```
#print a dictionary
x={
    "name":"praneeth",
    "friend":"rahul",
    "cllg":"gitam",
    "realation":"friendship"
   }
print(x)
```

```
{'name': 'praneeth', 'friend': 'rahul', 'cllg': 'gitam', 'realation': 'friendship'}
```

```
#Print the "brand" value of the dictionary
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019"
  }
print(x["brand"])
```

```
royalenfield
```

```
#Duplicates Not Allowed
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019",
    "since":"2022"
  }
print(x)
```

```
{'brand': 'royalenfield', 'model': 'continental gt 650', 'since': '2022'}
```

```
#type
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019",
    "since":"2022"
  }
print(len(x))
```

```
3
```

```
#data type
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019",
    "since":"2022"
  }
print(type(x))
```

```
     <class 'dict'>
```

```
#Accessing Items
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019",
    "since":"2022"
  }
y=x["model"]
print(y)
```

```
     continental gt 650
```

```
#get
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019",
    "since":"2022"
  }
y=x.get("model")
print(y)
```

```
     continental gt 650
```

```
#printing keys
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019",
    "since":"2022"
  }
y=x.keys()
print(y)
```

```
     dict_keys(['brand', 'model', 'since'])
```

```
#adding a new items
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019",
    "since":"2022"
  }
y=x.keys()
print(y)
x["colour"]="silver"
print(y)
```

```
     dict_keys(['brand', 'model', 'since'])
     dict_keys(['brand', 'model', 'since', 'colour'])
```

```
#printing values
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019",
    "since":"2022"
  }
y=x.values()
print(y)
```

```
dict_values(['royalenfield', 'continental gt 650', '2022'])
```

```
#adding a new items
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019",
    "since":"2022"
  }
y=x.values()
print(y)
x["colour"]="silver"
print(y)
```

```
dict_values(['royalenfield', 'continental gt 650', '2022'])
dict_values(['royalenfield', 'continental gt 650', '2022', 'silver'])
```

```
#printing items
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019",
    "since":"2022"
  }
y=x.items()
print(y)
```

```
dict_items([('brand', 'royalenfield'), ('model', 'continental gt 650'), ('since', '2
```

```
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019",
    "since":"2022"
  }
y=x.items()
print(y)
x["colour"]="silver"
print(y)
```

```
dict_items([('brand', 'royalenfield'), ('model', 'continental gt 650'), ('since', '2
dict_items([('brand', 'royalenfield'), ('model', 'continental gt 650'), ('since', '2
```

```
x = {
  "brand": "Ford",
  "model": "Mustang",
  "year": 1964
}
if "model" in x:
  print("Yes, 'model' is one of the keys in the thisdict dictionary")
```

```
Yes, 'model' is one of the keys in the thisdict dictionary
```

```
#change the year
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019"
  }
x["since"]="2021"
print(x)
```

```
{'brand': 'royalenfield', 'model': 'continental gt 650', 'since': '2021'}
```

```
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019"
  }
x.update({"since":"2021"})

print(x)
```

```
{'brand': 'royalenfield', 'model': 'continental gt 650', 'since': '2021'}
```

```
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019"
  }
x["color"] = "red"
print(x)
```

```
{'brand': 'royalenfield', 'model': 'continental gt 650', 'since': '2019', 'color': '
```

```
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019"
  }
  }
x.update({"color": "red"})
print(x)
```

```
{'brand': 'royalenfield', 'model': 'continental gt 650', 'since': '2019', 'color': 'r
```

```
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019"
  }
x.pop("model")
print(x)
```

```
{'brand': 'royalenfield', 'since': '2019'}
```

```
x={
    "brand":"royalenfield",
    "model":"continental gt 650",
    "since":"2019"
  }
x.clear()
print(x)
```

```
{}
```

```
x={"praneeth","yaswanth","rahul"}
print(x)
```

```
     {'rahul', 'yaswanth', 'praneeth'}
```

```
#duplicates not allowed
x={"praneeth","manish","bhanu","praneeth"}
print(x)
```

```
     {'manish', 'bhanu', 'praneeth'}
```

```
#len of set
x=("praneeth","manish","bhanu","yaswanth","rahul")
print(len(x))
```

```
     5
```

```
#data types
x={"praneeth"}
y={"yaswanth"}
z={"manish","manish"}
print(x)
print(y)
print(z)
```

```
     {'praneeth'}
     {'yaswanth'}
     {'manish'}
```

```
x={"praneeth","sai krishna","yaswanth"}
for x in x:
  print(x)
```

```
     sai krishna
     yaswanth
     praneeth
```

```
#praneeth is in set
x={"praneeth","sai krishna","yaswanth"}
print("praneeth" in x)
```

```
     True
```

```
#adding an item
x={"praneeth","sai krishna","yaswanth"}
x.add("seshu")
print(x)
```

```
     {'seshu', 'sai krishna', 'yaswanth', 'praneeth'}
```

```
#adding a set to another set
x={"praneeth","sai krishna","yaswanth"}
y={"manish","rahul","bhanu"}
x.update(y)
print(x)
```

```
{'bhanu', 'sai krishna', 'yaswanth', 'rahul', 'manish', 'praneeth'}
```

```
#Add Any Iterable
x={"praneeth","sai krishna","yaswanth"}
y=["manish","rahul"]
x.update(y)
print(x)
```

```
{'sai krishna', 'yaswanth', 'manish', 'rahul', 'praneeth'}
```

```
#Remove Item
x={"praneeth","sai krishna","yaswanth"}
x.remove("yaswanth")
print(x)
```

```
{'sai krishna', 'praneeth'}
```

```
#remove using pop()
x={"praneeth","sai krishna","yaswanth"}
y=x.pop()
print(y)
print(x)
```

```
sai krishna
{'yaswanth', 'praneeth'}
```

```
#clear the set
x={"praneeth","sai krishna","yaswanth"}
x.clear()
print(x)
```

```
set()
```

```
#loop items
x={"praneeth","sai krishna","yaswanth"}
for x in x:
  print(x)
```

```
sai krishna
yaswanth
praneeth
```

```
#join two sets
x={"praneeth","sai krishna","yaswanth"}
```

```
y={"manish","rahul"}
z=x.union(y)
print(z)
```

```
{'sai krishna', 'yaswanth', 'rahul', 'manish', 'praneeth'}
```

```
#intersection_update()
x={"praneeth","sai krishna","yaswanth"}
y={"rahu","praneeth","yaswanth"}
x.intersection_update(y)
print(x)
```

```
{'yaswanth', 'praneeth'}
```

```
#intersection() method
x={"praneeth","sai krishna","yaswanth"}
y={"rahu","praneeth","yaswanth"}
z=x.intersection(y)
print(z)
```

```
{'yaswanth', 'praneeth'}
```

```
#symmetric_difference_update
x={"praneeth","sai krishna","yaswanth"}
y={"rahul","praneeth","yaswanth"}
x.symmetric_difference_update(y)
print(x)
```

```
{'rahul', 'sai krishna'}
```

```
#symmetric_difference() method
x={"praneeth","sai krishna","yaswanth"}
y={"rahul","praneeth","yaswanth"}
z=x.symmetric_difference(y)
print(z)
```

```
{'sai krishna', 'rahul'}
```

