Regular Expression

- Pattern Matching
- Symbolic Notation of a pattern
  - Pattern : Format which repeat
  - Pattern(RE): Represent the set of all values
- [0-9] ->any digit
- [a-z] -> any lower case alphabet
- [2468] -> All single digit multiple of 2
- [8642]
- [6824]
- ^[0-9]{1}$ ->Only one digit number
- ^[0-9]{2}$ ->Only two digit number
- [0-9]*[0]$ ->All Multiple of 10
- ^[0-9][0-9]*[0]$ ->All Multiple of 5
  - ^[1-9][0-9]*[05]|([5])$ ->All Multiple of 5 and 10
  - ^[1-9][0-9]{9}$ - > 10 digit number

    -[w][o][r][d] or (word) -> searching for a word

```
- ^ [6-9][0-9]{9}$ | ^[0][6-9][0-9]{9}$| ^[+][9][1][6-9][0-9]{9}$ ->vali
d phone number

- Email Validation(username@domain.extension)
-Username
    -Length of Username :[6,15]
    -No special characters other than _.
    -should not begin and end with _.
    -character set: all digits and lowercase
-Domain
    -Length of Domain :  [3,18]
    -no special characters
    -character set: all digits and alphabet
-Extension
    -length of Extension :[2,4]
    -no special characters
    -character set: alphabet

  -^[0-9a-z][0-9a-z_.]{4,13}[0-9a-z][@][0-9a-z]{3,18}[.][a-z]{2,4}$
```

- ^[a]...[z]$Anystringoflength5thatstartwith\ "\ a\ "\ andendswith\ "\ z\ "\ -[a].*[z]$ any string starts with a and ends with z

In [3]:
```
#Function to validate a phone number
import re
def phonenumber(number):
    pattern='^[6-9][0-9]{9}$|^[0][6-9][0-9]{9}$ | ^[+][9][1][6-9][0-9]{9}$'
    if re.match(pattern,number):
        print("valid number")
    else:
        print("invalid number")
    return
number=input()
phonenumber(number)



def verifiedmail(email):
    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):
        return True
    return False
#verifiedmail("_12345@gmail.com")

```

```
9492363502
valid number
```

In [1]:
```
import re
def verifiedmail(email):
    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):
        print("valid email")
    else:
        print("invalid email")
    return
email=input()
verifiedmail(email)

```

```
_12345@gmail.com
invalid email
```

In [11]:
```
contacts={"name1":[9492363502,'name1@domain.ext'],"name2":[8790700295,"name2
def addcontact(name,phone,email):
    if name in contacts:
        print(name,"already exists")
    else:
        if not phonenumber(phone):
            print("invalid phone number")
        if not verifiedmail(email):
            print("invalid email address")
            return
        newcontact= []
        newcontact.append(phone)
        newcontact.append(email)
        contacts[name]=newcontact
    print(contacts)
addcontact("name3","9492363502","srikanya659@gmail.com")

```

```
valid number
invalid phone number
{'name1': [9492363502, 'name1@domain.ext'], 'name2': [8790700295, 'name2@domai
n.ext'], 'name3': ['9492363502', 'srikanya659@gmail.com']}
```

In [14]:
```
def searchcontact(name):
    if name in contacts:
        print(name)
        print("phone :",contacts[name][0])
        print("email :",contacts[name][1])
    else:
        print("%s does not exist" %name)
    return
searchcontact("name1")
```

```
name1
phone : 9492363502
email : name1@domain.ext
```

In [15]:
```
def importcontact(newcontacts):
    contacts.update(newcontacts)
    print(len(newcontacts.keys())), "added successfully"
    return
newcontacts={"name4":[9876543234,"name4_123@gmail.com"]}
importcontact(newcontacts)
contacts
```

```
1
```

Out[15]:
```
{'name1': [9492363502, 'name1@domain.ext'],
 'name2': [8790700295, 'name2@domain.ext'],
 'name3': ['9492363502', 'srikanya659@gmail.com'],
 'name4': [9876543234, 'name4_123@gmail.com']}
```

In [18]:
```
def listofcontacts(n):
    for i in n.keys():
        print(i,":",contacts[i])
    return
listofcontacts(contacts)
```

```
name1 : [9492363502, 'name1@domain.ext']
name2 : [8790700295, 'name2@domain.ext']
name3 : ['9492363502', 'srikanya659@gmail.com']
name4 : [9876543234, 'name4_123@gmail.com']
```

In [19]:
```
def listallcontacts():
    for contact, info in contacts.items():
        print(contact,"\n","phone :",info[0],"\n","email :",info[1])
    return
listallcontacts()

```

```
name1
 phone : 9492363502
 email : name1@domain.ext
name2
 phone : 8790700295
 email : name2@domain.ext
name3
 phone : 9492363502
 email : srikanya659@gmail.com
name4
 phone : 9876543234
 email : name4_123@gmail.com
```

In [ ]:
```
## Function to edit contact information
def editcontact(name,phone,email):

```

File Handling in Python

- File: Document containing information residing perminent storage
- Types: Text,PDF,CSV etc
- File I/o-Channelling I/o data to files
- Default I/o channels-keyboard / screen
- Change I/o channel to files for reading and writing
- Read a file-Input
- Write to a file-output
- Read/write a file : open(filename,mode)
-

In [18]:
```
#Function to rad a file
def readfile(filename):
    f=open(filename,'r')
    filedata=f.read()
    f.close()
    return filedata
filename='DataFiles/data.txt'
filedata=readfile(filename)
readfile(filename)
#for line in filedata.split('\n'):
    #print(line)




def printfiledata(filename):
    f=open(filename,'r')
    for line in f:
        print(line,end="")
    return
printfiledata(filename)
print(readfile(filename))
```

```
Line1
Line2
Line3Line1
Line2
Line3
```

In [24]:
```
#Function to write into a file
def writeinfile(filename,filedata):
    with open(filename,'w') as f:
        f.write(filedata)
    return
filename='DataFiles/data.txt'
writeinfile(filename,"new data\n")

```

In [34]:
```
# Function to append data to a file
def appenddata(filename,filedata):
    with open(filename,'a') as f:
        for line in filedata:

            f.write(line+'\n')
    return
filedata=["Line4","Line 5"]
appenddata(filename,filedata)
```

Goki and his breakup

In [3]:
```
## Goki and his breakup

def Breakup(n):
    if(n==1 or n>=1):
        print("YES")
    else:
        print("NO")

tcs=int(input())
l=int(input())
for i in range(1,tcs+1):
    n=int(input())
    Breakup(n)

```

```
5
100
130
YES
110
YES
95
NO
100
YES
190
YES
```

Ali and Heling innocent people

In [ ]:
```
# Ali and Helping innocent people
s=input()
li=['A','E','I','O','U','Y']
if len(s)==9:
    if s[2] not in li:
        if s[6]=='-':
            if ((int(s[0])+int(s[1]))%2==0 and (int(s[3])+int(s[4]))%2==0 an
                print("valid")
            else:
                print("invalid")
        else:
            print("invalid")

    else:
        print("invalid")
else:
    print("invalid")
```

Ladderophilia

In [11]:
```
# Ladderophilia
n=int(input())
for i in range(1,n+1):
    for j in range(2):
        print('*   *')
    print("*****")
for j in range(2):
        print('*   *')
```

```
4
*   *
*   *
*****
*   *
*   *
*****
*   *
*   *
*****
*   *
*   *
*****
*   *
*   *
```

In [ ]:
```
1
```