1)

```
print("hello")
```

```
[ ]  #2SD19EC009
     print("hello")

     hello
```

2)

```
print(float(5))
print(int(5.2))
print(int(5.9))
print(float(5))
print(complex(3.))
print(complex(0.,3.))
```

```
#2SD19EC009
print(float(5))
print(int(5.2))
print(int(5.9))
print(float(5))
print(complex(3.))
print(complex(0.,3.))

5.0
5
5
5.0
(3+0j)
3j
```

3)

```
print(2.7//2)
print(9/2)
print(8/4)
print(8//4)
print(9//2)
print(2.7//2)
print(2.7//2)
```

```
[ ]  #2SD19EC009
     print(2.7//2)
     print(9/2)
     print(8/4)
     print(8//4)
     print(9//2)
     print(2.7//2)
     print(2.7//2)

     1.0
     4.5
     2.0
     2
     4
     1.0
     1.0
```

4)

```
print(9%2)
print(4.5%3)
```

```
#2SD19EC009
print(9%2)
print(4.5%3)
```

```
1
1.5
```

5)

```
print((2+4)*3)
print(2+4*3)
(4+5j).real
(4+5j).imag
(4+5j).conjugate()
(3635151484987465685).bit_length()
```

```
#2SD19EC009
print((2+4)*3)
print(2+4*3)
(4+5j).real
(4+5j).imag
(4+5j).conjugate()
(3635151484987465685).bit_length()
```

```
18
14
62
```

6)

```
print(abs(-5.2))
print(abs(-2))
print(abs(3+4j))
print(round(-9.62))
print(round(7.5))
print(round(4.5))
print(round(7.49999))
print(round(7.63617)
```

```
#2SD19EC009
print(abs(-5.2))
print(abs(-2))
print(abs(3+4j))
print(round(-9.62))
print(round(7.5))
print(round(4.5))
print(round(7.49999))
print(round(7.636175))
```

```
5.2
2
5.0
-10
8
4
7
8
```

7)

```
import math as m
a9=2
b9=4
c9=5
s=(a9+b9+c9)/2
area=m.sqrt(s*(s-a9)*(s-b9)*(s-c9))
area
```

Area of triangle

```
[486] #2SD19EC009
      import math as m
      a9=2
      b9=4
      c9=5
      s=(a9+b9+c9)/2
      area=m.sqrt(s*(s-a9)*(s-b9)*(s-c9))
      area

      3.799671038392666
```

8)

```
a = float(input('enter any number : '))
print(type(a))
print(id(a))
```

```
#2SD19EC009
a = float(input('enter any number : '))
print(type(a))
print(id(a))
```

```
enter any number : 9
<class 'float'>
139903983974000
```

9)

```
print(7==8)
print(4>=3.142)
a9=0.01
b=90.1**2
print(a==b)
```

```
#2SD19EC009
print(7==8)
print(4>=3.142)
a9=0.01
b=90.1**2
print(a==b)
```

```
False
True
False
```

10)

```
print('6/2/4')
print(6/(2/4))
print(2**2**3)
print((2**2)**3)
print('3/4\n')
print('\n')
```

```
#2SD19EC009
print('6/2/4')
print(6/(2/4))
print(2**2**3)
print((2**2)**3)
print('3/4\n')
print('\n')
```

```
6/2/4
12.0
256
64
3/4
```

11)

```
a9=0.01
b9=9
c9=24
print('a9=',a9)
print('b9=',b9)
print('c9=',c9)
print('type of a9=',type(a),'id of a9==',id(a9))
print('type of b9=',type(b),'id of b9==',id(b9))
print('type of c9=',type(c),'id of c9==',id(c9))
```

```
#2SD19EC009
a9=0.01
b9=9
c9=24
print('a9=',a9)
print('b9=',b9)
print('c9=',c9)
print('type of a9=',type(a),'id of a9==',id(a9))
print('type of b9=',type(b),'id of b9==',id(b9))
print('type of c9=',type(c),'id of c9==',id(c9))
```

```
a9= 0.01
b9= 9
c9= 24
type of a9= <class 'float'> id of a9== 139903983976368
type of b9= <class 'float'> id of b9== 94157845650176
type of c9= <class 'float'> id of c9== 94157845650656
```

12)

```
a=0.01
b=0.1**2
c=a
print(a==c)
print(a==b)
print('a=',a,'b=',b,'c=',c)
```

```
#2SD19EC009
a=0.01
b=0.1**2
c=a
print(a==c)
print(a==b)
print('a=',a,'b=',b,'c=',c)
```

```
True
False
a= 0.01 b= 0.010000000000000002 c= 0.01
```

13)

```
print(7.0>4 and -1>0)
print(5<4 or 1!=2)
print(not 7.5<0.9 or 4==4)
print(not(7.5<0.9 or 4==4))
a=0
print(a or 4<3)
print(not a+1)
```

```
#2SD19EC009
print(7.0>4 and -1>0)
print(5<4 or 1!=2)
print(not 7.5<0.9 or 4==4)
print(not(7.5<0.9 or 4==4))
a=0
print(a or 4<3)
print(not a+1)
```

```
False
True
True
False
False
False
```

14)

```
a_9=0.01
b_9=9
c_9=24
print('a_9=',a_9)
print('b_9=',b_9)
c_9=a_9+b_9
a_9+1,b_9+1,c_9+1
```

```
#2SD19EC009
a_9=0.01
b_9=9
c_9=24
print('a_9=',a_9)
print('b_9=',b_9)
c_9=a_9+b_9
a_9+1,b_9+1,c_9+1
```

```
a_9= 0.01
b_9= 9
(1.01, 10, 10.01)
```

15)

```
print(bool(-1))
print(bool(0.0))
```

```
#2SD19EC009
print(bool(-1))
print(bool(0.0))
```

```
True
False
```

16)

```
a_9=4
print(a_9)
print('type of a_9=',type(a_9),'id of a_9==',id(a_9))
a_9=a_9+1
print('type of a_9=',type(a_9),'id of a_9==',id(a_9))
a_9=a_9+1
print('type of a_9=',type(a_9),'id of a==',id(a_9))
```

```
#2SD19EC009
a_9=4
print(a_9)
print('type of a_9=',type(a_9),'id of a_9==',id(a_9))
a_9=a_9+1
print('type of a_9=',type(a_9),'id of a_9==',id(a_9))
a_9=a_9+1
print('type of a_9=',type(a_9),'id of a==',id(a_9))
```

```
4
type of a_9= <class 'int'> id of a_9== 94157845650016
type of a_9= <class 'int'> id of a_9== 94157845650048
type of a_9= <class 'int'> id of a== 94157845650080
```

17)

```
a9='red'
print(id(a9))
b9='black'
print(id(b9))
colour=['red','black','green']
print(colour)
print(id(colour))
```

```
] #2SD19EC009
  a9='red'
  print(id(a9))
  b9='black'
  print(id(b9))
  colour=['red','black','green']
  print(colour)
  print(id(colour))
```

```
  139904680268848
  139904256275248
  ['red', 'black', 'green']
  139903994814944
```

18)

```
s9='SDM'
d9='CET'
m9=s9+d9
print(m9)
```

```
#2SD19EC009
s9='SDM'
d9='CET'
m9=s9+d9
print(m9)
```

```
SDMCET
```

19)

```
print('a'*4)
print('-a-'*5)
```

```
#2SD19EC009
print('a'*4)
print('-a-'*5)
```

```
aaaa
-a--a--a--a--a-
```

20)

```
print(str(42))
print(str(3.4e5))
print(str(3.4e20))
print(str(42))
print(str(3.4e4))
print(str(3.4e20))
print(str(3.4e9))
print(str(3.4e10))
print(str(3.4e11))
print(str(3.4e16))
#till 15 zeros can continue from 16 it won't
```

```
#2SD19EC009
print(str(42))
print(str(3.4e5))
print(str(3.4e20))
print(str(42))
print(str(3.4e4))
print(str(3.4e20))
print(str(3.4e9))
print(str(3.4e10))
print(str(3.4e11))
print(str(3.4e16))
#till 15 zeros can continue from 16 it won't
```

```
42
340000.0
3.4e+20
42
34000.0
3.4e+20
3400000000.0
34000000000.0
340000000000.0
3.4e+16
```

21)

```
print(('a'*4+'b')*3)
print(('shree'*2+'SDM'+'CET ')*3)
```

```
#2SD19EC009
print(('a'*4+'b')*3)
print(('shree'*2+'SDM'+'CET ')*3)
```

```
aaaabaaaabaaaab
shreeshreeSDMCET shreeshreeSDMCET shreeshreeSDMCET
```

22)

```
c9='quoth the Reven "never more"'
print(c9)
d9='he said, \"hi.\"'
print(d9)
```

```
#2SD19EC009
c9='quoth the Reven "never more"'
print(c9)
d9='he said, \"hi.\"'
print(d9)
```

```
quoth the Reven "never more"
he said, "hi."
```

23)

```
#result=str(10)
#print(result)
#result=result+1
#print(result)


#can only concatenate str (not "int") to str
```

24)

```
s9='hi\nhello'
print(s9)
raw_s9 = r'hi\nhello'
print(raw_s9)
```

```
#2SD19EC009
#result=str(10)
#print(result)
#result=result+1
#print(result)


#can only concatenate str (not "int") to str
```

```
#2SD19EC009
s9='hi\nhello'
print(s9)
raw_s9 = r'hi\nhello'
print(raw_s9)
```

```
hi
hello
hi\nhello
```

25)

```
s9="sdm"
d9="""cet
dharwad ka"""
q9=s9+d9
print(q9)
```

```
#2SD19EC009
s9="sdm"
d9="""cet
dharwad ka"""
q9=s9+d9
print(q9)
```

```
sdmcet
dharwad ka
```

26)

```
a9="""1
2
3"""
print(a9)
```

```
#2SD19EC009
a9="""1
2
3"""
print(a9)
```

```
1
2
3
```

27)

```
p9='abcdefghi'
print(p9[6])
print(p9[5])
print(p9[-1])
#IndexError: string index out of range {if selected out of range}
```

indexing and slicing

```
507] #2SD19EC009
     p9='abcdefghi'
     print(p9[6])
     print(p9[5])
     print(p9[-1])
     #IndexError: string index out of range {if selected out of range}
```

```
g
f
i
```

28)

```
a9='SDM CET Dharwad'
b9=a9[1:3]
c9=a9[:3]
d9=a9[3:]
e9=a9[:]
print(a9)
print(b9)
print(c9)
print(d9)
print(e9)
f9=c9 in a9
print(f9)
g9=e9 in c9
print(g9)
```

```
#2SD19EC009
a9='SDM CET Dharwad'
b9=a9[1:3]
c9=a9[:3]
d9=a9[3:]
e9=a9[:]
print(a9)
print(b9)
print(c9)
print(d9)
print(e9)
f9=c9 in a9
print(f9)
g9=e9 in c9
print(g9)
```

```
SDM CET Dharwad
DM
SDM
 CET Dharwad
SDM CET Dharwad
True
False
```

29)

```
a9='SDMCETDharwadDhavalagiri'
b9=a9[1:8]
c9=a9[8:25]
d9=a9[1:25:2]
e9=a9[-1:1:-1]
print(a9)
print(b9)
print(c9)
print(d9)
print(e9)
```

```
#2SD19EC009
a9='SDMCETDharwadDhavalagiri'
b9=a9[1:8]
c9=a9[8:25]
d9=a9[1:25:2]
e9=a9[-1:1:-1]
print(a9)
print(b9)
print(c9)
print(d9)
print(e9)
```

```
SDMCETDharwadDhavalagiri
DMCETDh
arwadDhavalagiri
DCThraDaaaii
irigalavahDdawrahDTECM
```

30)

```
a9='SDMCETDharwadDhavalagiri'
b9=a9[1:25:1]
c9=a9[1:25:2]
d9=a9[1:25:3]
print(b9)
print(c9)
print(d9)
```

```
#2SD19EC009
a9='SDMCETDharwadDhavalagiri'
b9=a9[1:25:1]
c9=a9[1:25:2]
d9=a9[1:25:3]
print(b9)
print(c9)
print(d9)
```

```
DMCETDharwadDhavalagiri
DCThraDaaaii
DEhwDvar
```

31)

```
a9='SDMCETDharwadDhavalagiri'
b9=a9[-24:-1:1]
c9=a9[-24:-1:2]
d9=a9[-24:-1:3]
print(b9)
print(c9)
print(d9)
```

```
#2SD19EC009
a9='SDMCETDharwadDhavalagiri'
b9=a9[-24:-1:1]
c9=a9[-24:-1:2]
d9=a9[-24:-1:3]
print(b9)
print(c9)
print(d9)
```

```
SDMCETDharwadDhavalagir
SMEDawdhvlgr
SCDrdali
```

32)

```
a9='SDMCETDharwadDhavalagiri'
a9=a9+' karnataka'
print(a9)
```

33)

```
a9='SDMCETDharwadDhavalagiri'
b9='karnataka india'
c9=a9+b9[:9]
d9=a9+b9[9:]
e9=len(c9)
print(c9)
print(d9)
print(e9)
```

```
#2SD19EC009
a9='SDMCETDharwadDhavalagiri'
a9=a9+' karnataka'
print(a9)
```

```
SDMCETDharwadDhavalagiri karnataka
```

```
#2SD19EC009
a9='SDMCETDharwadDhavalagiri'
b9='karnataka india'
c9=a9+b9[:9]
d9=a9+b9[9:]
e9=len(c9)
print(c9)
print(d9)
print(e9)
```

```
SDMCETDharwadDhavalagirikarnataka
SDMCETDharwadDhavalagiri india
33
```

34)

```
s9='i am going to BANGLORE tomorrow'
print('s9=',s9)
p2=(s9.lower())
print(p2)
p3=(p2.replace('banglore','mysore').strip('tomowwor'))
print('p3=',p3)
```

```
#2SD19EC009
s9='i am going to BANGLORE tomorrow'
print('s9=',s9)
p2=(s9.lower())
print(p2)
p3=(p2.replace('banglore','mysore').strip('tomowwor'))
print('p3=',p3)
```

```
s9= i am going to BANGLORE tomorrow
i am going to banglore tomorrow
p3= i am going to mysore
```

35)

```
txt9="my name is {0},I'm {1}".format('john','36')
print(txt9)

txt2="my name is {hname},I'm {age}".format(hname='john',age='36')
print(txt2)

txt3="my name is {},I'm {}".format('john','36')
print(txt3)
```

```
#2SD19EC009
txt9="my name is {0},I'm {1}".format('john','36')
print(txt9)

txt2="my name is {hname},I'm {age}".format(hname='john',age='36')
print(txt2)

txt3="my name is {},I'm {}".format('john','36')
print(txt3)
```

```
my name is john,I'm 36
my name is john,I'm 36
my name is john,I'm 36
```

36)

```
print("====={0:12}====".format('python'))
print("====={0:15}====".format('python'))
print("====={0:20}====".format('python'))
print("====={0:8}====".format('python'))
print("====={0:^20}====".format('python'))
print("====={0:>20}====".format('python'))
print("====={0:<20}====".format('python'))
```

37)

```
print("=={0:^20} ==".format('SDMCET'))
print("=={0:+^20} ==".format('SDMCET'))
print("=={0:-^20} ==".format('SDMCET'))
print("=={0:*^20} ==".format('SDMCET'))
print("=={0:^20} ==".format('SDMCET'))
print("=={0:^20} ==".format('SDMCET'))
```

```
#2SD19EC009
print("====={0:12}====".format('python'))
print("====={0:15}====".format('python'))
print("====={0:20}====".format('python'))
print("====={0:8}====".format('python'))
print("====={0:^20}====".format('python'))
print("====={0:>20}====".format('python'))
print("====={0:<20}====".format('python'))
```

```
====python      ====
====python         ====
====python             ====
====python  ====
====       python       ====
====              python====
====python             ====
```

```
#2SD19EC009
print("=={0:^20} ==".format('SDMCET'))
print("=={0:+^20} ==".format('SDMCET'))
print("=={0:-^20} ==".format('SDMCET'))
print("=={0:*^20} ==".format('SDMCET'))
print("=={0:^20} ==".format('SDMCET'))
print("=={0:^20} ==".format('SDMCET'))
```

```
==       SDMCET        ==
==+++++++SDMCET+++++++ ==
==-------SDMCET------- ==
==*******SDMCET******* ==
==       SDMCET        ==
==       SDMCET        ==
```

38)

```
print("{} plus {} equals {}".format(2,3,5))
print("{0} plus {1} equals {2}".format(2,3,+5))
print("{num1} plus {num2} equals {answer}".format(num1=2,num2=3,answer=
5))
```

```
#2SD19EC009
print("{} plus {} equals {}".format(2,3,5))
print("{0} plus {1} equals {2}".format(2,3,+5))
print("{num1} plus {num2} equals {answer}".format(num1=2,num2=3,answer=5))
```

```
2 plus 3 equals 5
2 plus 3 equals 5
2 plus 3 equals 5
```