● ✕

strings

```
a=praneeth
print(a)
print(type(a))
print(a[1])
```

```
a="RRR movie has realsed"
print ("movie" in a)
```

```
     True
```

```
a="RRR movie has realsed"
print("ntr" in a)
```

```
     False
```

slicing

```
a="RRR movie has realsed"
print(a[2:5])
```

```
     R m
```

```
a="RRR has been released"
print(a.replace("RRR","radheyshyam"))
```

```
     radheyshyam has been released
```

```
x="praneeth chowdary"
print(x.replace("p","k"))
```

```
     kraneeth chowdary
```

```
x="praneeth,chowdary"
print(x.split(","))
```

```
     ['praneeth', 'chowdary']
```

```
a="praneeth"
b="chowdary"
c=a+b
print(c)
```

```
     praneethchowdary
```

```
a="praneeth"
b="chowdary"
c=a+" "+b
print(c)
```

```
     praneeth chowdary
```

```
age=18
txt="my name is praneeth,and i am {}"
print(txt.format(age))
```

```
     my name is praneeth,and i am 18
```

```
sprite=2
thumpsup=3
mazza=4
myorder="i want sprite {} and thumpsup {} and mazza {} for 90 rupees"
print(myorder.format(sprite,thumpsup,mazza))
```

```
     i want sprite 2 and thumpsup 3 and mazza 4 for 90 rupees
```

```
sprite=2
thumpsup=3
mazza=4
myorder="i want sprite {2} and thumpsup {0} and mazza {1} for 90 rupees"
print(myorder.format(sprite,thumpsup,mazza))
```

```
     i want sprite 4 and thumpsup 2 and mazza 3 for 90 rupees
```

```
a="sai krishna is a\"lafoot fellow\" in the class"
print(a)
```

```
     sai krishna is a"lafoot fellow" in the class
```

```
a="praneeth \\chowdary"
print(a)
```

```
     praneeth \chowdary
```

```
a="praneth\nchowdary"
print(a)
```

```
     praneth
     chowdary
```

```
a="praneeth\bchowdary"
print(a)
```

```
     praneetchowdary
```

```
a="praneeth\fchowdary"
print(a)
```

```
praneethchowdary
```

```
a="praneeth\rchowdary"
print(a)
```

```
chowdary
```

