

```
In [4]: #Function to add the data to contacts
from Packages import Validators
def addcontact(name,phone,email):
    filename="DataFiles/contacts.txt"
    with open(filename,'a') as f:
        line=name + ',' + phone + ',' + email + '\n'
        f.write(line)
    print(name,"added to contcts")
    return
addcontact("name1",'9876543210',"name1@gmail.com")
```

name1 added to contcts

```
In [11]: #Function to check if contact already exists
import re
def checkcontactexists(name):
    filename="DataFiles/contacts.txt"
    with open(filename,'r') as f:
        filedata=f.read()
    return re.search(name,filedata)
if checkcontactexists("name1"):
    print("True")
else:
    print("False")
checkcontactexists("name1")
```

True

```
Out[11]: <re.Match object; span=(0, 5), match='name1'>
```

```
In [12]: #Function to add the data to contacts
from Packages import Validators
def addcontact(name,phone,email):
    filename="DataFiles/contacts.txt"
    if not checkcontactexists(name):
        with open(filename,'a') as f:
            line=name + ',' + str(phone) + ',' + email + '\n'
            f.write(line)
            print(name,"added to contcts")
    else:
        print(name,"already exists")
    return
addcontact("name1", '9876543210', "name1@gmail.com")

import re
def checkcontactexists(name):
    filename="DataFiles/contacts.txt"
    with open(filename,'r') as f:
        filedata=f.read()
    return re.search(name,filedata)
if checkcontactexists("name1"):
    print("True")
else:
    print("False")
checkcontactexists("name1")
```

```
name1 already exists
True
```

```
Out[12]: <re.Match object; span=(0, 5), match='name1'>
```

```

In [8]: #Function to add the data to contacts
from Packages.Validators import phoneNumberValidator as pnv
from Packages.Validators import emailValidator as email
import re
def checkcontactexists(name):
    filename="DataFiles/contacts.txt"
    with open(filename,'r') as f:
        filedata=f.read()
    return re.search(name,filedata)
addcontact("name2","12344","ds")
def addcontact(name,phone,email):
    filename="DataFiles/contacts.txt"
    if not checkcontactexists(name):
        if pnv(phone) and email(email):
            with open(filename,'a') as f:
                line=name + ',' + (phone) + ',' + email + '\n'
                f.write(line)
            print(name,"added to contcts")
        else:
            print("invalid phone number or email")
    else:
        print(name,"already exists")
    return
addcontact("name1",'9872343210',"name1@gmail.com")

```

invalid phone number or email
name1 already exists

```

In [16]: filename="DataFiles/contacts.txt"
def csvToList(filename):
    li=[]
    with open(filename,'r') as f:

        for line in f:
            li.append(line.split(','))
        return li
li=csvToList(filename)
def listToFile(li):
    s=""
    for i in li:
        s+=",".join(i)
    return s
#csvToList(filename)
listToFile(li)

```

Out[16]: 'name1,9876543210,name1@gmail.com\n'

```

In [1]: fname="DataFiles/mycontacts.txt"
f=open(fname,"w")
f.write("Alekhya,8328363233,alekhyaganji440@gmail.com")
f.close()

```

```
In [5]: with open(fname, 'r') as f:
        print(f.read())
```

Alekhyia,8328363233,alekhyaganji440@gmail.com

```
In [7]: with open(fname, 'a') as f:
        f.write("\nArchana,628371793918,alhuh@gmail.com")
```

```
In [16]: with open(fname, 'r') as f:
         print(f.readlines())
```

```
['import re\n', '\n', 'def phoneNumberValidator(number):\n', '    pattern = "^
[6-9][0-9]{9}$|^[0][6-9][0-9]{9}$|^[+][9][1][6-9][0-9]{9}$"\n', '    if re.mat
ch(pattern,str(number)):\n', '        return True\n', '    else:\n', '        re
turn False\n', '    return\n', '\n', 'def emailValidator(email_id):\n', '    pa
ttern = "^[0-9a-z][0-9a-z_.]{4,13}[0-9a-z][@][0-9a-z]{3,18}[.][a-z]{2,4}
$" \n', '    if re.match(pattern,email_id):\n', '        return True\n', '    ret
urn False \n']
```

```
In [17]: with open(fname, 'r') as f:
         for line in f.readlines():
             print(line,end="")
```

```
import re
```

```
def phoneNumberValidator(number):
    pattern = "^[6-9][0-9]{9}$|^[0][6-9][0-9]{9}$|^[+][9][1][6-9][0-9]{9}$"
    if re.match(pattern,str(number)):
        return True
    else:
        return False
    return
```

```
def emailValidator(email_id):
    pattern = "^[0-9a-z][0-9a-z_.]{4,13}[0-9a-z][@][0-9a-z]{3,18}[.][a-z]{2,4}
$"
    if re.match(pattern,email_id):
        return True
    return False
```

```
In [14]: def listcontacts(fname):

    with open(fname, 'r') as f:
        for line in f.readlines():
            print(line, end="")
fname="Packages/Validators.py"
listcontacts(fname)

import re

def phoneNumberValidator(number):
    pattern = "^[6-9][0-9]{9}$|^[0][6-9][0-9]{9}$|^[+][9][1][6-9][0-9]{9}$"
    if re.match(pattern, str(number)):
        return True
    else:
        return False
    return

def emailValidator(email_id):
    pattern = "^[0-9a-z][0-9a-z_]{4,13}[0-9a-z][@][0-9a-z]{3,18}[.][a-z]{2,4}$"
    if re.match(pattern, email_id):
        return True
    return False
```

```
In [35]: def listtocsv(fname):

    fname="DataFiles/mycontacts.txt"
    with open(fname, 'r') as f:
        f1=[]
        for line in f:
            f1.append(line.split(sep=","))
    return f1
listtocsv(fname)
```

```
Out[35]: [['Alekhya', '8328363233', 'alekhyaganji440@gmail.com\n'],
           ['Archana', '628371793918', 'alhuh@gmail.com']]
```

```
In [36]: def search(fname, name):
    f1=listtocsv(fname)
    flag=0
    for line in f1:
        if line[0]==name:
            flag=1
            print(','.join(line))
    if flag==0:
        print("contact not found")

search(fname, "Alekhya")

Alekhya,8328363233,alekhyaganji440@gmail.com
```

```
In [57]: l=list(input())
         for i in l:
             print(i,end=",")
```