39)

```
heading ='|Indexing of Dutch Tupil Prices|'
line ='+'+'-'*16+'-'*13+'+'
print(line,heading,line,
      '|Nov 23 1636 | 100|' ,
      '|Nov 25 1636 | 673|',
      '|Nov 25 1636 | 673|', line,sep='\n')
```

```
#2SD19EC009
heading ='|Indexing of Dutch Tupil Prices|'
line ='+'+'-'*16+'-'*13+'+'
print(line,heading,line,
      '|Nov 23 1636 | 100|' ,
      '|Nov 25 1636 | 673|',
      '|Nov 25 1636 | 673|', line,sep='\n')
```

```
+----------------------------+
|Indexing of Dutch Tupil Prices|
+----------------------------+
|Nov 23 1636 | 100|
|Nov 25 1636 | 673|
|Nov 25 1636 | 673|
+----------------------------+
```

```
40)
a9=254
print('a9={0:5d}'.format(a9))
print('a9={0:10d}'.format(a9))
print('a9={0:5o}'.format(a9))
print('a9={0:5x}'.format(a9))
print('a9={0:5X}'.format(a9))
print('a9={0:10d}'.format(a9))
print('a9={0:10b}'.format(a9))
print('a9={0:10o}'.format(a9))
print('a9={0:10x}'.format(a9))
print('a9={0:10X}'.format(a9))
```

```
#2SD19EC009
a9=254
print('a9={0:5d}'.format(a9))
print('a9={0:10d}'.format(a9))
print('a9={0:5o}'.format(a9))
print('a9={0:5x}'.format(a9))
print('a9={0:5X}'.format(a9))
print('a9={0:10d}'.format(a9))
print('a9={0:10b}'.format(a9))
print('a9={0:10o}'.format(a9))
print('a9={0:10x}'.format(a9))
print('a9={0:10X}'.format(a9))
```

```
a9=  254
a9=       254
a9=  376
a9=   fe
a9=   FE
a9=       254
a9=  11111110
a9=       376
a9=        fe
a9=        FE
```

```
41)
a9=254
print('a9={0:05d}'.format(a9))
print('a9={0:010d}'.format(a9))
print('a9={0:05o}'.format(a9))
print('a9={0:05x}'.format(a9))
print('a9={0:05X}'.format(a9))
print('a9={0:010d}'.format(a9))
print('a9={0:010b}'.format(a9))
print('a9={0:010o}'.format(a9))
print('a9={0:010x}'.format(a9))
print('a9={0:010X}'.format(a9))
```

```
#2SD19EC009
a9=254
print('a9={0:05d}'.format(a9))
print('a9={0:010d}'.format(a9))
print('a9={0:05o}'.format(a9))
print('a9={0:05x}'.format(a9))
print('a9={0:05X}'.format(a9))
print('a9={0:010d}'.format(a9))
print('a9={0:010b}'.format(a9))
print('a9={0:010o}'.format(a9))
print('a9={0:010x}'.format(a9))
print('a9={0:010X}'.format(a9))
```

```
a9=00254
a9=0000000254
a9=00376
a9=000fe
a9=000FE
a9=0000000254
a9=0011111110
a9=0000000376
a9=00000000fe
a9=00000000FE
```

42)

```
a9= -52
b9= 12
s9 = '{0:5d}\n{1:5d}\n= {2:3d}'.format(a9,b9,a9+b9)
print(s+9)
s9 = '{0:+5d}\n{1:+5d}\n= {2:+3d}'.format(a9,b9,a9+b9)
print(s9)
a9 =25
s9= '{0:+5d}\n{1:+5d}\n= {2:+3d}'.format(a9,b9,a9+b9)
print(s9)
```

```
#2SD19EC009
a9= -52
b9= 12
s9 = '{0:5d}\n{1:5d}\n= {2:3d}'.format(a9,b9,a9+b9)
print(s+9)
s9 = '{0:+5d}\n{1:+5d}\n= {2:+3d}'.format(a9,b9,a9+b9)
print(s9)
a9 =25
s9= '{0:+5d}\n{1:+5d}\n= {2:+3d}'.format(a9,b9,a9+b9)
print(s9)
```

```
14.5
  -52
  +12
= -40
  +25
  +12
= +37
```

43)

```
print('study', 'tonight', sep ='*')
print('study', 'tonight', sep ='&')
print('study', 'tonight', sep ='_',end='STUDENTS')
```

```
#2SD19EC009
print('study', 'tonight', sep ='*')
print('study', 'tonight', sep ='&')
print('study', 'tonight', sep ='_',end='STUDENTS')
```

```
study*tonight
study&tonight
study_tonightSTUDENTS
```

44)

```
a9=1.46e-10
print(a9)
print('{0:g}'.format(a9))
print('{0:10.2E}'.format(a9))
print('{0:12.2E}'.format(a9))
print('{0:12.3E}'.format(a9))
print('{0:15.13f}'.format(a9))
print('{0:15.12f}'.format(a9))
```

```
#2SD19EC009
a9=1.46e-10
print(a9)
print('{0:g}'.format(a9))
print('{0:10.2E}'.format(a9))
print('{0:12.2E}'.format(a9))
print('{0:12.3E}'.format(a9))
print('{0:15.13f}'.format(a9))
print('{0:15.12f}'.format(a9))
```

```
1.46e-10
1.46e-10
  1.46E-10
    1.46E-10
   1.460E-10
0.0000000001460
 0.000000000146
```

45)

```
list9=[1,'two',3.14,0]
list09=[2,'a',-0.1,list9,True]
print(list9)
print(list09)
print(list09[-1])
print(list09[3][1])
print(list09[3][1][1])
```

```
#2SD19EC009
list9=[1,'two',3.14,0]
list09=[2,'a',-0.1,list9,True]
print(list9)
print(list09)
print(list09[-1])
print(list09[3][1])
print(list09[3][1][1])
```

```
[1, 'two', 3.14, 0]
[2, 'a', -0.1, [1, 'two', 3.14, 0], True]
True
two
w
```

46)

```
list9=['physics','chem',1997,2000]
list_9=[1,2,3,4,5,6,7]
print('list9[0]: ',list9[0])
print('list_9[1:5]: ',list_9[1:5])
list_09 =['physics','chem',1997,2000]
print('value available at index 2 : ',list_09[2])
list_09[2]=2001
print('new value available at index 2 : ',list_09[2])
```

LIST

```
#2SD19EC009
list9=['physics','chem',1997,2000]
list_9=[1,2,3,4,5,6,7]
print('list9[0]: ',list9[0])
print('list_9[1:5]: ',list_9[1:5])
list_09 =['physics','chem',1997,2000]
print('value available at index 2 : ',list_09[2])
list_09[2]=2001
print('new value available at index 2 : ',list_09[2])
```

```
list9[0]:  physics
list_9[1:5]:  [2, 3, 4, 5]
value available at index 2 :  1997
new value available at index 2 :  2001
```

47)

```
list9=[1,'two',3.14,0]
list09=[2,'a',-0.1,list9,True]
print(list9)
print(list09)
del list9[2]
print('after del list9[2]= ',list9)
del list09[1]
print('after del list09[1]= ',list09)
```

```
#2SD19EC009
list9=[1,'two',3.14,0]
list09=[2,'a',-0.1,list9,True]
print(list9)
print(list09)
del list9[2]
print('after del list9[2]= ',list9)
del list09[1]
print('after del list09[1]= ',list09)
```

```
[1, 'two', 3.14, 0]
[2, 'a', -0.1, [1, 'two', 3.14, 0], True]
after del list9[2]=  [1, 'two', 0]
after del list09[1]=  [2, -0.1, [1, 'two', 0], True]
```

48)

```
a9=1
print('a9 =',a9,"adress of a9= ",id(a9))
a9=a9+1
print('a9=',a9,"adress of a9= ",id(a9))
a9=a9+1
print('a9=',a9,"adress of a9= ",id(a9))
```

```
#2SD19EC009
a9=1
print('a9 =',a9,"adress of a9= ",id(a9))
a9=a9+1
print('a9=',a9,"adress of a9= ",id(a9))
a9=a9+1
print('a9=',a9,"adress of a9= ",id(a9))
```

```
a9 = 1 adress of a9=  94157845649920
a9= 2 adress of a9=  94157845649952
a9= 3 adress of a9=  94157845649984
```

49)

```
list9=[1,'two',3.14,0]
list09=[2,'a',-0.1,list9,True]
print('adress of list9= @ ',id(list9))
print('adress of list09= @ ',id(list09))
del list9[2]
del list09[1]
print('adress of list9= @ ',id(list9))
print('adress of list09= @ ',id(list09))
#its mutable so  adress won't change even after del a number
```

```
#2SD19EC009
list9=[1,'two',3.14,0]
list09=[2,'a',-0.1,list9,True]
print('adress of list9= @ ',id(list9))
print('adress of list09= @ ',id(list09))
del list9[2]
del list09[1]
print('adress of list9= @ ',id(list9))
print('adress of list09= @ ',id(list09))
#its mutable so  adress won't change even after del a number
```

```
adress of list9= @  139903987754400
adress of list09= @  139903981423104
adress of list9= @  139903987754400
adress of list09= @  139903981423104
```

50)

```
q9 =[1,2,3]
q09 =q9
print(q9)
print(q09)
q9[2]='oops'
print(q9)
print(q09)
```

```
#2SD19EC009
q9 =[1,2,3]
q09 =q9
print(q9)
print(q09)
q9[2]='oops'
print(q9)
print(q09)
```

```
[1, 2, 3]
[1, 2, 3]
[1, 2, 'oops']
[1, 2, 'oops']
```

51)

```
list9=[1,2,3,4,5]
list09=list9
print('list9= ',list9)
print('list09= ',list09)
list9[2]='SDM'
print('list9= ',list9)
print('list09= ',list09)
print('adress of list9 = @ ',id(list9))
print('adress of list09= @ ',id(list09))
```

```
#2SD19EC009
list9=[1,2,3,4,5]
list09=list9
print('list9= ',list9)
print('list09= ',list09)
list9[2]='SDM'
print('list9= ',list9)
print('list09= ',list09)
print('adress of list9 = @ ',id(list9))
print('adress of list09= @ ',id(list09))
```

```
list9=  [1, 2, 3, 4, 5]
list09=  [1, 2, 3, 4, 5]
list9=  [1, 2, 'SDM', 4, 5]
list09=  [1, 2, 'SDM', 4, 5]
adress of list9 = @  139903987648944
adress of list09= @  139903987648944
```

```
52)
a9=12345
b9=a9
print("values os a9=',a9,' and adress of a9= ",id(a9))
print("values os a9=',b9,' and adress of a9*= ",id(b9))
b9=b9+1
print('after incrementing b9')
print("values os a9=',a9,' and adress of a9= ",id(a9))
print("values os a9=',b9,' and adress of a9= ",id(b9))
list9=[1,2,3,4,5]
list09=list9
print('list9= ',list9)
print('list09= ',list09)
print("values os a9=',a9,' and adress of a9= ",id(a9))
print("values os a9=',b9,' and adress of a9= ",id(b9))
```

```
] #2SD19EC009
  a9=12345
  b9=a9
  print("values os a9=',a9,' and adress of a9= ",id(a9))
  print("values os a9=',b9,' and adress of a9*= ",id(b9))
  b9=b9+1
  print('after incrementing b9')
  print("values os a9=',a9,' and adress of a9= ",id(a9))
  print("values os a9=',b9,' and adress of a9= ",id(b9))
  list9=[1,2,3,4,5]
  list09=list9
  print('list9= ',list9)
  print('list09= ',list09)
  print("values os a9=',a9,' and adress of a9= ",id(a9))
  print("values os a9=',b9,' and adress of a9= ",id(b9))
```

```
values os a9=',a9,' and adress of a9=  139903981983696
values os a9=',b9,' and adress of a9*=  139903981983696
after incrementing b9
values os a9=',a9,' and adress of a9=  139903981983696
values os a9=',b9,' and adress of a9=  139903981983248
list9=  [1, 2, 3, 4, 5]
list09=  [1, 2, 3, 4, 5]
values os a9=',a9,' and adress of a9=  139903981983696
values os a9=',b9,' and adress of a9=  139903981983248
```

53)

```
a9=3
q9=[1,2,a]
print('adress of a9= ',id(a9))
print('adress of q9[2]',id(q9[2]))
a9=4
print(q9)
print('adress of a9= ',id(a9))
print('adress of q9[2]',id(q9[2]))
```

```
#2SD19EC009
a9=3
q9=[1,2,a]
print('adress of a9= ',id(a9))
print('adress of q9[2]',id(q9[2]))
a9=4
print(q9)
print('adress of a9= ',id(a9))
print('adress of q9[2]',id(q9[2]))
```

```
adress of a9=  94157845649984
adress of q9[2] 94157845649888
[1, 2, 0]
adress of a9=  94157845650016
adress of q9[2] 94157845649888
```

54)

```
a9=3
q9=[1,2,a9]
print('adress of first q9= ',id(q9))
print('adress of first q9[0]= ',id(q9[0]))
print('adress of first q9[1]= ',id(q9[1]))
print('adress of first q9[2]= ',id(q9[2]))
q9[0]='good'
print('adress of first q9[0]= ',id(q9[0]))
print('adress of first q9= ',id(q9))
```

```
] #2SD19EC009
  a9=3
  q9=[1,2,a9]
  print('adress of first q9= ',id(q9))
  print('adress of first q9[0]= ',id(q9[0]))
  print('adress of first q9[1]= ',id(q9[1]))
  print('adress of first q9[2]= ',id(q9[2]))
  q9[0]='good'
  print('adress of first q9[0]= ',id(q9[0]))
  print('adress of first q9= ',id(q9))
```

```
adress of first q9=  139903983942384
adress of first q9[0]=  94157845649920
adress of first q9[1]=  94157845649952
adress of first q9[2]=  94157845649984
adress of first q9[0]=  139904299654128
adress of first q9=  139903983942384
```

55)

```
a9=3
q9=[1,2,a9]
print('adress of first q9= ',id(q9))
print('adress of id(q9[2]= ',id(q9[2]))
a9=4
print('before change of a9 ',q9)
print('adress of first a9= ',id(a9))
print('adress of id(q9[2]= ',id(q9[2]))
```

```
#2SD19EC009
a9=3
q9=[1,2,a9]
print('adress of first q9= ',id(q9))
print('adress of id(q9[2]= ',id(q9[2]))
a9=4
print('before change of a9 ',q9)
print('adress of first a9= ',id(a9))
print('adress of id(q9[2]= ',id(q9[2]))
```

```
adress of first q9=  139903982025536
adress of id(q9[2]=  94157845649984
before change of a9  [1, 2, 3]
adress of first a9=  94157845650016
adress of id(q9[2]=  94157845649984
```

56)

```
q9=[0,1,2,3,4,5,6]
print("adress of first q9= ",id(q9))
print(q9[1::4])
print("adress of first q9= ",id(q9))
print(q9[::-1])
print("adress of first q9= ",id(q9))
print(q9[1::2])
print("adress of first q9= ",id(q9))
```

```
#2SD19EC009
q9=[0,1,2,3,4,5,6]
print("adress of first q9= ",id(q9))
print(q9[1::4])
print("adress of first q9= ",id(q9))
print(q9[::-1])
print("adress of first q9= ",id(q9))
print(q9[1::2])
print("adress of first q9= ",id(q9))
```

```
adress of first q9=  139903981773920
[1, 5]
adress of first q9=  139903981773920
[6, 5, 4, 3, 2, 1, 0]
adress of first q9=  139903981773920
[1, 3, 5]
adress of first q9=  139903981773920
```

57)

```
a9=3
q9=[1,2,a,'good','SDM']
q09=q1
q009=q1[1:3]
q0009=q1[0:4]
print("adress of first q9= ",id(q9))
print("adress of first q09= ",id(q09))
print("adress of first q009= ",id(q009))
print("adress of first q0009= ",id(q0009))
```

```
#2SD19EC009
a9=3
q9=[1,2,a,'good','SDM']
q09=q1
q009=q1[1:3]
q0009=q1[0:4]
print("adress of first q9= ",id(q9))
print("adress of first q09= ",id(q09))
print("adress of first q009= ",id(q009))
print("adress of first q0009= ",id(q0009))
```

```
adress of first q9=  139903981840176
adress of first q09=  139904065452544
adress of first q009=  139903981773920
adress of first q0009=  139903995660880
```

58)

```
a9=[]
a9.append(4)
print(a9)
a9.extend([5,6,8])
print(a9)
a9.insert(1, 88)
print(a)
a9.remove(8)
print(a9)
#..........for index==>.index we shall get the value of the index valu
e..............
```

```
#2SD19EC009
a9=[]
a9.append(4)
print(a9)
a9.extend([5,6,8])
print(a9)
a9.insert(1, 88)
print(a)
a9.remove(8)
print(a9)
#..........for index==>.index we shall get the value of the index value.
a9=a9.index(5)
print(a9)
```

```
[4]
[4, 5, 6, 8]
0
[4, 88, 5, 6]
2
```

```
59)
a9=3
q9=[1,2,a,'good','SDM']
q09=q9
q009=q9[1:3]
q0009=q9[0:4]
print("adress of first q9= ",id(q9))
print("adress of first q09= ",id(q09))
print("adress of first q009 ",id(q009))
print("adress of first q0009= ",id(q0009))
q9[2]='change'
q09[1]='MLA'
q0009[2]='mp'
print('q9 =',q9)
print('q09 =',q09)
print('q009=',q009)
print('q0009 =',q0009)
```

```
] #2SD19EC009
  a9=3
  q9=[1,2,a,'good','SDM']
  q09=q9
  q009=q9[1:3]
  q0009=q9[0:4]
  print("adress of first q9= ",id(q9))
  print("adress of first q09= ",id(q09))
  print("adress of first q009 ",id(q009))
  print("adress of first q0009= ",id(q0009))
  q9[2]='change'
  q09[1]='MLA'
  q0009[2]='mp'
  print('q9 =',q9)
  print('q09 =',q09)
  print('q009=',q009)
  print('q0009 =',q0009)
```

```
adress of first q9=  139903981769904
adress of first q09=  139903981769904
adress of first q009  139903981840176
adress of first q0009=  139903984036992
q9 = [1, 'MLA', 'change', 'good', 'SDM']
q09 = [1, 'MLA', 'change', 'good', 'SDM']
q009= [2, 0]
q0009 = [1, 2, 'mp', 'good']
```