average of two numbers

```
a=29
b=58
c=38
average = (a+b+c)/3
print(average)
```

```
     41.666666666666664
```

multiplication of three numbers

```
a=15
b=46
c=37
mul=a*b*c
print(mul)
```

```
     25530
```

dynamic addition of 2 numbers

```
a=int(input("enter a number"))
b=int(input("enter a number"))
addition=(a+b)
print(addition)
print(type(addition))
```

```
     enter a number47
     enter a number32
     79
     <class 'int'>
```

dynamic of 3 numbers

```
a=int(input("enter a number"))
b=int(input("enter a number"))
c=int(input("enter a number"))
multi=(a*b*c)
```

```
print(multi)
print(type(multi))
```

```
enter a number20
enter a number37
enter a number28
20720
<class 'int'>
```

type conversions

1.int to float

```
a=10
print(type(a))
b=float(a)
print(type(b))
print(b)
```

```
<class 'int'>
<class 'float'>
10.0
```

2.float to int

```
a=10
print(type(a))
b=int(a)
print(type(b))
print(b)
```

```
<class 'int'>
<class 'int'>
10
```

3.int to string

```
a=10
print(type(a))
b=str(a)
print(type(b))
print(b)
```

```
<class 'int'>
<class 'str'>
10
```

4.float to string

```
a=10
print(type(a))
b=str(a)
print(type(b))
print(b)
```

```
     <class 'int'>
     <class 'str'>
     10
```

5.string to int

```
a=10
print(type(a))
b=int(a)
print(type(b))
print(b)
```

```
     <class 'int'>
     <class 'int'>
     10
```

Boolean

```
a=34
b=45
c=a<b
print(bool(c))
```

```
     True
```

Boolean to int

```
a=10
print(type(a))
b=bool(a)
print(type(b))
print(b)
```

```
     <class 'int'>
     <class 'bool'>
     True
```

boolean to string

```
a=10
print(type(a))
b=str(a)
print(type(b))
```

```
<class 'int'>
<class 'str'>
```

complex

```
a=5+2j
print(type(a))
c=str(a)
print (type(c))
```

```
<class 'complex'>
<class 'str'>
```

if

```
a=20
if a>=18:
    print("eligible for voting")
else:
    print("not eligible for voting")
```

```
eligible for voting
```

```
a=6
if a>=18:
    print("eligible for voting")
else:
    print("not eligible for voting")
```

```
not eligible for voting
```

find the greatest of three numbers

```
num1=float(input("enter first number:"))
num2=float(input("enter second number:"))
num3=float(input("enter third number:"))
if (num1>num2) and (num1>num3):
     largest=num1
elif (num2>num1) and (num2>num3):
     largest=num2
else:
     largest=num3
print("the largest numbers is:",largest)
```

```
enter first number:15
enter second number:12
enter third number:09
the largest numbers is: 15.0
```

print even num from 1 to 100

```
for i in range(1,100,2):
    print(i+1)
```

```
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
42
44
46
48
50
52
54
56
58
60
62
64
66
```

```
68
70
72
74
76
78
80
82
84
86
88
90
92
94
96
98
100
```

loops

while loop

```
n=int(input("enter n:"))
sum=0
i=1
while i<=n:
    sum=sum+1
    i=i+1
print("the sum is:",sum)
```

```
enter n:5
the sum is: 5
```

leap year

```
year = 2022
if (year % 400 == 0) and (year % 100 == 0):
    print("{0} is a leap year".format(year))
elif (year % 4 ==0) and (year % 100 != 0):
    print("{0} is a leap year".format(year))
else:
    print("{0} is not a leap year".format(year))
```

```
2022 is not a leap year
```

triangle

```
a=int(input("enter a:"))
b=int(input("enter b:"))
c=int(input("enter c:"))
if a==b and b==c:
    print("it is a isosceles triangle")
elif a==b or b==c or c==a:
    print("it is a equilateral triangle")
else:
    print("it is a scalar")
```

```
enter a:4
enter b:5
enter c:5
it is a equilateral triangle
```

min among 3

```
a=int(input("enter a"))
b=int(input("enter b:"))
c=int(input("enter c:"))
smallest=0

if a < b and a < c :
    smallest = a
if b < a and b < c :
    smallest = b
if c < a and c < b :
    smallest = c

print(smallest, "is the smallest of three numbers.")
```

```
enter a56
enter b:56
```