print many hello world .

```
print("Hello World\nHell0 World\nHello World")
```

```
Hello World
Hell0 World
Hello World
```

```
print("Hello"+" "+"Himanshu")
```

```
Hello Himanshu
```

Sting

```
a ="company"
print(a)
```

```
company
```

```
yah="python"   #sting written in single , double or triple Quotes.
they can be alphabets, characters and special characters.
print(type(yah))
```

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

Indexing in string given by [ ]

```
ram = "siddique"
print(ram[4])
```

```
i
```

```
ram = "siddique"
print(ram[-1])
```

```
e
```

```
name = "Himanshu"
game="football"
print("My name is",name,"i love",game,".")
```

```
My name is Himanshu i love football .
```

```
name = "Himanshu"
game="football"
print("My name is "+name+" i love "+game+".")
```

```
My name is Himanshu i love football.
```

f string in python very imp

```
name="Himanshu Salal"
age=23
height=1.7
print(f"Hi my name is {name}, I am {age} years old and my height is
{height} m.")
```

```
Hi my name is Himanshu Salal, I am 23 years old and my height is 1.7
m.
```

program to sum of two number

```
a=2
b=3
print(a+b)
```

```
5
```

Python prgram to find Square Root

```
a = 81
num_sqrt=a**0.5
print(int(float(num_sqrt)))
```

```
9
```

Write a program to calulate the area of traingle

```
a = 5
b = 6
c = 7
#first calulate area of perimeter.
s=(a+b+c)/2
#area of traingle
area=(s*(s-a)*(s-b)*(s-c))**0.5
print('The area of the triangle is %0.2f' %area)
```

```
The area of the triangle is 14.70
```

python program to check prime number?

```
num =int(input("Enter a number: "))
if num==0 or num==1:
  print(num,"is not a prime number")
elif num>1:
  for i in range(2,num):
    if(num%i)==0:
       print(num,"is not a prime nunber")
       print(i,num//i,"is",num)
       break
```

```
    else:
       print(num,"is a prime number")
else:
  print (num,"is not a prime number")


Enter a number: 6
6 is not a prime nunber
2 3 is 6
```

Program to check number is even or odd

```
num=int(input("Enter a number: "))
if (num%2)==0:
  print("0 is a even".format(num))
else:
  print("0 is odd".format(num))


Enter a number: 1444
0 is a even

numbers = [12,36,55,24,88,96,76,44,45,36]
def even(x):
  return x%2==0

evens= list(filter(even, numbers))
print("Even numbers:", evens)

Even numbers: [12, 36, 24, 88, 96, 76, 44, 36]
```

By lambda() function we dont need to create function we can execute task.

```
numbers=[12,36,55,24,88,96,76,44,45,36]
evens = list(filter(lambda x: x%2==0, numbers))
print("Even numbers:", evens)

Even numbers: [12, 36, 24, 88, 96, 76, 44, 36]
```

program to find the largest among three number

```
num1=int(input("Enter a first number: "))
num2=int(input("Enter a second number: "))
num3=int(input("Enter a third number: "))
largest=num1=num2=num3
if(num1>=num2) and (num1>=3):
  largest=num1
elif(num2>num3) and (num2>=num1):
  largest=num2
else:
```

```
  lagest=num3

print("The lagest number is", largest)
```

```
Enter a first number: 56
Enter a second number: 98
Enter a third number: 90
The lagest number is 90
```

find all prime number bwtween 900 and 1000

```
lower=700
upper=900

print("Prime number between", lower, "and", upper, "are:")

for num in range(lower, upper + 1):
  if num>1:
    for i in range(2,num):
      if (num%i)==0:
        break
  else:
   print(num)
```

```
Prime number between 700 and 900 are:
```

How to Swap two number

```
num1=input("Enter a number:")
num2=input("Enter a number:")

print("Value of num1 before swapping:",num1)
print("Value of num2 before swapping:",num2)

temp=num1
num1=num2
num2=temp

print("Value of num1 after swapping:",num1)
print("Value of num2 after swapping:",num2)
```

```
Enter a number:55
Enter a number:76
Value of num1 before swapping: 55
Value of num2 before swapping: 76
Value of num1 after swapping: 76
Value of num2 after swapping: 55
```