60)

```
a9=[11,3,5,8,9]
print(a9)
a9.sort()
print('sorted a9',a9)
a9.reverse()
print('reverse a9',a9)
```

```
#2SD19EC009
a9=[11,3,5,8,9]
print(a9)
a9.sort()
print('sorted a9',a9)
a9.reverse()
print('reverse a9',a9)
```

```
[11, 3, 5, 8, 9]
sorted a9 [3, 5, 8, 9, 11]
reverse a9 [11, 9, 8, 5, 3]
```

61)

```
num9=[6,9,3,1]
print('numbers:',num9)
num_sorted9= sorted (num9)
print('sorted numbers:',num_sorted9)
print('number address:',id(num9))
print('sorted address:',id(num_sorted9))
num9[2]=258
print('numbers=',num9)
print('sorted numbers:',num_sorted9)
print('sorted address:',id(num sorted9))
```

```
#2SD19EC009
num9=[6,9,3,1]
print('numbers:',num9)
num_sorted9= sorted (num9)
print('sorted numbers:',num_sorted9)
print('number address:',id(num9))
print('sorted address:',id(num_sorted9))
num9[2]=258
print('numbers=',num9)
print('sorted numbers:',num_sorted9)
print('sorted address:',id(num_sorted9))
```

```
numbers: [6, 9, 3, 1]
sorted numbers: [1, 3, 6, 9]
number address: 139903981846768
sorted address: 139903981849168
numbers= [6, 9, 258, 1]
sorted numbers: [1, 3, 6, 9]
sorted address: 139903981849168
```

62)

```
a9=[16,8,9,545]
print('a9',a9)
a9.sort()
print(a9)
a09=['a','E','e','A','i','I','o','O']#.....first uppercase,and then low
er case.......
a09.sort()
print(a09)
```

```
#2SD19EC009
a9=[16,8,9,545]
print('a9',a9)
a9.sort()
print(a9)
a09=['a','E','e','A','i','I','o','O']#.....first uppercase,and then lower case.......
a09.sort()
print(a09)
```

```
a9 [16, 8, 9, 545]
[8, 9, 16, 545]
['A', 'E', 'I', 'O', 'a', 'e', 'i', 'o']
```

63)

```
stack=[]
stack.append(1)
stack.append(2)
stack.append(3)
stack.append(5)
print(stack)
print(stack.pop())
print(stack.pop()) #....last element will be removed....
print(stack)
```

```
#2SD19EC009
stack=[]
stack.append(1)
stack.append(2)
stack.append(3)
stack.append(5)
print(stack)
print(stack.pop())
print(stack.pop()) #....last element will be removed...
print(stack)
```

```
[1, 2, 3, 5]
5
3
[1, 2]
```

64)

```
s9='jan feb mar apr may jun'
print(s9)
p9=s9.split()
print(p9)

s09="j .m. brown and menchen and r. p. van't rooden"
print(s09)
p09=s09.split('and')
print(p09)
```

```
#2SD19EC009
s9='jan feb mar apr may jun'
print(s9)
p9=s9.split()
print(p9)

s09="j .m. brown and menchen and r. p. van't rooden"
print(s09)
p09=s09.split('and')
print(p09)
```

```
jan feb mar apr may jun
['jan', 'feb', 'mar', 'apr', 'may', 'jun']
j .m. brown and menchen and r. p. van't rooden
['j .m. brown ', ' menchen ', " r. p. van't rooden"]
```

65)

```
#l1=[6,3,9,1]
#t1=(6,3,9,1)
#print('l1: ',l1)
#print('t1: ',t1)
#l1[2]=88
#print('l1: ',l1)
#t1[2]=88
#print('t1: ',t1)
#ANSWER: TypeError: 'tuple' object does not support item assignment
```

```
5] #2SD19EC009
   #l1=[6,3,9,1]
   #t1=(6,3,9,1)
   #print('l1: ',l1)
   #print('t1: ',t1)
   #l1[2]=88
   #print('l1: ',l1)
   #t1[2]=88
   #print('t1: ',t1)
   #ANSWER: TypeError: 'tuple' object does not support item assignment
```

```
66)
t9=(1,['a','b','c'],0)
print(t9)
print('t9 address: ',id(t9))
print('address of first t9: ',id(t9))
print('add of t9[0]',id(t9[0]))
print('add of t9[1][2]',id(t9[1][2]))
t9[1][2]='wtf'
print(t9)
print('t9 address: ',id(t9))
print('address of sec t9: ',id(t9))
print('add of t9[0]',id(t9[0]))
print('add of t9[1][2]',id(t9[1][2]))
```

```
#2SD19EC009
t9=(1,['a','b','c'],0)
print(t9)
print('t9 address: ',id(t9))
print('address of first t9: ',id(t9))
print('add of t9[0]',id(t9[0]))
print('add of t9[1][2]',id(t9[1][2]))
t9[1][2]='wtf'
print(t9)
print('t9 address: ',id(t9))
print('address of sec t9: ',id(t9))
print('add of t9[0]',id(t9[0]))
print('add of t9[1][2]',id(t9[1][2]))
```

```
(1, ['a', 'b', 'c'], 0)
t9 address:  139903996032816
address of first t9:  139903996032816
add of t9[0] 94157845649920
add of t9[1][2] 139904709501296
(1, ['a', 'b', 'wtf'], 0)
t9 address:  139903996032816
address of sec t9:  139903996032816
add of t9[0] 94157845649920
add of t9[1][2] 139904065498800
```

67)tup9=('phy','chem',1997,2000)
tup09=(1,2,3,4,5)
tup009='a','b','c','d','e'
tup0009=tup9+tup09
tup_9=tup4+tup3
print(tup9,'\n',tup09,'\n',tup009)
print(tup0009)
print(tup_9)

```
tup9=('phy','chem',1997,2000)
tup09=(1,2,3,4,5)
tup009='a','b','c','d','e'
tup0009=tup9+tup09
tup_9=tup4+tup3
print(tup9,'\n',tup09,'\n',tup009)
print(tup0009)
print(tup_9)
```

```
('phy', 'chem', 1997, 2000)
 (1, 2, 3, 4, 5)
 ('a', 'b', 'c', 'd', 'e')
('phy', 'chem', 1997, 2000, 1, 2, 3, 4, 5)
('phy', 'chem', 1997, 2000, 1, 2, 3, 4, 5, 'a', 'b', 'c', 'd', 'e')
```

68)

```
fruits9=['apple','banana','cherry']
for x9 in fruits9:
  print(x9)
```

```
#2SD19EC009
fruits9=['apple','banana','cherry']
for x9 in fruits9:
  print(x9)
```

```
apple
banana
cherry
```

69)

```
fruits9=['apple','banana','cherry','orange']
for x9 in fruits9:
  for letter in x9:
    print(letter ,end='.')
  print()
```

```
#2SD19EC009
fruits9=['apple','banana','cherry','orange']
for x9 in fruits9:
  for letter in x9:
    print(letter ,end='.')
  print()
```

```
a.p.p.l.e.
b.a.n.a.n.a.
c.h.e.r.r.y.
o.r.a.n.g.e.
```

70)

```
for x9 in range(2,6):
  print(x9)
for x_9 in range(2,38,3):
  print(x_9)
else:
  print('finally finished')
```

```
#2SD19EC009
for x9 in range(2,6):
  print(x9)
for x_9 in range(2,38,3):
  print(x_9)
else:
  print('finally finished')
```

```
2
3
4
5
2
5
8
11
14
17
20
23
26
29
32
35
finally finished
```

71)

```
a9=5
fib=[1,1]
for i in range(2,a9+1):
  fib.append(fib[i-1]+fib[i-2])
print(fib)
```

```
] #2SD19EC009
  a9=5
  fib=[1,1]
  for i in range(2,a9+1):
    fib.append(fib[i-1]+fib[i-2])
  print(fib)
```

```
[1, 1, 2, 3, 5, 8]
```

72)

```
#zip
name9 = [ "Manjeet", "Nikhil", "Shambhavi", "Astha" ]
roll_no = [ 4, 1, 3, 2 ]

# using zip() to map values
mapped = zip(name9, roll_no)

print(set(mapped))
```

```
#2SD19EC009
#zip
name9 = [ "Manjeet", "Nikhil", "Shambhavi", "Astha" ]
roll_no = [ 4, 1, 3, 2 ]

# using zip() to map values
mapped = zip(name9, roll_no)

print(set(mapped))
```

```
{('Nikhil', 1), ('Manjeet', 4), ('Shambhavi', 3), ('Astha', 2)}
```

73)

```
a9=33
b9=200
if b>a:
  print('b is greater than a')
else:
    print('b is smaller that a')
```

```
#2SD19EC009
a9=33
b9=200
if b>a:
  print('b is greater than a')
else:
    print('b is smaller that a')
```

```
b is greater than a
```

74)

```
for x9 in range(10):
  if x9<=3:
   print(x9,'is less than or equal to three')
  elif x9>5:
    print(x9,'is greater than five')
else:
    print(x9,'must be four or five')
```

```
5] #2SD19EC009
  for x9 in range(10):
    if x9<=3:
     print(x9,'is less than or equal to three')
    elif x9>5:
      print(x9,'is greater than five')
  else:
      print(x9,'must be four or five')
```

```
  0 is less than or equal to three
  1 is less than or equal to three
  2 is less than or equal to three
  3 is less than or equal to three
  6 is greater than five
  7 is greater than five
  8 is greater than five
  9 is greater than five
  9 must be four or five
```

75)

```
#leap year
year9=2020
if not year9%400:
  is_leap_year=True
elif not year9%100:
  is_leap_year9=False
elif not year9%4:
  is_leap_year9=True
else:
  is_leap_year9=False
s_ly ='is a' if is_leap_year9 else 'is not a'
print('{:4d} {:s} leap_year9'.format(year9, s_ly))
```

```
#2SD19EC009
#leap year
year9=2020
if not year9%400:
  is_leap_year=True
elif not year9%100:
  is_leap_year9=False
elif not year9%4:
  is_leap_year9=True
else:
  is_leap_year9=False
s_ly ='is a' if is_leap_year9 else 'is not a'
print('{:4d} {:s} leap_year9'.format(year9, s_ly))
```

```
2020 is a leap_year9
```

76)

```
for val9 in "string":
    if val9 == "i":
        break
    print(val9)
print("The end")
```

```
for val9 in "string":
    if val9 == "i":
        break
    print(val9)

print("The end")
```

```
s
t
r
The end
```

77)

```
a9=500
b9=400
if a9>b9:
  print('a9 is greater than b9')
a9,b9=b9,a9
print('a9 is greater than b9') if a9>b9 else print('b9 is greater than
a9')
```

```
#2SD19EC009
a9=500
b9=400
if a9>b9:
  print('a9 is greater than b9')
a9,b9=b9,a9
print('a9 is greater than b9') if a9>b9 else print('b9 is greater than a9')
```

```
a9 is greater than b9
b9 is greater than a9
```

78)

```
i=1
while i<9:
  print(i)
  i+=1
else:
  print('i is no longer less than 9')
```

```
#2SD19EC009
i=1
while i<9:
  print(i)
  i+=1
else:
  print('i is no longer less than 9')
```

```
1
2
3
4
5
6
7
8
i is no longer less than 9
```

79)

```
num9=int(input('enter a number'))
print(num9)
num_9=int(input('enter num2'))
print(num_9)
sum9=num9+num_9
print(sum9)
```

```
#2SD19EC009
num9=int(input('enter a number'))
print(num9)
num_9=int(input('enter num2'))
print(num_9)
sum9=num9+num_9
print(sum9)
```

```
enter a number65
65
enter num254
54
119
```

80)

```
#if int is not used
num9=input('enter a number')
print(num9)
num_9=input('enter num2')
print(num_9)
sum9=num9+num_9
print(sum9)
```

```
] #2SD19EC009
  #if int is not used
  num9=input('enter a number')
  print(num9)
  num_9=input('enter num2')
  print(num_9)
  sum9=num9+num_9
  print(sum9)
```

```
enter a number51
51
enter num225
25
5125
```

```
81)
n9=input('enter a number: ')
print(n9)
n09=input('enter second number: ')
print(n09)
sum9=n9+n09
print(sum9)
print('type of number',type(n9))
print('type of number',type(n09))
n009=int(input('enter a number: '))
print(n009)
n0009=int(input('enter second number: '))
print(n0009)
sum9=n009+n0009
print(sum9)
print('type of number',type(n009))
print('type of number',type(n0009))
n_9=float(input('enter a number: '))
print(n_9)
n_09=float(input('enter a number: '))
print(n_09)
sum9=n_9+n_09
print('type of number',type(n_9))
```

```
#2SD19EC009
n9=input('enter a number: ')
print(n9)
n09=input('enter second number: ')
print(n09)
sum9=n9+n09
print(sum9)
print('type of number',type(n9))
print('type of number',type(n09))
n009=int(input('enter a number: '))
print(n009)
n0009=int(input('enter second number: '))
print(n0009)
sum9=n009+n0009
print(sum9)
print('type of number',type(n009))
print('type of number',type(n0009))
n_9=float(input('enter a number: '))
print(n_9)
n_09=float(input('enter a number: '))
print(n_09)
sum9=n_9+n_09
print('type of number',type(n_9))
print('type of number',type(n_09))
```

```
enter a number: 5
5
enter second number: 6
6
56
type of number <class 'str'>
type of number <class 'str'>
enter a number: 3
3
enter second number: 6
6
9
type of number <class 'int'>
type of number <class 'int'>
enter a number: 5
5.0
enter a number: 4
4.0
type of number <class 'float'>
type of number <class 'float'>
```

82)

```
#function
def sq(x9):
  xsq=x9**2
  return xsq
p=2;
p1=sq(p)
print('x9 value =',p)
print('x9 squre value = ',p1)
```

```
#2SD19EC009
#function
def sq(x9):
  xsq=x9**2
  return xsq
p=2;
p1=sq(p)
print('x9 value =',p)
print('x9 squre value = ',p1)
```

```
x9 value = 2
x9 squre value =  4
```

83)

```
import math
def roots(a,b,c):
  d=b**2-4*a*c
  r1=(-b+math.sqrt(d))/2/a
  r2=(-b-math.sqrt(d))/2/a
  return r1,r2
print('roots ARE ',roots(1.,-1.,-6.))
```

```
#2SD19EC009
import math
def roots(a,b,c):
  d=b**2-4*a*c
  r1=(-b+math.sqrt(d))/2/a
  r2=(-b-math.sqrt(d))/2/a
  return r1,r2
print('roots ARE ',roots(1.,-1.,-6.))
```

```
roots ARE  (3.0, -2.0)
```

```
84)
def my_function():
    '''Demonstrates triple double quotes
    docstrings and does nothing really.'''

    return None

print("Using __doc__:")
print(my_function.__doc__)

print("Using help:")
help(my_function)
```

```
#2SD19EC009
def my_function():
    '''Demonstrates triple double quotes
    docstrings and does nothing really.'''

    return None

print("Using __doc__:")
print(my_function.__doc__)

print("Using help:")
help(my_function)
```

```
Using __doc__:
Demonstrates triple double quotes
    docstrings and does nothing really.
Using help:
Help on function my_function in module __main__:

my_function()
    Demonstrates triple double quotes
    docstrings and does nothing really.
```

85)

```
def student9(firstname, lastname ='Mark', standard ='Fifth'):
     print(firstname, lastname, 'studies in', standard, 'Standard')

# 1 positional argument
student9('John')

# 3 positional arguments
student9('John', 'Gates', 'Seventh')

# 2 positional arguments
student9('John', 'Gates')
student9('John', 'Seventh')
```

```
56] #2SD19EC009
    def student9(firstname, lastname ='Mark', standard ='Fifth'):
         print(firstname, lastname, 'studies in', standard, 'Standard')

    # 1 positional argument
    student9('John')

    # 3 positional arguments
    student9('John', 'Gates', 'Seventh')

    # 2 positional arguments
    student9('John', 'Gates')
    student9('John', 'Seventh')
```

```
John Mark studies in Fifth Standard
John Gates studies in Seventh Standard
John Gates studies in Fifth Standard
John Seventh studies in Fifth Standard
```

86)

```
def myfunc1():
  x = "John"
  def myfunc2():
    nonlocal x
    x = "hello"
  myfunc2()
  return x
```

```
#2SD19EC009
def myfunc1():
  x = "John"
  def myfunc2():
    nonlocal x
    x = "hello"
  myfunc2()
  return x

print(myfunc1())
```

```
hello
```

