In [ ]:
```
a=10
b=20
a+b
```

In [ ]:
```
a="Helo world"
type(a)
```

In [ ]:
```
a="10"
type(a)
```

In [ ]:
```
a="10"
b=int(a)
type(b)
```

History Of python

Monty Python fly circus

- In 1989 started implemented of python with version 0.9
- In 1991 Implemented 1.0 version with features of lambda, map,filter,reduce
- In 2000 implemented 2.0 version with features of datastructures like List, Tuple ,Dictionary,sets..etc
- In 2008 Republished 3.0 version with features with large no of libraries,grabagecollection And FastWork

Features of Python

- easy to learn
- crossplatform
- scalable
- portable

In [ ]:
```
# KEY WORDS IN PYTHON

# Key words or Rederved words in python

import keyword
dir(keyword)
keyword.kwlist
help()
```

Operators

- Arthematic Operator (+,-,,/,%,*,//)
- Relational or Comparasion operator (>,<,<=,>=,==,!=)

- Assingment Operator (+=,-=,=,/=,*=,//=)
- Logical Operator (and,or,not)
- Bitwise Operator (&,|,^,>>,<<)
- Membership operator
- Indentity Operator

In [ ]:
```
a=10
b=20
print(a+b)
print(a-b)
print(a/b)
print(a*b)
print(a%b)
print(a**b)
print(a//b)
print(pow(a,b))
```

In [ ]:
```
x=100
y=90
print(x>y)
print(x<y)
print(x==y)
print(x!=y)
```

In [ ]:
```
#Assinment operator

p=10
p+=40
p-=30
p*=3
p/=3
p//=3
p**=5
print(p)
```

In [ ]:
```
p=10
p+=40
p-=30
p*=3
p/=3
p//=2
print(p)
```

In [ ]:
```
#Logical Operators
#And
print(False and False)
print(False and True)
print(True and False )
print(True and True)
```

In [ ]:
```
# or
print(False or False)
print(False or True)
print(True or False )
print(True or True)
```

In [ ]:
```
#not
print(not(False and False))
print(not(False and True))
print(not(True and False))
print(not(True and True))
```

In [ ]:
```
#not
print(not(False or False))
print(not(False or True))
print(not(True or False))
print(not(True or True))
```

In [ ]:
```
bin(2)
```

In [ ]:
```
oct(8)
```

In [ ]:
```
#Bitwisw operators
print(2&2)
print(10|5)
print(5^6)
bin(2)
```

In [ ]:
```
x=100
y=10000
print(x is y)
print(x is not y)
```

In [ ]:
```
a=b=c=10
print(a)
print(b)
print(c)
```

In [ ]:
```
a=10
b=20
suresh=a+b
print(suresh)
```

In [ ]:
```
a,b,c=10,20,30
print(a)
print(b)
print(c)
```

In [ ]:
```
#static variable

a=10
b=100
print(a+b)
```

In [ ]:
```
a=10
b=20
print("Addtion of two numbers",a+b)
print('Addition of',a,b,'=',a+b)
print('Addition of %d + %d = %d'%(a,b,a+b))
```

In [ ]:
```
x=int(input("Enter the x value: "))
y=int(input("Enter the y value: "))
print("Addtion of ",x + y)
print("Addtion %d + %d = %d"%(x,y,x+y))
```

Expression

1. Take 5 inputs from user
2. to perform operation
3. store result in final value
4. print final result

In [1]:
```
a=int(input("Enter the a value: "))
b=int(input("Enter the b value: "))
c=int(input("Enter the c value: "))
d=int(input("Enter the d value: "))
e=int(input("Enter the e value: "))
x=a+b-c*d/e
print(x)
```

```
Enter the a value: 5
Enter the b value: 8
Enter the c value: 9
Enter the d value: 6
Enter the e value: 2
-14.0
```

Conditonal Statements