Same program in different format

```
num1=input("Enter a number:")
num2=input("Enter a number:")

print("Value of num1 before swapping:",num1)
print("Value of num2 before swapping:",num2)

num1,num2=num2,num1

print("Value of num1 after swapping:",num1)
print("Value of num2 after swapping:",num2)
```

```
Enter a number:98
Enter a number:99
Value of num1 before swapping: 98
Value of num2 before swapping: 99
Value of num1 after swapping: 99
Value of num2 after swapping: 98
```

Number is prime or not

```
num=61
count=0
if num>1:
  for i in range(1,num+1):
    if(num%i==0):
      count=count+1

  if count==2:
    print("number is prime")
  else:
    print("num is not prime")
```

```
number is prime
```

Nested for loop

```
list1 =["hello","Welcome","hi"]
list2 =["Shyam","Karan","Mohan"]
for items in list1:
  for names in list2:
    print(items,names)
```

```
hello Shyam
hello Karan
hello Mohan
Welcome Shyam
Welcome Karan
Welcome Mohan
hi Shyam
```

```
hi Karan
hi Mohan
```

Nested for loop (break control statement)

```
list1 =["hello","Welcome","hi"]
list2 =["Shyam","Karan","Mohan"]
for items in list1:
  for names in list2:
    print(items,names)
    if items=="Welcome" and names=="Karan":
      break
  print("out from loop")
print("out from loop")
```

```
hello Shyam
hello Karan
hello Mohan
out from inner loop
Welcome Shyam
Welcome Karan
out from inner loop
hi Shyam
hi Karan
hi Mohan
out from inner loop
out from loop
```

Continue control statement in loop in this if (condition) is true the above iteration of for loop runs

```
count=1
while count<=10:
  print(count)
  count+=1
  if count==7:
    continue
  print("Hi")
print("out from lopp")
```

```
1
Hi
2
Hi
3
Hi
4
```

```
Hi
5
Hi
6
7
Hi
8
Hi
9
Hi
10
Hi
out from lopp
```

Pass control statement it is placeholder for future reference it output nothing at that but in fututre we can use it.

```
for i in range(1,11):
  pass
```

Conditional Statement or decision making statement (if else, elif, else)

```
'''if condition:
      do this
else:
  do this'''

{"type":"string"}

height= int(input("Enter height in feet:"))
if height>3:
  print("Token required")
else:
        print("no token required")



Enter height in feet:2
no token required
```

if condition

```
height= int(input("Enter height in feet:"))
if height>3:
  print("Token required")
print("aree hai bhai")
print("yaa")
```

```
Enter height in feet:7
Token required
aree hai bhai
yaa
```

Nested if means - if inside if

```
'''num =[1,2]
if condition:
  statement1
  if condition2:
    statement2("matlb prin")
else:
  statement'''
```

```
{"type":"string"}
```

```
height=int(input("Enter your height:"))
if height>3:
  print("you can ride")
  age=int(input("Enter your age:"))
  if age>18:
    print("you have to have to pay 500")
    if age<18:
      print("you have to pay 200")
  else:
    print("no entry")
print("bye")
```

```
Enter your height:7
you can ride
Enter your age:77
you have to have to pay 500
bye
```

elif - (else if) conditional statement if we have more condition than 3. we can see in code if ,elif ,else.

```
height=int(input("Enter your height:"))
if height>3:
  print("you can ride")
  age=int(input("Enter your age:"))
  if age<12:
    print("you have to have to pay 150")
  elif age<=18:
    print("you have to pay 250")
  else:
    print("pay 500")
else:
```

```
  print("can't ride")
print("bye")
```

```
Enter your height:12
you can ride
Enter your age:12
you have to pay 250
bye
```

function

```
def sum(a,b):
  c=a+b
  print("sum is:",c)
sum(22,3)
sum(12,3)
```

```
sum is: 25
sum is: 15
```

```
def greet(name):
  print(f"Hi {name}")
  print("are you from cs department?")
greet("Jerry")
greet("Jadega")
```

```
Hi Jerry
are you from cs department?
Hi Jadega
are you from cs department?
```

recursion function, also it is program of factorial of a number

```
def fact(n):  # isme n koi bhi number from 1 to n
  if n==0 or n==1:
    return 1
  else:
    return n*fact(n-1)     #reecursive case, mai fact function ko call kiya or uski value change ho gyi

f=fact(4)    #call fact function
print(f)
```

```
24
```