87)

```
def my_function9(name09):
  print(name09 + " hi")

my_function9("Emil")
my_function9("Tobias")
my_function9("Linus")
```

```
#2SD19EC009
def my_function9(name09):
  print(name09 + " hi")

my_function9("Emil")
my_function9("Tobias")
my_function9("Linus")
```

```
Emil hi
Tobias hi
Linus hi
```

88)

```
def student9(firstname,lastname='p',standard='fifth sem'):
  print(firstname,lastname,'studies in',standard,'standard')
student9('ajeet')
student9('tejas','c','fifth sem')
student9('james','gate')
```

```
#2SD19EC009
def student9(firstname,lastname='p',standard='fifth sem'):
  print(firstname,lastname,'studies in',standard,'standard')
student9('ajeet')
student9('tejas','c','fifth sem')
student9('james','gate')
```

```
ajeet p studies in fifth sem standard
tejas c studies in fifth sem standard
james gate studies in fifth sem standard
```

89)

```
def func1():
  x9=22
  print('x9 value in func1=',x9)
def func2():
  x=7
  x+=1
  print('x9 value in func2=',x9)
func1()
func2()
print('x9 value in the main=',x9)
```

```
#2SD19EC009
def func1():
  x9=22
  print('x9 value in func1=',x9)
def func2():
  x=7
  x+=1
  print('x9 value in func2=',x9)
func1()
func2()
print('x9 value in the main=',x9)
```

```
x9 value in func1= 22
x9 value in func2= 9
x9 value in the main= 9
```

90)

```
def evenodd(x9):
  if(x9%2==0):
    print('The number',x9,'is even')
  else:
    print('The number',x9,'is odd')
evenodd(212)
evenodd(21233)
```

```
#2SD19EC009
def evenodd(x9):
  if(x9%2==0):
    print('The number',x9,'is even')
  else:
    print('The number',x9,'is odd')
evenodd(212)
evenodd(21233)
```

```
The number 212 is even
The number 21233 is odd
```

91)

```
def func1():
  x9=22
  print('x9 value in func1=',x9)
def func2():
  x=7
  x+=1
  print('x9 value in func2=',x9)
func1()
func2()
print('x9 value in the main=',x9)
```

```
#2SD19EC009
def my_function9(x):
  x=[20,30,615]
lst=[10,11,12,13,14,15]
my_function9(lst)
print(lst)
```

```
[10, 11, 12, 13, 14, 15]
```

92)

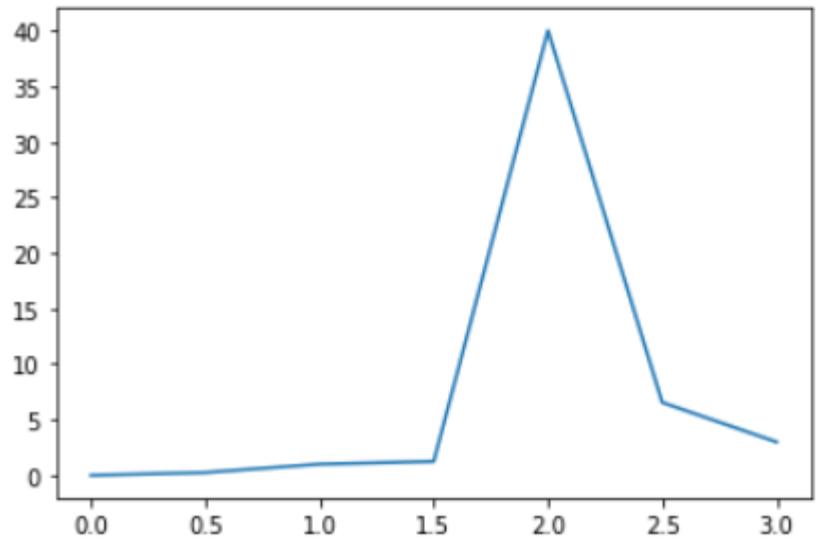
```
total =0
def sum9(arg1,arg2):
  total=arg1+arg2
  print('inside local time',total)
  return total
  sum(120, 50)
print('outside local time',total)
```

```
#2SD19EC009
total =0
def sum9(arg1,arg2):
  total=arg1+arg2
  print('inside local time',total)
  return total
  sum(120, 50)
print('outside local time',total)
```

```
outside local time 0
```

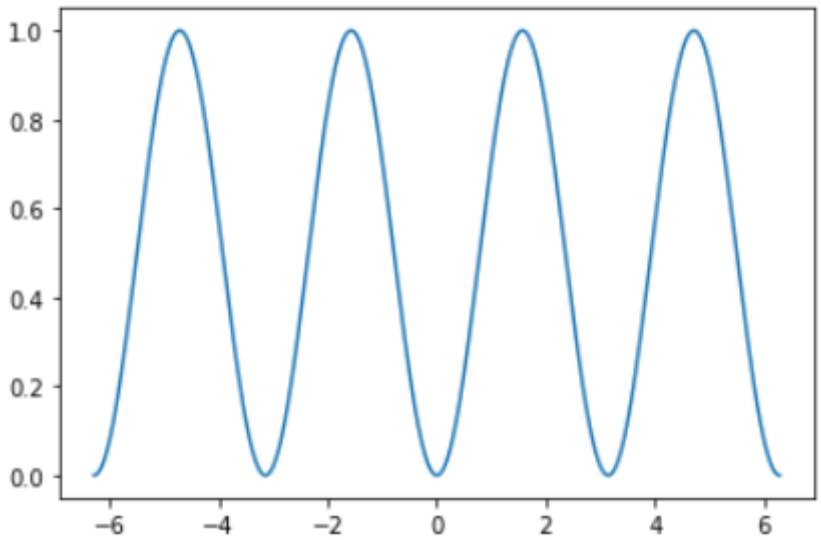
```
93)
import pylab
ax=[0.0,0.5,1.0,1.5,2.0,2.5,3.0]
ay=[0.0,0.25,1.0,1.25,40.,6.55,3.0]
pylab.plot(ax,ay)
pylab.show()
```

```
#2SD19EC009
import pylab
ax=[0.0,0.5,1.0,1.5,2.0,2.5,3.0]
ay=[0.0,0.25,1.0,1.25,40.,6.55,3.0]
pylab.plot(ax,ay)
pylab.show()
```



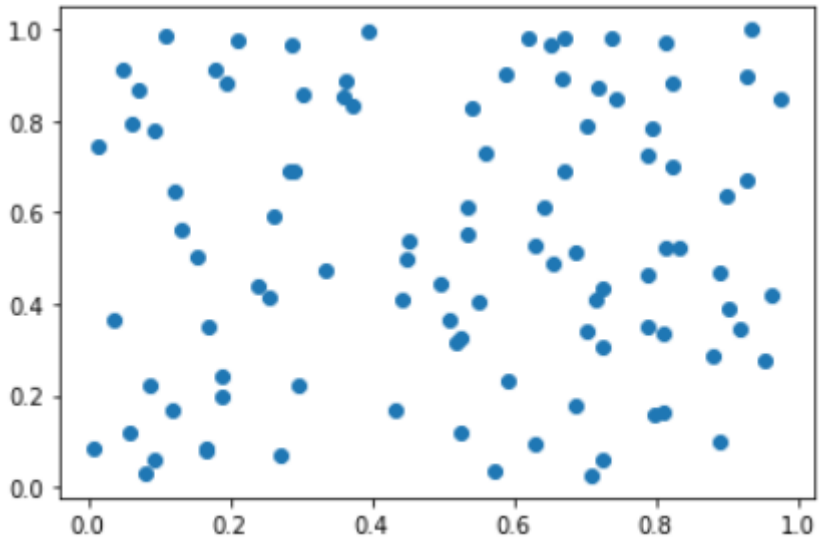
```
94)
import math
xmin9,xmax9=-2.*math.pi,2.*math.pi
n=1000
x=[0.]*n
y=[0.]*n
dx=(xmax9-xmin9)/(n-1)
for i in range(n):
  xpt=xmin+i*dx
  x[i]=xpt
  y[i]=math.sin(xpt)**2
pylab.plot(x,y)
pylab.show()
```

```
#2SD19EC009
import math
xmin9,xmax9=-2.*math.pi,2.*math.pi
n=1000
x=[0.]*n
y=[0.]*n
dx=(xmax9-xmin9)/(n-1)
for i in range(n):
  xpt=xmin+i*dx
  x[i]=xpt
  y[i]=math.sin(xpt)**2
pylab.plot(x,y)
pylab.show()
```



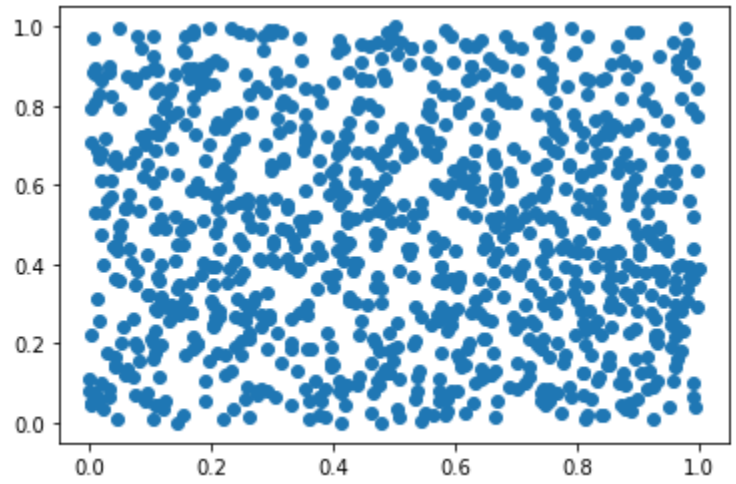
```
95)
import random
ax9,ay9=[],[]
for i in range(100):
  ax9.append(random.random())
  ay9.append(random.random())
pylab.scatter(ax9,ay9)
pylab.show()
```

```
#2SD19EC009
import random
ax9,ay9=[],[]
for i in range(100):
  ax9.append(random.random())
  ay9.append(random.random())
pylab.scatter(ax9,ay9)
pylab.show()
```



```
96)
import random
import pylab
ax9,ay9=[],[]
for i in range(1000):
  ax9.append(random.random())
  ay9.append(random.random())
pylab.scatter(ax9,ay9)
pylab.show()
```

```
#2SD19EC009
import random
import pylab
ax9,ay9=[],[]
for i in range(1000):
  ax9.append(random.random())
  ay9.append(random.random())
pylab.scatter(ax9,ay9)
pylab.show()
```



97)

```
x9=pylab.linspace(-20,20,20)
print(x9)
```

linespace()

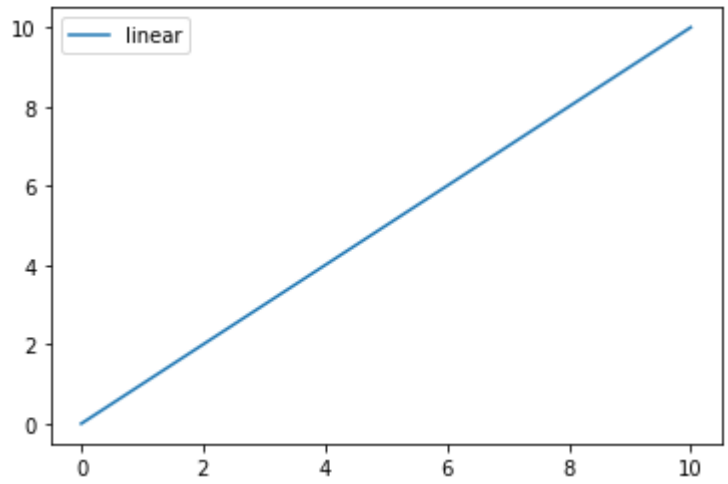
```
[4]  #2SD19EC009
     x9=pylab.linspace(-20,20,20)
     print(x9)
```

```
[-20.         -17.89473684 -15.78947368 -13.68421053 -11.57894737
  -9.47368421  -7.36842105  -5.26315789  -3.15789474  -1.05263158
   1.05263158   3.15789474   5.26315789   7.36842105   9.47368421
  11.57894737  13.68421053  15.78947368  17.89473684  20.        ]
```

98)

```
import numpy as np
import pylab as plt
x9=np.linspace(0,10,100)
plt.plot(x9,x9,label='linear')
plt.legend()
plt.show()
```

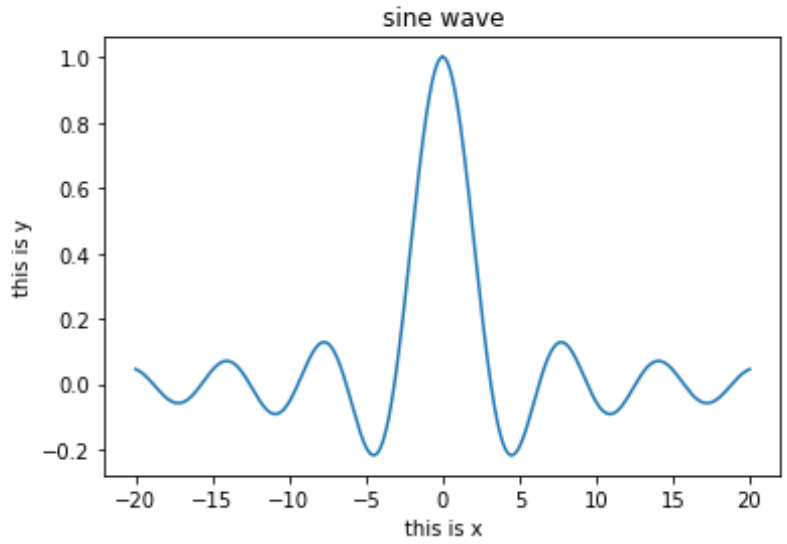
```
#2SD19EC009
import numpy as np
import pylab as plt
x9=np.linspace(0,10,100)
plt.plot(x9,x9,label='linear')
plt.legend()
plt.show()
```



```
99)
plt.xlabel('this is x')
plt.ylabel('this is y')
plt.title('sine wave')
x9=plt.linspace(-20,20,1001)
y9=plt.sin(x9)/x9
plt.plot(x9,y9)
plt.show()
#RuntimeWarning: invalid value encountered in true_divide
```

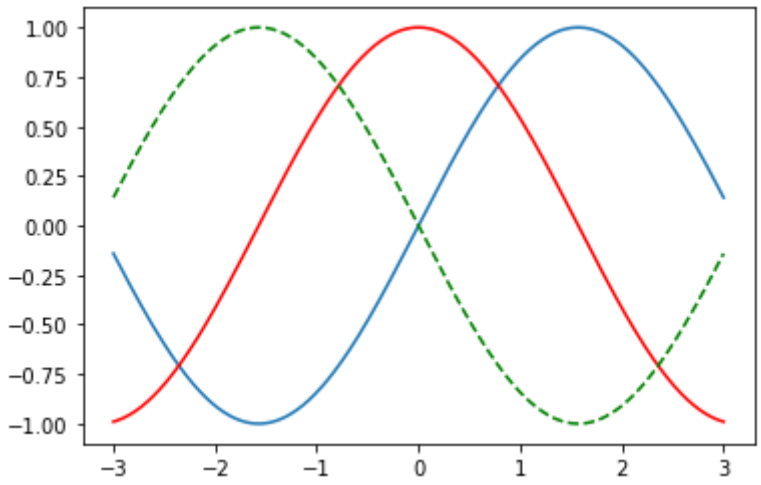
```
#2SD19EC009
plt.xlabel('this is x')
plt.ylabel('this is y')
plt.title('sine wave')
x9=plt.linspace(-20,20,1001)
y9=plt.sin(x9)/x9
plt.plot(x9,y9)
plt.show()
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_lau
```



```
100)
from pylab import*
x9=pylab.linspace(-3,3,99)
plot(x9,sin(x9))
plot(x9,cos(x9),'r-')
plot(x9,-sin(x9),'g--')
show()
```

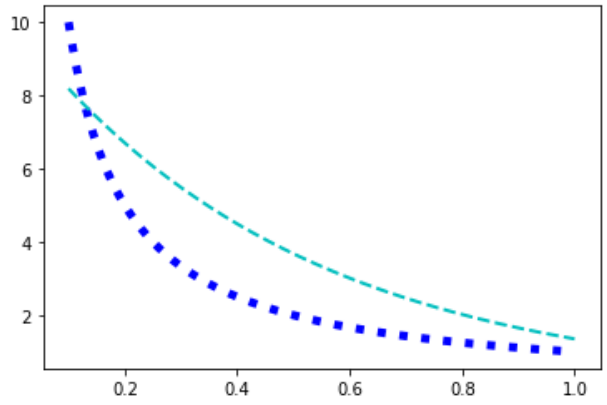
```
#2SD19EC009
from pylab import*
x9=pylab.linspace(-3,3,300)
plot(x9,sin(x9))
plot(x9,cos(x9),'r-')
plot(x9,-sin(x9),'g--')
show()
```



101)

```
import pylab
a9=pylab.linspace(0.1,1.,1000)
y9=1./a9
z9=10. *pylab.exp(-2.*a9)
pylab.plot(a9,y9,color='b',linestyle=':',linewidth=5.)
pylab.plot(a9,z9,color='c',linestyle='--',linewidth=2.)
pylab.show()
```

