```
a=10
b=20
c=12.5
d="hello"
e=True
```

```
a=b=c=d
```

```
#Demonstration of Relational/Comparison Operators
```

```
a,b,c=10,20,10
```

```
a==c
```

```
    True
```

```
a!=c
```

```
    False
```

```
a<=c
```

```
    True
```

```
a>=b
```

```
    False
```

Assignment operators

```
a,b,c=5,2,0
```

```
c=a+b
print(c)
```

```
    7
```

```
#a=a+2
a=5
a+=3
print(a)
```

```
    8
```

```
a=5
```

```
print(a)
```

```
2
```

```
a=5
a*=3
print(a)
```

```
15
```

```
a=5
a/=3
print(a)
```

```
1.6666666666666667
```

```
a=5
a%=3
print(a)
```

```
2
```

```
a=5
a**=3
print(a)
```

```
125
```

```
a=5
a//=3
print(a)
```

```
1
```

Bitwise Operator

```
a=60            #0011 1100
b=13            #0000 1101
```

```
a&b              #0000 1100
```

```
12
```

```
a|b               #0011 1101
```

```
61
```

```
a^b         #0011 0001
```

```
49
```

```
~a
```

```
-61
```

```
a<<2
```

```
240
```

```
a>>3
```

```
7
```

```

```

Logical Operators: and, or, not

```
a=10
b=20
```

```
(a<0) or (b>0)
```

```
True
```

```
not((a<0) or (b>0))
```

```
False
```

Mmebership operators: in, not in

```
a=10
b=2
seq=[1,2,3,4,5]
```

```
a not in seq
```

```
True
```

identity operator: is, is not

```
a=10
b=10
```

```
b=10
c="hello"
d="hello"
e="HELLO"
```

```
d is not e
```

```
True
```

```
c==d
```

```
True
```

Decision making statements: if, if.....else, if...elif.....else statements

```
#to see whether the number is positive, negative or zero..... indent (spa
#if statement
a=-5
if(a>0):
  print("a is positive")
  print("hello")
  print("hii")
if(a<0):
  print("a is negative")
if(a==0):
  print("a is zero")
```

```
a is negative
```

```
a=0
if(a>0):
  print("a is positive")
  print("hello")
  print("hi")
elif(a<0):
  print("a is negative")
else:
  print("a is zero")
```

```
a is zero
```

```
#find the maximum value
a,b=3,2
```

```
if(a>b):
  print("a is maximum")
if(a<b):
  print("b is maximum")
if(a==b):
  print("a and b are same")
```

```
    a is maximum
```

```
if(a>b):
  print("a is maximum")
elif(a<b):
  print("b is maximum")
else:
  print("a and b are same")
```

```
    b is maximum
```

```
a=150
if(a<100):
  print("a is smaller than 100 excluding")
elif(a>=100 and a<125):
  print("a lies between 100 including and 125 excluding")
elif(a>=125 and a<150):
  print("a lies between 125 including and 150 excluding")
elif(a>=150 and a<200):
  print("a is between 150 including and 200 excluding")
else:
  print("a is greater than 200")
```

```
    a is between 150 including and 200 excluding
```

```
#to determine whether the number is odd or even
#a%2==0: even || a%2==1: odd   %--> remainder
a=-4
if(a%2==0 and a!=0 and a>0):
  print("a is even")
elif(a%2!=0):
  print("a is odd")
else:
  print("a is zero")
```

```
    a is zero
```

```
#to determine whether the number is odd or even
#a%2==0: even || a%2==1: odd   %--> remainder
a=0
if(a%2==0):
  print("a is even")
if(a%2!=0):
  print("a is odd")
if(a==0):
  print("a is zero")
```

```
    a is even
    a is zero
```

```
#conditional operator
```

```
a,b=10,10
```

```
#to find maximum of 2 numbers
if(a>b):
  print(a)
else:
  print(b)
```

```
10
```

```
#conditional operator
```

```
a if a>b else b
```

```
10
```