Recursion function , using sum of 1 to n numbers.

```
def sum(n):
  if n==1:
```

```
    return 1
  else:
    return n+sum(n-1)
a=sum(100)
print(a)
```

```
5050
```

local scope (it print inside of the function value)

```
a=10
def dinosor():
  a=15
  print(a)
dinosor()
```

```
15
```

global variable (it print outside of the function variable value.)

```
a= 111
def himansh():
  a=222
himansh()
print(a)
```

```
111
```

first inside function variable value then outside variable value

```
a= 111
def himansh():
  a=222
  print(a)
himansh()
print(a)
```

```
222
111
```

if not assigned value to inside function it go outside and print

```
a=10
def dinosor():
  print(a)
dinosor()
```

```
10
```

Global Keyword is used when you want to modify a global variable (scope) value.

```
a=10
def mood():
  global a
  a=a+15
  print(a)
mood()
```

```
25
```

global variable value modify in global space possible without using global keyword.

```
a=10
def mood():
  a
mood()
a=a+10
print(a)
```

```
20
```

Object Oriented Programming

class is a design or blue print, objects are instance(example) etc, variarble called attribute - laptop,name, camera etc, functions called methods(what they actually do).

Creating a class

```
class InstructorInfo:
  pass
intructor_1 = InstructorInfo()

class Instructor:
  pass

oran= Instructor()
print(type(oran))
```

```
<class '__main__.Instructor'>
```

```
class Jenny:
  pass
car_design=Jenny() #car_design is object
print(type(car_design))
```

```
<class '__main__.Jenny'>
```

```
class School:
  pass
teacher = School()
```

```
teacher.name= "Payal"  #Payal is attribute or variable
teacher.address="Gautam Budh Nagar"
print(teacher.name)
```

```
Payal
```

```
class School:
  def __init__(self):
    print("creating a new object")

teacher = School()
teacher.name= "Payal"  #Payal is attribute or variable
teacher.address="Gautam Budh Nagar"
print(teacher.name)
print(teacher.address)
```

```
creating a new object
Payal
Gautam Budh Nagar
```

```
class School:
  def __init__(self):
    print("creating a new object")

teacher = School()    #teacher is a object  # School class is called.
It will print creating a new object.
teacher.name= "Payal"  #Payal is attribute or variable
print(teacher.name)

teacher_2= School()   #teacher_2 is a object, we have called School
class.
teacher_2.address ="Gautam Budh Nagar"
print(teacher_2.address)
```

```
creating a new object
Payal
creating a new object
Gautam Budh Nagar
```

```
class Instructor:
    def __init__(self, instructor_name, address):  #we have iniitilize
the attribute create attribute as much as we need.
        self.name = instructor_name
        self.address = address


instructor_1 = Instructor("Payal", "Gurgoan")
print(instructor_1.name)  # Fixed typo
instructor_2 = Instructor("Salal", "Gurgoan")
print(instructor_2.name)
```

```
instructor_3 = Instructor("Sagar", "Gurgoan")
print(instructor_3.name, instructor_3.address)
```

```
Payal
Salal
Sagar Gurgoan
```

Class Methods(what the object do)

```
class Instructor:
    def __init__(self, instructor_name, address):  #we have iniitilize
the attribute create attribute as much as we need.
        self.name = instructor_name
        self.address = address
    def display(self):    # Method created
      print("Python")

instructor_1 = Instructor("Payal", "Gurgoan")
print(instructor_1.name)
instructor_1.display()       # calling Method
instructor_2 = Instructor("Salal", "Gurgoan")
print(instructor_2.name)
instructor_3 = Instructor("Sagar", "Gurgoan")
print(instructor_3.name, instructor_3.address)
```

```
Payal
Python
Salal
Sagar Gurgoan
```

```
class Instructor:
    def __init__(self, instructor_name, address):  #we have iniitilize
the attribute create attribute as much as we need.
        self.name = instructor_name
        self.address = address
    def display(self):    # Method created
      print(f"Hi i am {self.name}")

instructor_1 = Instructor("Payal", "Gurgoan")
print(instructor_1.name)
instructor_1.display()       # calling Method
instructor_2 = Instructor("Salal", "Gurgoan")
print(instructor_2.name)
instructor_3 = Instructor("Sagar", "Gurgoan")
print(instructor_3.name, instructor_3.address)
```

```
Payal
Hi i am Payal
Salal
Sagar Gurgoan
```