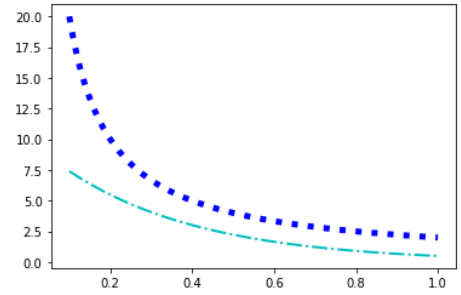
```
#2SD19EC009
import pylab
a9=pylab.linspace(0.1,1.,1000)
y9=1./a9
z9=10. *pylab.exp(-2.*a9)
pylab.plot(a9,y9,color='b',linestyle=':',linewidth=5.)
pylab.plot(a9,z9,color='c',linestyle='--',linewidth=2.)
pylab.show()
```



102)

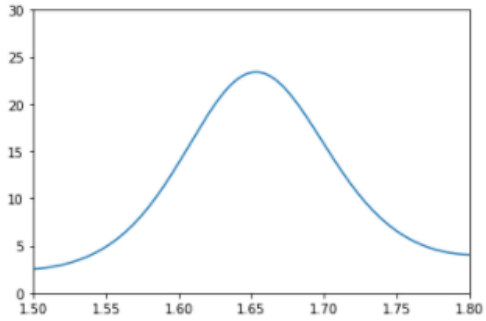
```
import pylab
a9=pylab.linspace(0.1,1.,1000)
y9=2./a9
z9=10. *pylab.exp(-3.*a9)
pylab.plot(a9,y9,color='b',linestyle=':',linewidth=5.)
pylab.plot(a9,z9,color='c',linestyle='-.',linewidth=2.)
```

```
#2SD19EC009
import pylab
a9=pylab.linspace(0.1,1.,1000)
y9=2./a9
z9=10. *pylab.exp(-3.*a9)
pylab.plot(a9,y9,color='b',linestyle=':',linewidth=5.)
pylab.plot(a9,z9,color='c',linestyle='-.',linewidth=2.)
pylab.show()
```

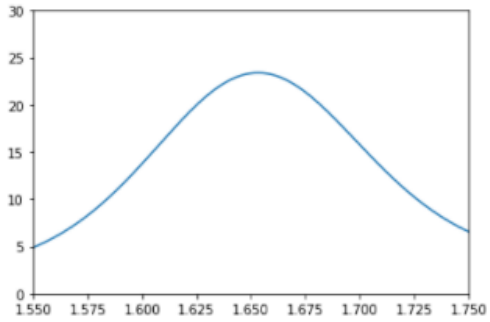


```
103)
t9=pylab.linspace(0,2,1000)
f9=t9*pylab.exp(t9+pylab.sin(20*t9))
pylab.plot(t9,f9)
pylab.xlim(1.5,1.7)
pylab.ylim(0,30)
pylab.show()
 and if any part of the graph to be zoomed!
2SD19EC009
t9=pylab.linspace(0,2,1000)
f9=t9*pylab.exp(t9+pylab.sin(20*t9))
pylab.plot(t9,f9)
pylab.xlim(1.55,1.75)
pylab.ylim(0,30)
pylab.show()
```

```
6] #2SD19EC009
   t9=pylab.linspace(0,2,1000)
   f9=t9*pylab.exp(t9+pylab.sin(20*t9))
   pylab.plot(t9,f9)
   pylab.xlim(1.5,1.8)
   pylab.ylim(0,30)
   pylab.show()
```

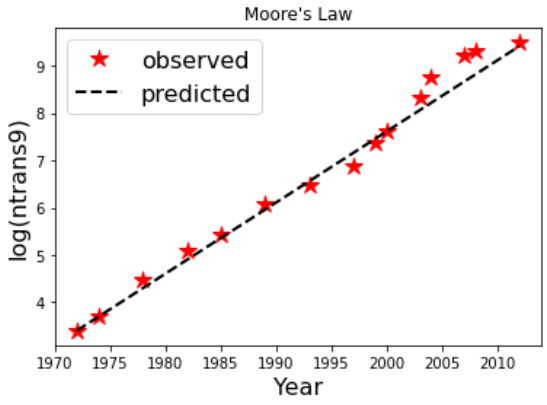



```
#2SD19EC009
t9=pylab.linspace(0,2,1000)
f9=t9*pylab.exp(t9+pylab.sin(20*t9))
pylab.plot(t9,f9)
pylab.xlim(1.55,1.75)
pylab.ylim(0,30)
pylab.show()
```

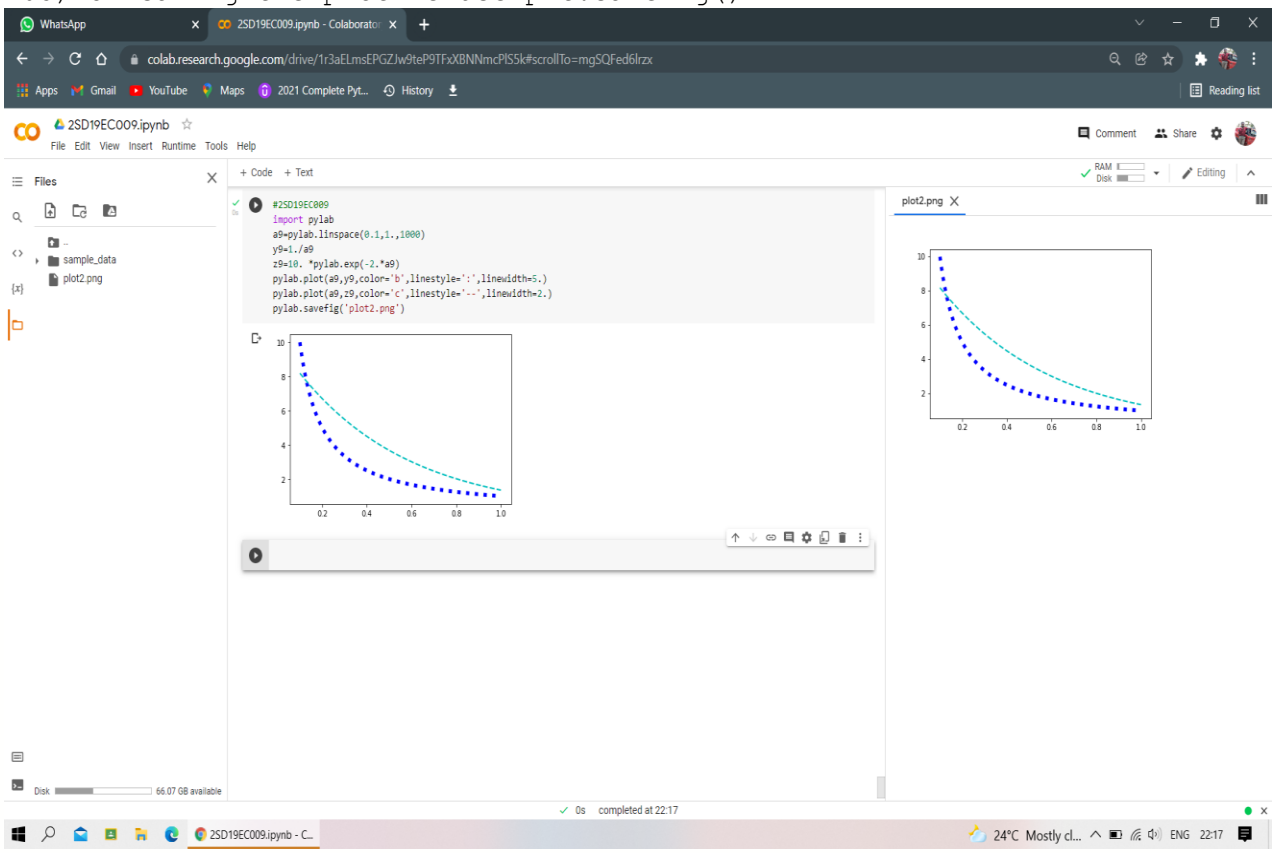


```
104)
import pylab
year9 = [1972, 1974, 1978, 1982, 1985, 1989, 1993, 1997, 1999, 2000, 20
03, 2004, 2007, 2008, 2012]
ntrans9 = [0.0025, 0.005, 0.029, 0.12, 0.275, 1.18, 3.1, 7.5, 24.0, 42.
0, 220.0, 592.0, 1720.0, 2046.0, 3100.0]
ntrans9 = pylab.array(ntrans) * 1.e6
y09, n09 = year[0], ntrans[0]
y009 = pylab.linspace(y0, year[-1], year[-1] - y0 + 1)
T9 = 2.
moore = pylab.log10(n09) + (y009 - y09) / T2 * pylab.log10(2)

pylab.plot(year9, pylab.log10(ntrans), '*', markersize=12, color='r',
           markeredgecolor='r', label='observed')
pylab.plot(y009, moore, linewidth=2, color='k', linestyle='--
', label='predicted')
pylab.legend(fontsize=12, loc='upper left')
pylab.xlabel('Year', fontsize=16)
pylab.ylabel('log(ntrans9)', fontsize=12)
pylab.title("Moore's Law")
pylab.show()
```

```
#2SD19EC009
import pylab
year9 = [1972, 1974, 1978, 1982, 1985, 1989, 1993, 1997, 1999, 2000, 2003,
       2004, 2007, 2008, 2012]
ntrans9 = [0.0025, 0.005, 0.029, 0.12, 0.275, 1.18, 3.1, 7.5, 24.0, 42.0,
         220.0, 592.0, 1720.0, 2046.0, 3100.0]
ntrans9 = pylab.array(ntrans) * 1.e6

y09, n09 = year[0], ntrans[0]
y009 = pylab.linspace(y0, year[-1], year[-1] - y0 + 1)
# Time taken in years for the number of transistors to double
T9 = 2.
moore = pylab.log10(n09) + (y009 - y09) / T2 * pylab.log10(2)

pylab.plot(year9, pylab.log10(ntrans), '*', markersize=12, color='r',
           markeredgecolor='r', label='observed')
pylab.plot(y009, moore, linewidth=2, color='k', linestyle='--', label='predicted'
pylab.legend(fontsize=16, loc='upper left')
pylab.xlabel('Year', fontsize=16)
pylab.ylabel('log(ntrans9)', fontsize=16)
pylab.title("Moore's Law")
pylab.show()
```



105)For saving the plot we use plt.savefig()



After saving it as plot2.png, it gets saved and via that we can even download that plot and view it separately.

106)

```
#marks9=10000
#a9=marks9/0
#print(a9)
```

```
#2SD19EC009
#marks9=10000
#a9=marks9/0
#print(a9)
```

```
---------------------------------------------------------------------------
ZeroDivisionError                         Traceback (most recent call last)
<ipython-input-1-f014ae278cee> in <module>()
      1 marks9=10000
----> 2 a9=marks9/0
      3 print(a9)

ZeroDivisionError: division by zero
```

SEARCH STACK OVERFLOW

107)

```
#for lambda in range a(8):
  #because lambda is a reserved keyword.
```

```
#2SD19EC009
#for lambda in range a(8):
  #because lambda is a reserved keyword.
```

```
  File "<ipython-input-4-f079f7d0aeee>", line 1
    for lambda in range a(8):
                ^
SyntaxError: invalid syntax
```

SEARCH STACK OVERFLOW

108)

```
#for f in range (8:
```

```
#2SD19EC009
#a=[1,2,3,4,
   b=5
```

```
  File "<ipython-input-7-a3256f65e596>", line 2
    b=5
     ^
SyntaxError: invalid syntax
```

SEARCH STACK OVERFLOW

109)

```
#a=[1,2,3,4,
   b=5
```

```
#2SD19EC009
#if a=5:
  #we are supposed to use if a==5
```

```
  File "<ipython-input-8-bd1640a34efc>", line 1
    if a=5:
       ^
SyntaxError: invalid syntax
```

SEARCH STACK OVERFLOW

110)

```
#if a=5:
  #we are supposed to use if a==5
```

```
#2SD19EC009
print('4z =',4*z)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-9-c6da18b70544> in <module>()
      1 #2SD19EC009
----> 2 print('4z =',4*z)

NameError: name 'z' is not defined
```

SEARCH STACK OVERFLOW

111)

```
#print('4z =',4*z)
#2SD19EC009
'00'+77
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-11-3f517026654d> in <module>()
      1 #2SD19EC009
----> 2 '00'+77

TypeError: can only concatenate str (not "int") to str
```

SEARCH STACK OVERFLOW

112)

```
#'00'+77
#2SD19EC009
'00'+77
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-11-3f517026654d> in <module>()
      1 #2SD19EC009
----> 2 '00'+77

TypeError: can only concatenate str (not "int") to str
```

SEARCH STACK OVERFLOW

113)

```
#x9=70
#if x9>50:
#  raise Exception('x9 should not exceed 50. the value of x9 was: ',x9)
#if x9<20:
 # raise Exception('x9 should be less than 20. the value os x9 was: ',x
9)
#2SD19EC009
x9=70
if x9>50:
  raise Exception('x9 should not exceed 50. the value of x9 was: ',x9)
if x9<20:
  raise Exception('x9 should be less than 20. the value os x9 was: ',x9)
```

```
---------------------------------------------------------------------------
Exception                                 Traceback (most recent call last)
<ipython-input-20-77498208babb> in <module>()
      2 x9=70
      3 if x9>50:
----> 4   raise Exception('x9 should not exceed 50. the value of x9 was: ',x9)
      5 if x9<20:
      6   raise Exception('x9 should be less than 20. the value os x9 was: ',x9

Exception: ('x9 should not exceed 50. the value of x9 was: ', 70)
```

SEARCH STACK OVERFLOW

114)

```
n9=8
if n9%2:
  raise ValueError('n9 must be even! ')
print('a')
```

```
#2SD19EC009
n9=8
if n9%2:
  raise ValueError('n9 must be even! ')
print('a')
```

```
a
```

115)

```
#n9=9
#if n9%2:
#  raise ValueError('n9 must be even! ')
#print('a')
```

```
#2SD19EC009
n9=9
if n9%2:
  raise ValueError('n9 must be even! ')
print('a')
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-25-b730959b9a6e> in <module>()
      2 n9=9
      3 if n9%2:
----> 4   raise ValueError('n9 must be even! ')
      5 print('a')

ValueError: n9 must be even!
```

SEARCH STACK OVERFLOW

116)

```
x9=0
try:
  y9=1/x9
  print('1/', x9,'=',y9)
except ZeroDivisionError:
  print('1/0 is not defined')
```

```
#2SD19EC009
x9=0
try:
  y9=1/x9
  print('1/', x9,'=',y9)
except ZeroDivisionError:
  print('1/0 is not defined')
```

```
1/0 is not defined
```

117)

```
x9=10.2
if x9>50:
  raise Exception('x9 should not exceed 50. the value of x9 was: ',x9)
if x9<20:
   raise Exception('x9 should not less than 20. the value of x9 was: ',
x9)
```

```
#2SD19EC009
x9=10.2
if x9>50:
  raise Exception('x9 should not exceed 50. the value of x9 was: ',x9)
if x9<20:
   raise Exception('x9 should not less than 20. the value of x9 was: ',x9)
```

```
---------------------------------------------------------------------------
Exception                                 Traceback (most recent call last)
<ipython-input-5-bed9fe58595b> in <module>()
      4   raise Exception('x9 should not exceed 50. the value of x9 was: ',x9)
      5 if x9<20:
----> 6    raise Exception('x9 should not less than 20. the value of x9 was: ',x9)

Exception: ('x9 should not less than 20. the value of x9 was: ', 10.2)
```

SEARCH STACK OVERFLOW

```
118)
n9=8
if n9%2:
  raise ValueError('n9 must be even!!')
print('no')
```

and

```
119)
n9=9
if n9%2:
  raise ValueError('n9 must be even!!')
print('no')
```

```
#2SD19EC009
n9=8
if n9%2:
  raise ValueError('n9 must be even!!')
print('no')
```

```
no
```

```
#2SD19EC009
n9=9
if n9%2:
  raise ValueError('n9 must be even!!')
print('no')
```

```
---------------------------------------------------
ValueError                                Traceba
<ipython-input-8-ae5948a4edab> in <module>()
      2 n9=9
      3 if n9%2:
----> 4   raise ValueError('n9 must be even!!')
      5 print('no')

ValueError: n9 must be even!!
```

SEARCH STACK OVERFLOW

120)

```
list_num9=[1,2,3,4]
tup_num9=(1,2,3,4)
list_num9[2]=5
print(list_num9)
print(dir(list_num9))
```

```
#2SD19EC009
list_num9=[1,2,3,4]
tup_num9=(1,2,3,4)
list_num9[2]=5
print(list_num9)
print(dir(list_num9))
```

```
[1, 2, 5, 4]
['__add__', '__class__', '__contains__', '__delattr__', '__delitem__', '__dir__', '_
```