Strings

```
a="Hello World"
b="Students"
```

```
name='ydbhyvubdyfubv'
surname='fdryfbdfhbv'
name+'.'+surname+'@daiict.ac.in'
```

```
'ydbhyvubdyfubv.fdryfbdfhbv@daiict.ac.in'
```

indexing of the string starts from 0

```
a[0:4]  #starts from 0 to 4.... last number is not included
```

```
'Hell'
```

```
a[:4]
```

```
'Hell'
```

```
a
```

```
'Hello World'
```

```
a[4:]
```

```
'o World'
```

```
o World
```

```
a[2:8]
```

```
'llo Wo'
```

multiple lines as a string

```
abc="""Python is an interpreted high-level general-purpose programming language.
use of significant indentation. Its language constructs as well as its object-or
logical code for small and large-scale projects.[30]"""
```

```
abc
```

```
'Python is an interpreted high-level general-purpose programming language.
Its design philosophy emphasizes code readability with its\nuse of signifi
cant indentation. Its language constructs as well as its object-oriented a
```

```
a="this is a python learning course"
b="THIS IS A PYTHON LEARNING COURSE"
```

```
#length of a string
len(a)  #index: 0 to len-1
```

```
32
```

```
a[len(a)-1]
```

```
'e'
```

```
#capitalize the first character
a.capitalize()
```

```
'This is a python learning course'
```

```
#lowercase
b.lower()
```

```
'this is a python learning course'
```

```
#uppercase
a.upper()
```

```
'THIS IS A PYTHON LEARNING COURSE'
```

```
#titlecase
a.title()
```

```
'This Is A Python Learning Course'
```

```
b.isupper()
```

```
True
```

```
a.istitle()
```

```
False
```

```
a
```

```
'this is a python learning course'
```

```
#frequency/count of a character
a.count('i')
```

```
3
```

```
a.count('i',5,17)
```

```
1
```

```
a[5:17]
```

```
'is a python '
```

```
#find the index of a character/substring
a.find('isa')
```

```
-1
```

```
a.index('isa')
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call
last)
<ipython-input-170-0d1ebd41312f> in <module>()
----> 1 a.index('isa')

ValueError: substring not found

SEARCH STACK OVERFLOW
```

```
#collection of numbers, characters and special characters
c="abcd@1234%#"
```

```
d="1234"
```

```
#alphaneumeric??
```

```
#alphanumeric
d.isalnum()
```

```
True
```

```
#alphabetic
d.isalpha()
```

```
True
```

```
#only digits
d.isdigit()
```

```
True
```

```
#only spaces
d="           "
d.isspace()
```

```
True
```

```
a
```

```
'this is a python learning course'
```

```
#max character of a string--> ASCII equivalent
max(a)
```

```
'y'
```

```
min(a)
```

```
' '
```

```
ord('a')  #order  find the ascii equivalent of a character
```

```
97
```

```
#ascii to character
chr(9798)
```

```
'♦'
```

```
chr(97)
```

```
'a'
```

```
str(97)
```

```
'97'
```

```
#split method
a
```

```
'this is a python learning course'
```

```
a.split(' ')
```

```
['this', 'is', 'a', 'python', 'learning', 'course']
```

```
#join function
```

```
'#'.join(a.split(' '))
```

```
'this#is#a#python#learning#course'
```

```
#replace
a
```

```
'this is a python learning course'
```

```
a.replace('i','e')
```

```
'thes es a python learneng course'
```

```
a.replace('i','e',2)
```

```
'thes es a python learning course'
```

```
password="adCD14@"
countsmaller=0
countcaps=0
countdigits=0
countspc=0
if(a[0])
a[1]
a[2]
a[3]
a[4]
```

```
password="aA1#"
password[0]
```

```
'a'
```

```
countsmaller=0
countcaps=0
```

```
countdigits=0
countspc=0
```

```
if(password[0].isalnum()):
  if(password[0].isupper()):
    countcaps+=1
  elif(password[0].islower()):
    countsmaller+=1
  elif(password[0].isdigit()):
    countdigits+=1
  else:
    countspc+=1
if(password[1].isalnum()):
  if(password[1].isupper()):
    countcaps+=1
  elif(password[1].islower()):
    countsmaller+=1
  elif(password[1].isdigit()):
    countdigits+=1
  else:
    countspc+=1
if(password[2].isalnum()):
  if(password[2].isupper()):
    countcaps+=1
  elif(password[2].islower()):
    countsmaller+=1
  elif(password[2].isdigit()):
    countdigits+=1
  else:
    countspc+=1
if(password[3].isalnum()):
  if(password[3].isupper()):
    countcaps+=1
  elif(password[3].islower()):
    countsmaller+=1
  elif(password[3].isdigit()):
    countdigits+=1
  else:
    countspc+=1
```

```
print(countsmaller)
print(countcaps)
print(countdigits)
print(countspc)
```

```
1
1
1
0
```

```
if(countsmaller>0 and countcaps>0 and countdigits>0 and countspc>0):
  print("valid password")
```

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

```
invalid password
```

```

```