Heading 1

Heading 2

Heading 3

Heading 4

Heading 5

Heading 6

- line 1
- line 2
    - subline 1
    - subline 2

Bold

italic

In [3]:
```
dir(set)
```

Out[3]:
```
['__and__',
 '__class__',
 '__contains__',
 '__delattr__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__gt__',
 '__hash__',
 '__iand__',
 '__init__',
 '__init_subclass__',
 '__ior__',
 '__isub__',
 '__iter__',
 '__ixor__',
 '__le__',
 '__len__',
 '__lt__',
 '__ne__',
 '__new__',
 '__or__',
 '__rand__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__ror__',
 '__rsub__',
 '__rxor__',
 '__setattr__',
 '__sizeof__',
 '__str__',
 '__sub__',
 '__subclasshook__',
 '__xor__',
 'add',
 'clear',
 'copy',
 'difference',
 'difference_update',
 'discard',
 'intersection',
 'intersection_update',
 'isdisjoint',
 'issubset',
 'issuperset',
 'pop',
 'remove',
 'symmetric_difference',
 'symmetric_difference_update',
 'union',
 'update']
```

In [6]:
```
s1={2,6,43,7}
s2={83,21,75,9}
print(s1)
```

```
{2, 43, 6, 7}
```

In [10]:
```
s1.add(88)
print(s1)
```

```
{2, 6, 7, 43, 88}
```

In [11]:
```
s1.update(s2)
s1
```

Out[11]:
```
{2, 6, 7, 9, 21, 43, 75, 83, 88}
```

In [12]:
```
s2.pop()
s2
```

Out[12]:
```
{21, 75, 83}
```

In [15]:
```
s2.discard(75)
s2
```

Out[15]:
```
{21, 83}
```

In [16]:
```
a=[2,3]
b={8,43}
b.update(a)
b
```

Out[16]:
```
{2, 3, 8, 43}
```

In [24]:
```
s1={1,2,5,7,3}
s2={5,4,3,9}
s1.union(s2)
```

Out[24]:
```
{1, 2, 3, 4, 5, 7, 9}
```

In [22]:
```
s1|s2
```

Out[22]:
```
{1, 2, 3, 4, 5, 7, 9}
```

In [23]:
```
s1.intersection(s2)
```

Out[23]:
```
{3, 5}
```

In [25]:
```
s1-s2
```

Out[25]:
```
{1, 2, 7}
```

In [26]:
```
s2-s1
```

Out[26]: {4, 9}

In [27]:
```
s1&s2
```

Out[27]: {3, 5}

In [28]:
```
s1.difference(s2)
```

Out[28]: {1, 2, 7}

In [30]:
```
a=s1.symmetric_difference(s2)
a
```

Out[30]: {1, 2, 4, 7, 9}

In [32]:
```
s1.issuperset(s2)
```

Out[32]: False

In [33]:
```
s2.issuperset(s1)
```

Out[33]: False

In [34]:
```
s1.issubset(s2)
```

Out[34]: False

In [35]:
```
s2.issubset(s1)
```

Out[35]: False

In [39]:
```
x={23,353,674,88}
y={22,77,23,88}
x
```

Out[39]: {23, 88, 353, 674}

In [40]:
```
x.intersection_update(y)
x
```

Out[40]: {23, 88}

Regular Expressions

In [52]:
```
import re
s1="Python Workshop"
a= re.match("P",s1)
print(a)
if a:
    print("Match")
else:
    print("Not Match")
```

```
None
Not Match
```

In [49]:
```
import re
s1="Python Workshop"
a= re.search("o",s1)
print(a)
if a:
    print("Match")
else:
    print("Not Match")
```

```
<re.Match object; span=(4, 5), match='o'>
Match
```

In [87]:
```
import re
s1="Python Workshop"
a= re.findall("h",s1)
print(a)
if a:
    print("Match")
else:
    print("Not Match")
```

```
['h', 'h']
Match
```

In [69]:
```
import re
s1="Python Workshop"
a= re.findall("^p",s1)
print(a)
if a:
    print("Match")
else:
    print("Not Match")
```

```
[]
Not Match
```

In [70]:
```
import re
s1="Python Workshop"
a= re.findall("p$",s1)
print(a)
if a:
    print("Match")
else:
    print("Not Match")
```

```
['p']
Match
```

In [81]:
```
import re
s1="Python Workshop"
a= re.findall("k...",s1)
print(a)
if a:
    print("Match")
else:
    print("Not Match")
```

```
['ksho']
Match
```