alekhya
a,l,e,k,h,y,a,

```
In [59]: l=list(input())
         for i in l:
             print(i,end=" ")
```

alekhya
a l e k h y a

```
In [20]: l=input().split('+')
         sum=0
         for i in l:
             sum=sum+int(i)
         print(sum)
         print(type(l))
```

1+2+3
6
<class 'list'>

```
In [39]: del sum
```

```
In [41]: s='123456T7'
         sum(map(int,list(filter(str.isdigit,s))))
         #List(map(int,s))
```

Out[41]: 28

```
In [5]: del sum
```

```
In [43]: s=['k','a','c',1,2,3]
         s
```

Out[43]: ['k', 'a', 'c', 1, 2, 3]

```
In [44]: str.split?
```

```
In [45]: int('1')
         int('10')
         ord('A')
```

Out[45]: 65

```
In [67]: s=list(map(int,input().split(",")))
         min(s)
```

13,4,5

Out[67]: 4

```
In [70]: s="abd"
         for i in s:
             print(type(i))
             break
```

<class 'str'>

```
In [73]: l=[1,2,3]
         l.insert(2,4)
         l
```

Out[73]: [1, 2, 4, 3]

```
In [74]: dir(l)
```

```
Out[74]: ['__add__',
          '__class__',
          '__contains__',
          '__delattr__',
          '__delitem__',
          '__dir__',
          '__doc__',
          '__eq__',
          '__format__',
          '__ge__',
          '__getattr__',
          '__getitem__',
          '__gt__',
          '__hash__',
          '__iadd__',
          '__imul__',
          '__init__',
          '__init_subclass__',
          '__iter__',
          '__le__',
          '__len__',
          '__lt__',
          '__mul__',
          '__ne__',
          '__new__',
          '__reduce__',
          '__reduce_ex__',
          '__repr__',
          '__reversed__',
          '__rmul__',
          '__setattr__',
          '__setitem__',
          '__sizeof__',
          '__str__',
          '__subclasshook__',
          'append',
          'clear',
          'copy',
          'count',
          'extend',
          'index',
          'insert',
          'pop',
          'remove',
          'reverse',
          'sort']
```

```
In [92]: d={5:9,8:7,1:2}
         for key in d.keys():
             if key==1:
                 d[key]=3
         print(d)
```

```
{5: 9, 8: 7, 1: 3}
```



```
In [87]: dir(dict)
```

```
Out[87]: ['__class__',
          '__contains__',
          '__delattr__',
          '__delitem__',
          '__dir__',
          '__doc__',
          '__eq__',
          '__format__',
          '__ge__',
          '__getattribute__',
          '__getitem__',
          '__gt__',
          '__hash__',
          '__init__',
          '__init_subclass__',
          '__iter__',
          '__le__',
          '__len__',
          '__lt__',
          '__ne__',
          '__new__',
          '__reduce__',
          '__reduce_ex__',
          '__repr__',
          '__setattr__',
          '__setitem__',
          '__sizeof__',
          '__str__',
          '__subclasshook__',
          'clear',
          'copy',
          'fromkeys',
          'get',
          'items',
          'keys',
          'pop',
          'popitem',
          'setdefault',
          'update',
          'values']
```

```
In [98]: d={1:2,3:4}
```

```
    #d[key]=7
d[4]=13
print(d)
```

```
{1: 2, 3: 4, 4: 13}
```

```
In [ ]:
```