121)

```
a9=(1,2,3,4,5,6,7,8,9,0)
b9=[1,2,3,4,5,6,7,8,9,0]
print('a9=',a9.__sizeof__())
print('b9=',b9.__sizeof__())
```

```
#2SD19EC009
a9=(1,2,3,4,5,6,7,8,9,0)
b9=[1,2,3,4,5,6,7,8,9,0]
print('a9=',a9.__sizeof__())
print('b9=',b9.__sizeof__())
```

```
a9= 104
b9= 120
```

```
122)
a9=['a','b','c','d']
b9=['d','e','f','g']
print(a9==b9)
c9=['aa','bb','cc','dd']
d9=c9
print(d9)
print(c9[1])
print(c9[-1])
print(c9[2])
print(c9[-3])
print(c9[2:4])
print(c9[-4:-2])
print(c9[:4])
print(c9[0:4])
print(c9[3:])
print(c9[1:-1])
print(c9[::-1])
print(c9[:])
print(c9==a9)
print(b9==c9)
print(c9[:] is c9)
```

```
#2SD19EC009
a9=['a','b','c','d']
b9=['d','e','f','g']
print(a9==b9)
c9=['aa','bb','cc','dd']
d9=c9
print(d9)
print(c9[1])
print(c9[-1])
print(c9[2])
print(c9[-3])
print(c9[2:4])
print(c9[-4:-2])
print(c9[:4])
print(c9[0:4])
print(c9[3:])
print(c9[1:-1])
print(c9[::-1])
print(c9[:])
print(c9==a9)
print(b9==c9)
print(c9[:] is c9)
```

```
False
['aa', 'bb', 'cc', 'dd']
bb
dd
cc
bb
['cc', 'dd']
['aa', 'bb']
['aa', 'bb', 'cc', 'dd']
['aa', 'bb', 'cc', 'dd']
['dd']
['bb', 'cc']
['dd', 'cc', 'bb', 'aa']
['aa', 'bb', 'cc', 'dd']
False
False
False
```

123)

```
s9=['aa','bb','cc']
v9=['dd','ee']
print(s9+v9)
print(v9*2)
print(len(s9))
print(min(s9))
print(max(v9))
```

```
#2SD19EC009
s9=['aa','bb','cc']
v9=['dd','ee']
print(s9+v9)
print(v9*2)
print(len(s9))
print(min(s9))
print(max(v9))
```

```
['aa', 'bb', 'cc', 'dd', 'ee']
['dd', 'ee', 'dd', 'ee']
3
aa
ee
```

124)

```
s9=['aa','bb','cc']
v9=['dd','ee']
v09=['ff','gg','hh',v9,s9]
m9=['ii','jj',v09]
print(v09[3][0])
print(m9[2][3][1])
print(m9[2][4][-1])
```

```
#2SD19EC009
s9=['aa','bb','cc']
v9=['dd','ee']
v09=['ff','gg','hh',v9,s9]
m9=['ii','jj',v09]
print(v09[3][0])
print(m9[2][3][1])
print(m9[2][4][-1])
```

```
dd
ee
cc
```

125)

```
s9=['a','b']
s9.append([5,6,7])
s9.extend([7,8,9])
print(s9)
```

```
5] #2SD19EC009
   s9=['a','b']
   s9.append([5,6,7])
   s9.extend([7,8,9])
   print(s9)
```

```
['a', 'b', [5, 6, 7], 7, 8, 9]
```

126)

```
st9=(1,2,3,4,5)
print(st9)
print ('addres of st= ',id(st9))
st09=(1,2,3,4,5,98)
print(st09)
print ('addres of st= ',id(st09))
```

```
#2SD19EC009
st9=(1,2,3,4,5)
print(st9)
print ('addres of st= ',id(st9))
st09=(1,2,3,4,5,98)
print(st09)
print ('addres of st= ',id(st09))
```

```
(1, 2, 3, 4, 5)
addres of st=  139932143990832
(1, 2, 3, 4, 5, 98)
addres of st=  139932143763360
```

127)

```
b9=(1,[1,2,3],4)
print('b9',b9)
print('addressof b9[1]=',id(b9[1]))
b9[1].append(7)
print('b9',b9)
print('addressof b9[1]=',id(b9[1]))
```

```
#2SD19EC009
b9=(1,[1,2,3],4)
print('b9',b9)
print('addressof b9[1]=',id(b9[1]))
b9[1].append(7)
print('b9',b9)
print('addressof b9[1]=',id(b9[1]))
```

```
b9 (1, [1, 2, 3], 4)
addressof b9[1]= 139932014181264
b9 (1, [1, 2, 3, 7], 4)
addressof b9[1]= 139932014181264
```

128)

```
def myfun(*argv):
  for arg in argv:
    print(arg)
myfun("hello","welcome","to","dwr")
```

```
#2SD19EC009
def myfun(*argv):
  for arg in argv:
    print(arg)
myfun("hello","welcome","to","dwr")
```

```
hello
welcome
to
dwr
```

```
129)
tel9={'jack':1010,'sape':1215}
tel9['guido']=2585
print(tel9)
tel9['jack']=4040
print(tel9)
tel9['kalpana']=4548
print(tel9)
print(sorted(tel9))
for k9,v9 in tel9.items():
  print(k9,v9)
print(tel9.keys()
print(tel9.items())
print(id(tel9))
print(id(tel9['guido']))
```

```
#2SD19EC009
tel9={'jack':1010,'sape':1215}
tel9['guido']=2585
print(tel9)
tel9['jack']=4040
print(tel9)
tel9['kalpana']=4548
print(tel9)
print(sorted(tel9))
for k9,v9 in tel9.items():
  print(k9,v9)
print(tel9.keys())
print(tel9.items())
print(id(tel9))
print(id(tel9['guido']))
```

```
{'jack': 1010, 'sape': 1215, 'guido': 2585}
{'jack': 4040, 'sape': 1215, 'guido': 2585}
{'jack': 4040, 'sape': 1215, 'guido': 2585, 'kalpana': 4548}
['guido', 'jack', 'kalpana', 'sape']
jack 4040
sape 1215
guido 2585
kalpana 4548
dict_keys(['jack', 'sape', 'guido', 'kalpana'])
dict_items([('jack', 4040), ('sape', 1215), ('guido', 2585), ('kalpana', 4548)])
139932014066576
139932014107440
```

```
130)
import math
body = {'Sun': (1.988e30, 6.955e5),
         'Mercury': (3.301e23, 2440.),
         'Venus': (4.867e+24, 6052.),
         'Earth': (5.972e24, 6371.),
         'Mars': (6.417e23, 3390.),
         'Jupiter': (1.899e27, 69911.),
         'Saturn': (5.685e26, 58232.),
         'Uranus': (8.682e25, 25362.),
         'Neptune': (1.024e26, 24622.)
        }
planets = list(body.keys())
planets.remove('Sun')
def calc_density(m, r):
    """ Returns the density of a sphere with mass m and radius r. """
    return m / (4/3 * math.pi * r**3)
rho = {}
for planet in planets:
    m, r = body[planet]
    rho[planet] = calc_density(m*1000, r*1.e5)
for planet, density in sorted(rho.items()):
    print('The density of {0} is {1:3.2f} g/cm3'.format(planet, density
```

```
#2SD19EC009
import math
body = {'Sun': (1.988e30, 6.955e5),
         'Mercury': (3.301e23, 2440.),
         'Venus': (4.867e+24, 6052.),
         'Earth': (5.972e24, 6371.),
         'Mars': (6.417e23, 3390.),
         'Jupiter': (1.899e27, 69911.),
         'Saturn': (5.685e26, 58232.),
         'Uranus': (8.682e25, 25362.),
         'Neptune': (1.024e26, 24622.)
        }

planets = list(body.keys())
planets.remove('Sun')

def calc_density(m, r):
    """ Returns the density of a sphere with mass m and radius r. """
    return m / (4/3 * math.pi * r**3)

rho = {}
for planet in planets:
    m, r = body[planet]
    rho[planet] = calc_density(m*1000, r*1.e5)

for planet, density in sorted(rho.items()):
    print('The density of {0} is {1:3.2f} g/cm3'.format(planet, density))
```

```
The density of Earth is 5.51 g/cm3
The density of Jupiter is 1.33 g/cm3
The density of Mars is 3.93 g/cm3
The density of Mercury is 5.42 g/cm3
The density of Neptune is 1.64 g/cm3
The density of Saturn is 0.69 g/cm3
The density of Uranus is 1.27 g/cm3
The density of Venus is 5.24 g/cm3
```

))

131)

```
def adder(*num):
  sum=0
  for n in num:
    sum+=n
  print("sum:",sum)
adder(3,5)
adder(5,7,99)
adder(78,87,95)
```

```
#2SD19EC009
def adder(*num):
  sum=0
  for n in num:
    sum+=n
  print("sum:",sum)
adder(3,5)
adder(5,7,99)
adder(78,87,95)
```

```
sum: 8
sum: 111
sum: 260
```

132)

```
from numpy.core.defchararray import multiply

def multiply(*args):
  z=1
  for num in args:
    z*=num
  print(z)
multiply(3,5)
multiply(5,7,99)
multiply(78,87,95)
```

```
#2SD19EC009
def multiply(*args):
  z=1
  for num in args:
    z*=num
  print(z)
multiply(3,5)
multiply(5,7,99)
multiply(78,87,95)
```

```
15
3465
644670
```

133)

```
s9=set([1,1,4,3,2,2,3,4,1,3,"surprise!"])
print('set s9=',s9)
print('length of the set=',len(s9))
```

```
#2sd19ec009
s9=set([1,1,4,3,2,2,3,4,1,3,"surprise!"])
print('set s9=',s9)
print('length of the set=',len(s9))
```

```
set s9= {1, 2, 3, 4, 'surprise!'}
length of the set= 5
```

134)

```
s9=set([1,1,4,3,2,2,3,4,1,3,"surprise!"])
print('set s9=',s9)
print('length of the set=',len(s9))
s9.add("orange")
print(s9)
s9.update(["orange","mango","grapes"])
print(s9)
print(len(s9))
s9.remove("orange")
print(s9)
```

```
#2sd19ec009
s9=set([1,1,4,3,2,2,3,4,1,3,"surprise!"])
print('set s9=',s9)
print('length of the set=',len(s9))
s9.add("orange")
print(s9)
s9.update(["orange","mango","grapes"])
print(s9)
print(len(s9))
s9.remove("orange")
print(s9)
```

```
set s9= {1, 2, 3, 4, 'surprise!'}
length of the set= 5
{1, 2, 3, 4, 'surprise!', 'orange'}
{1, 2, 3, 4, 'surprise!', 'mango', 'grapes', 'orange'}
8
{1, 2, 3, 4, 'surprise!', 'mango', 'grapes'}
```

135)

```
s9=set([1,1,4,3,2,2,3,4,1,3,"surprise!"])
print('set s9=',s9)
p9=s9.pop()
print("poppes=d is p9=",p9)
print("therefore now s9=",s9)
p09=s9.clear()
print(p09)
print(s9)
```

```
] #2SD19EC009
  s9=set([1,1,4,3,2,2,3,4,1,3,"surprise!"])
  print('set s9=',s9)
  p9=s9.pop()
  print("poppes=d is p9=",p9)
  print("therefore now s9=",s9)
  p09=s9.clear()
  print(p09)
  print(s9)
```

```
set s9= {1, 2, 3, 4, 'surprise!'}
poppes=d is p9= 1
therefore now s9= {2, 3, 4, 'surprise!'}
None
set()
```

136)

```
s9=set([1,1,4,3,2,2,3,4,1,3,"surprise!"])
s09=set(("sdm","bvb","git"))
s9.update(s09)
print('set s9=',s9)
```

```
#2SD19EC009
s9=set([1,1,4,3,2,2,3,4,1,3,"surprise!"])
s09=set(("sdm","bvb","git"))
s9.update(s09)
print('set s9=',s9)
```

```
set s9= {1, 2, 3, 4, 'git', 'bvb', 'surprise!', 'sdm'}
```

137)

```
a9={1,2,3,4,5,6}
b9={4,5,6,7,8,9}
print("the union of a9 and b9=",'\n',a9|b9)
print(1 in a9)
print(88 in b9)
for letters in set("SDMCET"):
  print(letters)
```

```
#2SD19EC009
a9={1,2,3,4,5,6}
b9={4,5,6,7,8,9}
print("the union of a9 and b9=",'\n',a9|b9)
print(1 in a9)
print(88 in b9)
for letters in set("SDMCET"):
  print(letters)
```

```
the union of a9 and b9=
 {1, 2, 3, 4, 5, 6, 7, 8, 9}
True
False
E
T
S
C
M
D
```

138)

```
a=frozenset((1,2,3))
b=set(('q',(1,2),a))
print('b=',b)
```

```
#2sd19ec009
a=frozenset((1,2,3))
b=set(('q',(1,2),a))
print('b=',b)
```

```
b= {(1, 2), 'q', frozenset({1, 2, 3})}
```

139)

```
a9={1,2,3,4,5,6}
b9={4,5,6,7,8,9}
print("the union of a9 and b9=",'\n',a9|b9)
print("the intersection of a9 and b9",'\n',a9&b9)
print("the difference:",'\n',a9-b9)
print('the symmetric diff:','\n',a9^b9)
```

```
#2SD19EC009
a9={1,2,3,4,5,6}
b9={4,5,6,7,8,9}
print("the union of a9 and b9=",'\n',a9|b9)
print("the intersection of a9 and b9",'\n',a9&b9)
print("the difference:",'\n',a9-b9)
print('the symmetric diff:','\n',a9^b9)
```

```
the union of a9 and b9=
 {1, 2, 3, 4, 5, 6, 7, 8, 9}
the intersection of a9 and b9
 {4, 5, 6}
the difference:
 {1, 2, 3}
the symmetric diff:
 {1, 2, 3, 7, 8, 9}
```

140)

```
a9={'a','b','c'}
a9.add('d')
print(a9)
fa9=frozenset(a9)
fa9=fa9|{'e'}
print(fa9)
```

```
#2sd19ec009
a9={'a','b','c'}
a9.add('d')
print(a9)
fa9=frozenset(a9)
fa9=fa9|{'e'}
print(fa9)
```

```
{'c', 'a', 'b', 'd'}
frozenset({'c', 'e', 'b', 'a', 'd'})
```

141)

```
print(hash('python'))
print(hash('n'))
print(hash('a'))
print(hash('sdmcet'))
print(hash('lol'))
```

```
#2sd19ec009
print(hash('python'))
print(hash('n'))
print(hash('a'))
print(hash('sdmcet'))
print(hash('lol'))
```

```
-6527180022189089663
-3107355805124117254
-8632883097262667572
-9042608897187856618
3389976898487339059
```

142)

```
a9=99
b9=90
if a9==b9==34:
  print('a9 and b9 equal to 34')
else:
  print('no')
x9=85
if -1<x9<1:
  print('x9 is between -1 and 1')
else:
  print("NO")
```

```
#2sd19ec009
a9=99
b9=90
if a9==b9==34:
  print('a9 and b9 equal to 34')
else:
  print('no')
x9=85
if -1<x9<1:
  print('x9 is between -1 and 1')
else:
  print("NO")
```

```
no
NO
```

```
143)
x9=2
def const(X9):
  return 1.
def lin(x9):
  return x9
def square(x9):
  return x9**2
def cube(x9):
  return x9**3
flist=[const(x9),lin(x9),square(x9),cube(x9)]
print("x9:",x9)
print("const,lin,square,cube")
print("flist:",flist)
print("flist[2]:",flist[2])
```

```
#2sd19ec009
x9=2
def const(X9):
  return 1.
def lin(x9):
  return x9
def square(x9):
  return x9**2
def cube(x9):
  return x9**3
flist=[const(x9),lin(x9),square(x9),cube(x9)]
print("x9:",x9)
print("const,lin,square,cube")
print("flist:",flist)
print("flist[2]:",flist[2])
```

```
x9: 2
const,lin,square,cube
flist: [1.0, 2, 4, 8]
flist[2]: 4
```

144)

```
flist=[lambda x9:1,
       lambda x9:x9,
       lambda x9:x9**2,
       lambda x9:x9**3]
print("flist[0](4)",flist[0](4))
print("flist[1](4)",flist[1](4))
print("flist[2](4)",flist[2](4))
print("flist[3](4)",flist[3](4))
```

```
#2sd19ec009
flist=[lambda x9:1,
       lambda x9:x9,
       lambda x9:x9**2,
       lambda x9:x9**3]
print("flist[0](4)",flist[0](4))
print("flist[1](4)",flist[1](4))
print("flist[2](4)",flist[2](4))
print("flist[3](4)",flist[3](4))
```

```
flist[0](4) 1
flist[1](4) 4
flist[2](4) 16
flist[3](4) 64
```

145)

```
print("The horse raced past the barn fell.".split())
print(sorted("The horse raced past the barn fell.".split()))
print(sorted("The horse raced past the barn fell.".split(),key=str.lowe
r))
```

```
#2sd19ec009
print("The horse raced past the barn fell.".split())
print(sorted("The horse raced past the barn fell.".split()))
print(sorted("The horse raced past the barn fell.".split(),key=str.lower))
```

```
['The', 'horse', 'raced', 'past', 'the', 'barn', 'fell.']
['The', 'barn', 'fell.', 'horse', 'past', 'raced', 'the']
['barn', 'fell.', 'horse', 'past', 'raced', 'The', 'the']
['barn', 'fell.', 'horse', 'past', 'raced', 'The', 'the']
```