In [73]:
```
import re
s1="Python Workshop"
a= re.findall("\s",s1)
print(a)
if a:
    print("Match")
else:
    print("Not Match")
```

```
[' ']
Match
```

In [74]:
```
import re
s1="Python Workshop"
a= re.findall("\S",s1)
print(a)
if a:
    print("Match")
else:
    print("Not Match")
```

```
['P', 'y', 't', 'h', 'o', 'n', 'W', 'o', 'r', 'k', 's', 'h', 'o', 'p']
Match
```

In [84]:
```
import re
s1="Python Workshop"
a= re.findall("o*",s1)
print(a)
if a:
    print("Match")
else:
    print("Not Match")
```

```
['', '', '', '', 'o', '', '', '', 'o', '', '', '', '', 'o', '', '']
Match
```

In [85]:
```
import re
s1="Python Workshop"
a= re.findall("o+",s1)
print(a)
if a:
    print("Match")
else:
    print("Not Match")
```

```
['o', 'o', 'o']
Match
```

In [90]:
```
import re
s1="Python Workshop"
a= re.findall("o.+",s1)
print(a)
if a:
    print("Match")
else:
    print("Not Match")
```

```
['on Workshop']
Match
```

In [91]:
```
import re
s1="Python Workshop"
a= re.findall("o.*",s1)
print(a)
if a:
    print("Match")
else:
    print("Not Match")
```

```
['on Workshop']
Match
```

In [93]:
```
import re
s1="Python Workshop"
a= re.findall("[n-p]",s1)
print(a)
if a:
    print("Match")
else:
    print("Not Match")
```

```
['o', 'n', 'o', 'o', 'p']
Match
```

In [98]:
```
import re
s1="python Workshop"
a= re.findall("^[a-q]",s1)
print(a)
if a:
    print("Match")
else:
    print("Not Match")
```

```
['p']
Match
```

In [101]:
```
import re
s1="0123456789"
a= re.findall("[0-9]",s1)
print(a)
if a:
    print("Match")
else:
    print("Not Match")
```

```
['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']
Match
```

- **ph. no. pattern validator**

In [159]:
```
import re
pattern="^[6-9][0-9]{9}$|^[0][6-9][0-9]{9}|^[+][9][1][6-9][0-9]{9}$"
n=input()
if re.match(pattern,n):
    print("Valid Input")
else:
    print("Invalid Input")
```

```
+919676742518
Valid Input
```

In [185]:
```
import re
def phonenumberValidator(phone):
    pattern="^[6-9][0-9]{9}$|^[0][6-9][0-9]{9}|^[+][9][1][6-9][0-9]{9}$"
    if re.match(pattern,str(phone)):
        return True
    return False
phonenumberValidator(phone=int(input()))
```

```
9676742518
```

Out[185]: True

- **email validator**

In [169]:
```
import re
pattern="^[0-9a-z][0-9a-z._]{4,13}[@][0-9a-z]{3,18}[.][a-z]{2,4}|[.][a-z]{2,4}
$"
n=input()
if re.match(pattern,n):
    print("Valid Input")
else:
    print("Invalid Input")
```

```
dileep@gmail.com
Valid Input
```

In [170]:
```
import re
def emailValidator(email):
    pattern="^[0-9a-z][0-9a-z._]{4,13}[@][0-9a-z]{3,18}[.][a-z]{2,4}|[.][a-z]
{2,4}$"
    if re.match(pattern,str(email)):
        return True
    return False
emailValidator(email=(input(" ")))
```

```
 dileep@gmail.com
```

Out[170]: True

In [188]:
```
contacts={"name1":[9948441174,"narasingarao1010@gmail.com"],"name2":[812544117
4,"chandinigorli194@gmail.com"]}
def addcontacts(name,phone,email):
    if name in contacts:
        print(name,"already exists")
    else:
        if not phonenumberValidator(phone):
            print("Invalid Phone Number")
            return
        if not emailValidator(email):
            print("Invalid email address")
            return
        newcontact=[]
        newcontact.append(phone)
        newcontact.append(email)
        contacts[name]=newcontact
        print(name,"added,successfully")
    return
addcontacts("name3",7680929089,"sweety422@gmail.com")
addcontacts("name4",9676742518,"dileep4799@gmail.com")
```

```
name3 added,successfully
name4 added,successfully
```

In [ ]:
```

```