited strings on the given file-like object. *csvfile* can be any object with a write() method. If *csvfile* is a file object, it should be opened with newline=''

```
# creating csv file
import csv
def main():
    with open("student.csv","w",newline='') as f:
        w=csv.writer(f)
        w.writerow(['rollno','name','course'])
        while True:
            rno=input("enter rollno")
            name=input("enter name")
            course=input("enter course")
            row=[rno,name,course]
            w.writerow(row)
            ans=input("add another student?")
            if ans=='no':
                break

main()
```

csv.DictWriter(*f, fieldnames)*
Create an object which operates like a regular writer but maps dictionaries onto output rows. The *fieldnames* parameter is a sequence of keys that identify the order in which values in the dictionary passed to the writerow() method are written to file *f*

```
import csv
def main():
    with open("product.csv","w",newline='') as f:
        dw=csv.DictWriter(f,fieldnames=['pname','qty'])
        dw.writeheader()
        while True:
            pn=input("ProductName :")
            q=input("Qty:")
            row={'pname':pn,'qty':q}
            dw.writerow(row)
            ans=input("Add another product?")
            if ans=='no':
                break

main()
```

Output:

```
ProductName :mouse
Qty:5
Add another product?yes
ProductName :keyboard
Qty:10
Add another product?yes
ProductName :monitor
Qty:10
Add another product?no
>>>
```

JSON

JSON (JavaScript Object Notation) is a lightweight data-interchange format. It is easy for humans to read and write. It is easy for machines to parse and generate. It is based on a subset of the JavaScript Programming Language Standard ECMA-262 3rd Edition - December 1999. JSON is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl, Python, and many others. These properties make JSON an ideal data-interchange language.

Python provides json module.

This json module provide json encoder and decoder

JSON	**Python**
object	dict
array	list
string	str
number (int)	int
number (real)	float
true	True
false	False
null	None

json.dump(*obj*, *fp) : convert python object into json string and write inside file*

json.dumps(*obj) : convert python object into json string*