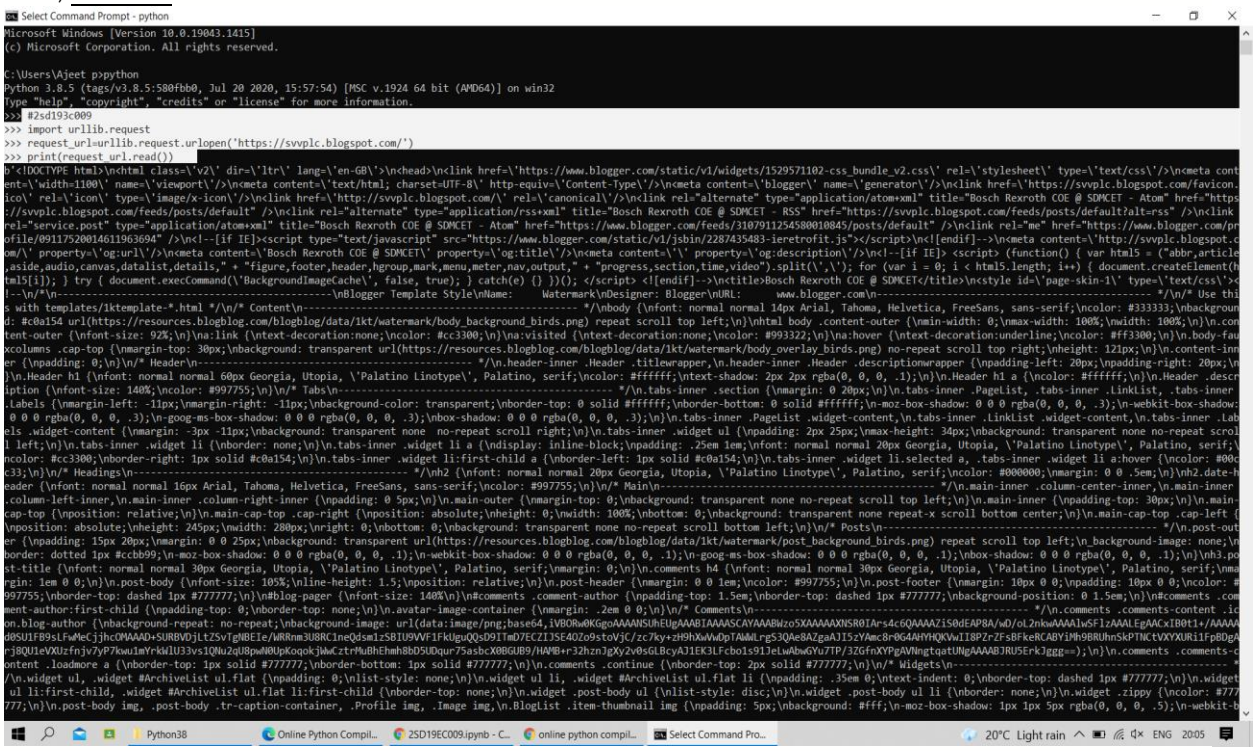
```
#2sd19ec009
import random
print(random.random())
print(random.random())
```

```
146)
import random
print(random.random())
print(random.random())
print(random.randint(0,5))
print(random.random())
random.seed(42)
print(random.random())
print(random.random())
random.seed(42)
print(random.random())
print(random.random())
random.seed(66)
print(random.randint(1,5))
s9=random.uniform(5,14)
v9=random.uniform(7,15)
print("random float uniform(5,14: )",s9)
print("random float uniform(7,15: )",v9)
print(random.choice('computer'))
print(random.choice([12,13,14,15,16,17,18,19]))
numbers=[12,13,14,15,16,17,18,19]
random.shuffle(numbers)
print("shuffeled",numbers)
cl=["newtork","los angles","chicago","houston","new york"]
print("random cl:",random.choice(cl))
print("random cl:",random.sample(cl,2))
```

```
#2sd19ec009
import random
print(random.random())
print(random.random())
print(random.randint(0,5))
print(random.random())
random.seed(42)
print(random.random())
print(random.random())
random.seed(42)
print(random.random())
print(random.random())
random.seed(66)
print(random.randint(1,5))
s9=random.uniform(5,14)
v9=random.uniform(7,15)
print("random float uniform(5,14: )",s9)
print("random float uniform(7,15: )",v9)
print(random.choice('computer'))
print(random.choice([12,13,14,15,16,17,18,19]))
numbers=[12,13,14,15,16,17,18,19]
random.shuffle(numbers)
print("shuffeled",numbers)
cl=["newtork","los angles","chicago","houston","new york"]
print("random cl:",random.choice(cl))
print("random cl:",random.sample(cl,2))
```

```
0.12795722422850875
0.7842725419704123
0
0.7144578796877428
0.6394267984578837
0.025010755222666936
0.6394267984578837
0.025010755222666936
1
random float uniform(5,14: ) 7.807878046566916
random float uniform(7,15: ) 14.390290334597216
r
16
shuffeled [14, 13, 17, 15, 12, 19, 18, 16]
random cl: new york
random cl: ['los angles', 'newtork']
```

147) ***urllib***




```
147)
import datetime
x9=datetime.datetime.now()
print(x9)
&
148)
x9=datetime.datetime.now()
print(x9)
print("year:",x9.year)
print("day:",x9.strftime("%A"))
print("monnth name:",x9.strftime("%B"))
print("year full year:",x9.strftime("%Y"))
print("week number:",x9.strftime("%U"))
```

```
#2sd19ec009
import datetime
x9=datetime.datetime.now()
print(x9)
```

```
2022-01-01 16:08:22.172349
```

```
#2SD19EC009
x9=datetime.datetime.now()
print(x9)
print("year:",x9.year)
print("day:",x9.strftime("%A"))
print("monnth name:",x9.strftime("%B"))
print("year full year:",x9.strftime("%Y"))
print("week number:",x9.strftime("%U"))
```

```
2022-01-01 16:12:13.625054
year: 2022
day: Saturday
monnth name: January
year full year: 2022
week number: 00
```

149)

```
class Bank_Account:
    def __init__(self):
        self.balance=0
        print("Welcome to Deposit & Withdrawal Machine!")

    def deposit(self):
        amount=float(input("Enter amount to be deposited: "))
        self.balance += amount
        print("Amount Deposited: ",amount)
    def withdraw(self):
        amount = float(input("Enter amount to withdraw: "))
        if self.balance>=amount:
            self.balance-=amount
            print("You withdraw: ",amount)
        else:
            print("Insufficient balance ")
    def display(self):
        print("Net Available Balance=",self.balance)
#creating an object of class
s = Bank_Account()

#calling functions with that class
s.deposit()
s.withdraw()
s.display()
```

```
#2sd19ec009
class Bank_Account:
    def __init__(self):
        self.balance=0
        print("Welcome to Deposit & Withdrawal Machine!")

    def deposit(self):
        amount=float(input("Enter amount to be deposited: "))
        self.balance += amount
        print("Amount Deposited: ",amount)
    def withdraw(self):
        amount = float(input("Enter amount to withdraw: "))
        if self.balance>=amount:
            self.balance-=amount
            print("You withdraw: ",amount)
        else:
            print("Insufficient balance ")
    def display(self):
        print("Net Available Balance=",self.balance)
#creating an object of class
s = Bank_Account()

#calling functions with that class
s.deposit()
s.withdraw()
s.display()
```

```
Welcome to Deposit & Withdrawal Machine!
Enter amount to be deposited: 100000
Amount Deposited:  100000.0
Enter amount to withdraw: 0
You withdraw:  0.0
Net Available Balance= 100000.0
```

150)

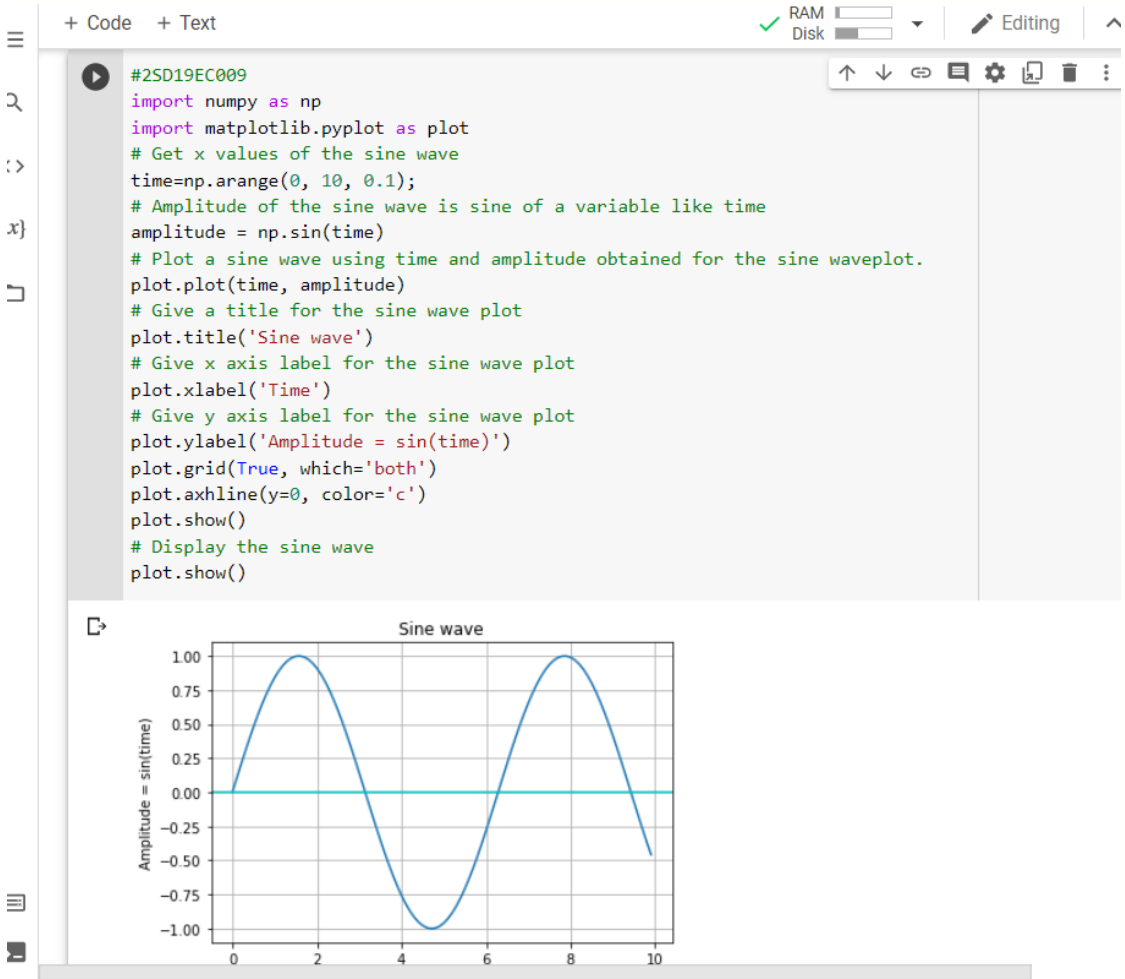
```
class Bank_Account:
  currency='$'
  def __init__(self,costumer,account_number,Balance=0):
    self.costumer=costumer
    self.account_number=account_number
    self.balance=Balance
  def deposit(self,amount):
    if amount>0:
      self.balance+=amount
    else:
      print("invalid deposit amount:",amount)
  def withdraw(self,amount):
    if amount>0:
      if amount>self.balance:
        print("insufficient funds")
      else:
        self.balance-=amount
    else:
      print("invalid without amount:",amount)

my_account=Bank_Account("ajeet",50000)
print("my account number:",my_account.account_number)
my_account.deposit(505050)
my_account.deposit(45858)
print("my account deposit",my_account.balance)
```

```python
#2sd19ec009
class Bank_Account:
  currency='$'
  def __init__(self,costumer,account_number,Balance=0):
    self.costumer=costumer
    self.account_number=account_number
    self.balance=Balance
  def deposit(self,amount):
    if amount>0:
      self.balance+=amount
    else:
      print("invalid deposit amount:",amount)
  def withdraw(self,amount):
    if amount>0:
      if amount>self.balance:
        print("insufficient funds")
      else:
        self.balance-=amount
    else:
      print("invalid without amount:",amount)

my_account=Bank_Account("ajeet",50000)
print("my account number:",my_account.account_number)
my_account.deposit(505050)
my_account.deposit(45858)
print("my account deposit",my_account.balance)
```

```
my account number: 50000
my account deposit 550908
```

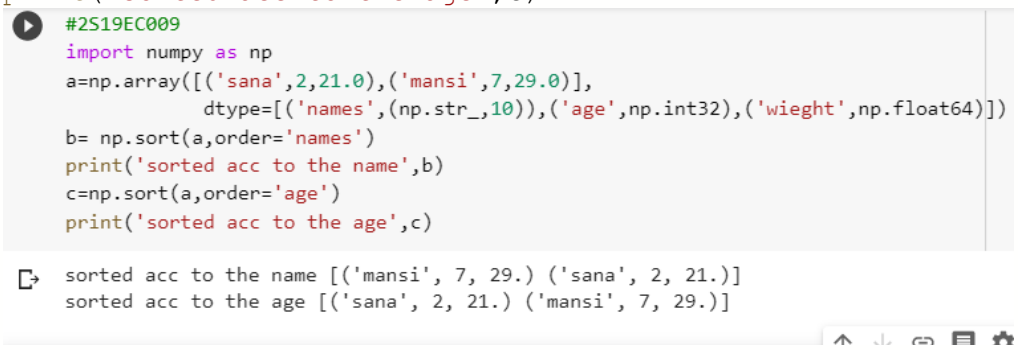
151)

```
import numpy as np
import matplotlib.pyplot as plot
# Get x values of the sine wave
time=np.arange(0, 10, 0.1);
# Amplitude of the sine wave is sine of a variable like time
amplitude = np.sin(time)
# Plot a sine wave using time and amplitude obtained for the sine wavep
lot.
plot.plot(time, amplitude)
# Give a title for the sine wave plot
plot.title('Sine wave')
# Give x axis label for the sine wave plot
plot.xlabel('Time')
# Give y axis label for the sine wave plot
plot.ylabel('Amplitude = sin(time)')
plot.grid(True, which='both')
plot.axhline(y=0, color='c')
plot.show()
# Display the sine wave
plot.show()
```



152) #2S19EC009

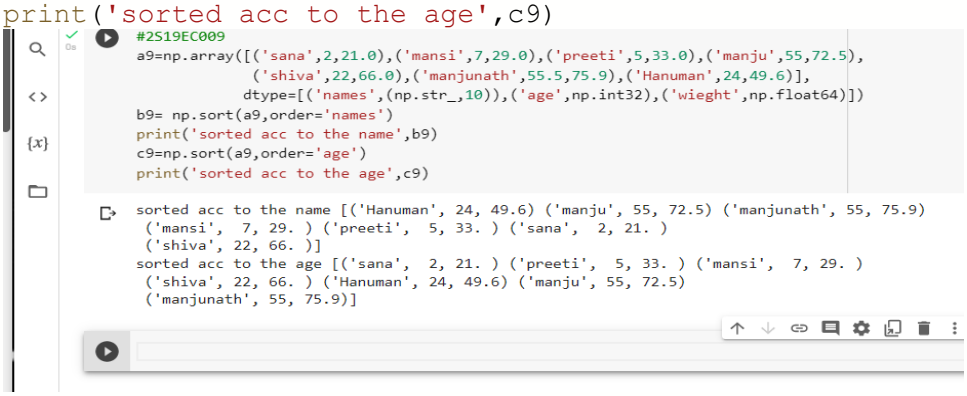
```
import numpy as np
a=np.array([('sana',2,21.0),('mansi',7,29.0)],
           dtype=[('names',(np.str_,10)),('age',np.int32),('wieght',np
.float64)])
b= np.sort(a,order='names')
print('sorted acc to the name',b)
c=np.sort(a,order='age')
print('sorted acc to the age',c)
```



```
#2S19EC009
import numpy as np
a=np.array([('sana',2,21.0),('mansi',7,29.0)],
           dtype=[('names',(np.str_,10)),('age',np.int32),('wieght',np.float64)])
b= np.sort(a,order='names')
print('sorted acc to the name',b)
c=np.sort(a,order='age')
print('sorted acc to the age',c)

sorted acc to the name [('mansi', 7, 29.) ('sana', 2, 21.)]
sorted acc to the age [('sana', 2, 21.) ('mansi', 7, 29.)]
```

153) #2S19EC009

```
a9=np.array([('sana',2,21.0),('mansi',7,29.0),('preeti',5,33.0),('manju
',55,72.5),
             ('shiva',22,66.0),('manjunath',55.5,75.9),('Hanuman',24,49
.6)],
           dtype=[('names',(np.str_,10)),('age',np.int32),('wieght',np
.float64)])
b9= np.sort(a9,order='names')
print('sorted acc to the name',b9)
c9=np.sort(a9,order='age')
print('sorted acc to the age',c9)
```



```
#2S19EC009
a9=np.array([('sana',2,21.0),('mansi',7,29.0),('preeti',5,33.0),('manju',55,72.5),
             ('shiva',22,66.0),('manjunath',55.5,75.9),('Hanuman',24,49.6)],
           dtype=[('names',(np.str_,10)),('age',np.int32),('wieght',np.float64)])
b9= np.sort(a9,order='names')
print('sorted acc to the name',b9)
c9=np.sort(a9,order='age')
print('sorted acc to the age',c9)

sorted acc to the name [('Hanuman', 24, 49.6) ('manju', 55, 72.5) ('manjunath', 55, 75.9)
 ('mansi',  7, 29. ) ('preeti',  5, 33. ) ('sana',  2, 21. )
 ('shiva', 22, 66. )]
sorted acc to the age [('sana',  2, 21. ) ('preeti',  5, 33. ) ('mansi',  7, 29. )
 ('shiva', 22, 66. ) ('Hanuman', 24, 49.6) ('manju', 55, 72.5)
 ('manjunath', 55, 75.9)]
```

154)

```
a9=np.array([1,0,-3])
b9=np.array([2,-2,5])
print('a9=',a9)
print('b9=',b9)
print('a9.dot(b9)=',a9.dot(b9))
print('np.cross(a9,b9=)',np.cross(a9,b9))
```

```
[22] #2sd19ec009
     a9=np.array([1,0,-3])
     b9=np.array([2,-2,5])
     print('a9=',a9)
     print('b9=',b9)
     print('a9.dot(b9)=',a9.dot(b9))
     print('np.cross(a9,b9=)',np.cross(a9,b9))

     a9= [ 1  0 -3]
     b9= [ 2 -2  5]
     a9.dot(b9)= -13
     np.cross(a9,b9=) [ -6 -11  -2]
```

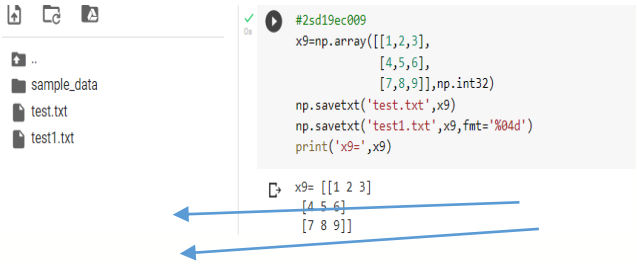
155)

```
a9=np.array([[1],[0],[-3]])
b9=np.array([[2],[-2],[5]])
print('a9',a9)
c9=a9.ravel().dot(b9.ravel())
print('dot product c9=',c9)
```

```
[25] #2sd19ec009
     a9=np.array([[1],[0],[-3]])
     b9=np.array([[2],[-2],[5]])
     print('a9',a9)
     c9=a9.ravel().dot(b9.ravel())
     print('dot product c9=',c9)

     a9 [[ 1]
      [ 0]
      [-3]]
     dot product c9= -13
```

156)

```
x9=np.array([[1,2,3],
             [4,5,6],
             [7,8,9]],np.int32)
np.savetxt('test.txt',x9)
np.savetxt('test1.txt',x9,fmt='%04d')
print('x9=',x9)
```



157)

```
a9=np.linspace(1,12,12)
print(a9)
print('np.min(a9)=',np.min(a9))
print('np.max(a9)=',np.max(a9))
```

```
#2SD19EC009
a9=np.linspace(1,12,12)
print(a9)
print('np.min(a9)=',np.min(a9))
print('np.max(a9)=',np.max(a9))
```

```
[ 1.  2.  3.  4.  5.  6.  7.  8.  9. 10. 11. 12.]
np.min(a9)= 1.0
np.max(a9)= 12.0
```

158)

```
a9=np.sqrt(np.linspace(-2,2,4))
print('a9=',a9)
print('np.min(a9)=',np.min(a9))
print('np.max(a9)=',np.max(a9))
print('np.nanmin(a9)=',np.nanmin(a9))
print('np.nanmax(a9)=',np.nanmax(a9))
print(np.argmin(a9))
print(np.argmax(a9))
print(np.nanargmin(a9))
print(np.nanargmax(a9))
```

```
#2SD19EC009
a9=np.sqrt(np.linspace(-2,2,4))
print('a9=',a9)
print('np.min(a9)=',np.min(a9))
print('np.max(a9)=',np.max(a9))
print('np.nanmin(a9)=',np.nanmin(a9))
print('np.nanmax(a9)=',np.nanmax(a9))
```

```
a9= [       nan        nan 0.81649658 1.41421356]
np.min(a9)= nan
np.max(a9)= nan
np.nanmin(a9)= 0.8164965809277259
np.nanmax(a9)= 1.4142135623730951
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:2: Runtimel
```

159)

```
b9=np.random.rand(10)
print(b9)
print('minimum=',min(b9))
print('maximun=',max(b9))
print('minimun=',np.min(b9))
print('maximun=',np.max(b9))
```

```
1] #2SD19EC009
   b9=np.random.rand(10)
   print(b9)
   print('minimum=',min(b9))
   print('maximun=',max(b9))
   print('minimun=',np.min(b9))
   print('maximun=',np.max(b9))
```

```
[0.46915177 0.65322811 0.85025464 0.67872416 0.3812748  0.710505
 0.31396956 0.11400019 0.0943749  0.91212639]
minimum= 0.09437490472103482
maximun= 0.9121263891329815
minimun= 0.09437490472103482
maximun= 0.9121263891329815
```

160)

```
m9=np.random.random((3,4))
print(m9)
print('sum=',m9.sum)
print('col min=',m9.min(axis=0))
print('row max=',m9.max(axis=1))
```

```
#2SD19EC009
m9=np.random.random((3,4))
print(m9)
print('sum=',m9.sum)
print('col min=',m9.min(axis=0))
print('row max=',m9.max(axis=1))
```

```
[[0.78103732 0.80436821 0.5836887  0.55707744]
 [0.34032746 0.44682172 0.75396193 0.4621472 ]
 [0.35743795 0.21726021 0.50963136 0.23249589]]
sum= <built-in method sum of numpy.ndarray object at 0x7f45c296e990>
col min= [0.34032746 0.21726021 0.50963136 0.23249589]
row max= [0.80436821 0.75396193 0.50963136]
```

161)

```
arr9=[20,2,7,1,35]
print('arr9: ',arr9)
print('50th percentile of arr9: ',np.percentile(arr9,50))
print('25th percentile of arr9: ',np.percentile(arr9,25))
print('75th percentile of arr9: ',np.percentile(arr9,75))
```

```
] #2sd19ec009
  arr9=[20,2,7,1,35]
  print('arr9: ',arr9)
  print('50th percentile of arr9: ',np.percentile(arr9,50))
  print('25th percentile of arr9: ',np.percentile(arr9,25))
  print('75th percentile of arr9: ',np.percentile(arr9,75))
```

```
arr9:  [20, 2, 7, 1, 35]
50th percentile of arr9:  7.0
25th percentile of arr9:  2.0
75th percentile of arr9:  20.0
```

162) #2sd193c009

```
ages=[50,54,58,56,52,57,11,2,5,3,95,75,85,98,99,47,52,14,25]
x9=np.percentile(ages,75)
print('percentile 75',x9)
x9=np.percentile(ages,100)
print('percentile 100',x9)
x9=np.percentile(ages,2)
print('percentile 2',x9)
```

```
#2sd193c009
ages=[50,54,58,56,52,57,11,2,5,3,95,75,85,98,99,47,52,14,25]
x9=np.percentile(ages,75)
print('percentile 75',x9)
x9=np.percentile(ages,100)
print('percentile 100',x9)
x9=np.percentile(ages,2)
print('percentile 2',x9)
```

```
percentile 75 66.5
percentile 100 99.0
percentile 2 2.3600000000000003
```

163)

```
print(np.average([1,2,3],weights=[1,0,1]))
print(np.average([1,2,3],weights=[2,1,1]))
print(np.average([1,2,3],weights=[1,1,1]))
print(np.average([1,2,3],weights=[7,2,1]))
```

```
#2sd19ec009
print(np.average([1,2,3],weights=[1,0,1]))
print(np.average([1,2,3],weights=[2,1,1]))
print(np.average([1,2,3],weights=[1,1,1]))
print(np.average([1,2,3],weights=[7,2,1]))
```

```
2.0
1.75
2.0
1.4
```

164)

```
import statistics
data=[11,12,14,15,16,1,78,25,46,48,85,100]
x9=statistics.mean(data)
print('mean=',x9)
x99=statistics.median(data)
print('median=',x99)
x0999=statistics.variance(data)
print('variance=',x0999)
x00999=statistics.stdev(data)
print('std',x00999)
```

```
#2sd19ec009
import statistics
data=[11,12,14,15,16,1,78,25,46,48,85,100]
x9=statistics.mean(data)
print('mean=',x9)
x99=statistics.median(data)
print('median=',x99)
x0999=statistics.variance(data)
print('variance=',x0999)
x00999=statistics.stdev(data)
print('std',x00999)
```

```
mean= 37.583333333333336
median= 20.5
variance= 1122.4469696969697
std 33.502939717239286
```

165)

```
Polynomial=np.polynomial.Polynomial
p=Polynomial([6,-5,1])
print('p=',p)
print('p.coef=',p.coef)
print('p(4)=',p(4))
x=np.linspace(-5,5,11)
print('x',x)
print('p(x)=',p(x))
```

```
#2sd19ec009
Polynomial=np.polynomial.Polynomial
p=Polynomial([6,-5,1])
print('p=',p)
print('p.coef=',p.coef)
print('p(4)=',p(4))
x=np.linspace(-5,5,11)
print('x',x)
print('p(x)=',p(x))
```

```
p= poly([ 6. -5.  1.])
p.coef= [ 6. -5.  1.]
p(4)= 2.0
x [-5. -4. -3. -2. -1.  0.  1.  2.  3.  4.  5.]
p(x)= [56. 42. 30. 20. 12.  6.  2.  0.  0.  2.  6.]
```

166)

```
Polynomial=np.polynomial.Polynomial
p=Polynomial([6,-5,1])
q=Polynomial([2,-3])
print('p+q=',p+q)
print('p-q=',p-q)
print('p*q=',p*q)
print('p//q=',p//q)
print('p%q=',p%q)
```

```
#2sd19ec009
Polynomial=np.polynomial.Polynomial
p=Polynomial([6,-5,1])
q=Polynomial([2,-3])
print('p+q=',p+q)
print('p-q=',p-q)
print('p*q=',p*q)
print('p//q=',p//q)
print('p%q=',p%q)
```

```
p+q= poly([ 8. -8.  1.])
p-q= poly([ 4. -2.  1.])
p*q= poly([ 12. -28.  17.  -3.])
p//q= poly([ 1.44444444 -0.33333333])
p%q= poly([3.11111111])
```

167)

```
a9=np.array([[0,-1],[1,-2]])
b9=np.array([[0,-1],[1,-2]])
print('array a9=\n',a9)
print('array a9*a9=\n',a9*a9)
print('array a9*3=\n',a9*3)
print('mtrx b9=\n',b9)
print('mtrx b9*2=\n',b9*2)
print('mtrx b9*3=\n',b9*3)
```

```
#2sd19ec009
a9=np.array([[0,-1],[1,-2]])
b9=np.array([[0,-1],[1,-2]])
print('array a9=\n',a9)
print('array a9*a9=\n',a9*a9)
print('array a9*3=\n',a9*3)
print('mtrx b9=\n',b9)
print('mtrx b9*2=\n',b9*2)
print('mtrx b9*3=\n',b9*3)
```

```
array a9=
 [[ 0 -1]
 [ 1 -2]]
array a9*a9=
 [[0 1]
 [1 4]]
array a9*3=
 [[ 0 -3]
 [ 3 -6]]
mtrx b9=
 [[ 0 -1]
 [ 1 -2]]
mtrx b9*2=
 [[ 0 -2]
 [ 2 -4]]
mtrx b9*3=
 [[ 0 -3]
 [ 3 -6]]
```

168)

```
r9=np.matrix('50 0 -30; 0 40 -20;-30 -21 100')
v9=np.matrix('80;80;0')
i9=np.linalg.inv(r9)*v9
print(i9)
```

```
74] #2sd19ec009
    r9=np.matrix('50 0 -30; 0 40 -20;-30 -21 100')
    v9=np.matrix('80;80;0')
    i9=np.linalg.inv(r9)*v9
    print(i9)
```

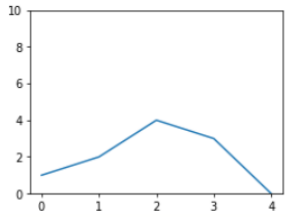
```
[[2.35524476]
 [2.62937063]
 [1.25874126]]
```

169)

```
w = 4
h = 3
d = 70
plt.figure(figsize=(w, h), dpi=d)
x = [1, 2, 4, 3, 0]
plt.ylim(0, 10)
plt.plot(x)
```

```
#2sd19ec009
w = 4
h = 3
d = 70
plt.figure(figsize=(w, h), dpi=d)
x = [1, 2, 4, 3, 0]
plt.ylim(0, 10)
plt.plot(x)
```

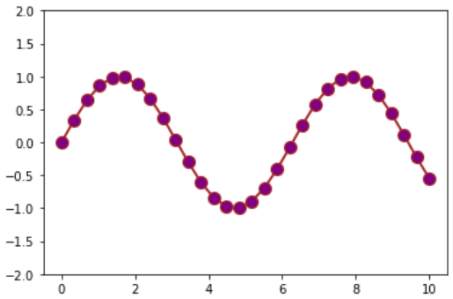
```
[<matplotlib.lines.Line2D at 0x7f45b9818290>]
```



170)

```
import numpy as np
import pylab as plt
x=np.linspace(0,10,30)
y=np.sin(x)
plt.plot(x,y,'-o',color='brown',markersize=10,linewidth='2'
,markerfacecolor='purple',markeredgewidth='1.2')
plt.ylim(-2,2,1,1)
plt.show()
```

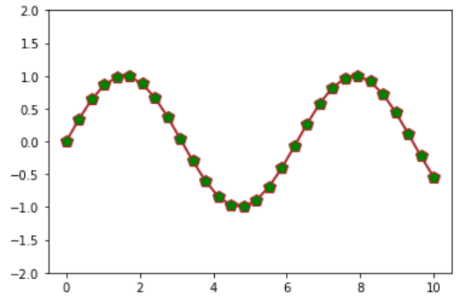
```
#2SD19EC009
import numpy as np
import pylab as plt
x=np.linspace(0,10,30)
y=np.sin(x)
plt.plot(x,y,'-o',color='brown',markersize=10,linewidth='2'
,markerfacecolor='purple',markeredgewidth='1.2')
plt.ylim(-2,2,1,1)
plt.show()
```



171)

```
import numpy as np
import pylab as plt
x=np.linspace(0,10,30)
y=np.sin(x)
plt.plot(x,y,'-p',color='brown',markersize=10,linewidth='2'
,markerfacecolor='green',markeredgewidth='1.2')
plt.ylim(-2,2,1,1)
plt.show()
```

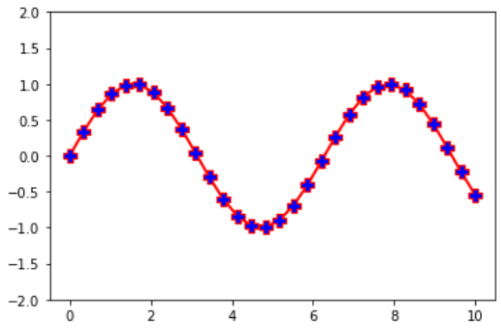
```
#2SD19EC009
import numpy as np
import pylab as plt
x=np.linspace(0,10,30)
y=np.sin(x)
plt.plot(x,y,'-p',color='brown',markersize=10,linewidth='2'
,markerfacecolor='green',markeredgewidth='1.2')
plt.ylim(-2,2,1,1)
plt.show()
```



172)

```
import numpy as np
import pylab as plt
x=np.linspace(0,10,30)
y=np.sin(x)
plt.plot(x,y,'-P',color='red',markersize=10,linewidth='2'
,markerfacecolor='blue',markeredgewidth='1.2')
plt.ylim(-2,2,1,1)
plt.show()
```

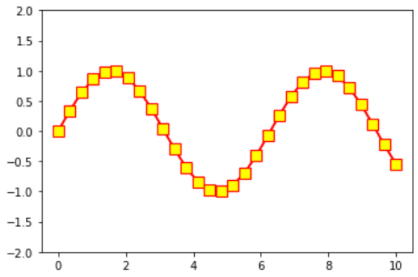
```
#2SD19EC009
import numpy as np
import pylab as plt
x=np.linspace(0,10,30)
y=np.sin(x)
plt.plot(x,y,'-P',color='red',markersize=10,linewidth='2
,markerfacecolor='blue',markeredgewidth='1.2')
plt.ylim(-2,2,1,1)
plt.show()
```



173)

```
import numpy as np
import pylab as plt
x=np.linspace(0,10,30)
y=np.sin(x)
plt.plot(x,y,'-s',color='red',markersize=10,linewidth='2'
,markerfacecolor='yellow',markeredgewidth='1.2')
plt.ylim(-2,2,1,1)
plt.show()
```

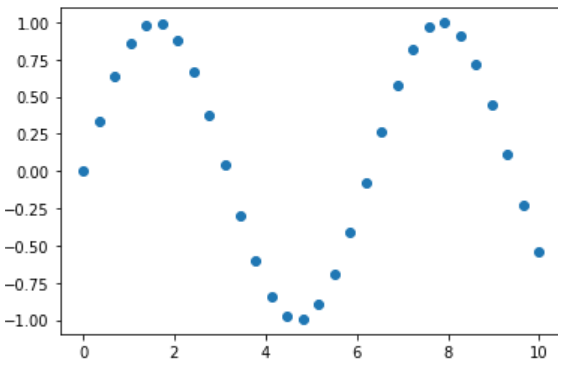
```
#2SD19EC009
import numpy as np
import pylab as plt
x=np.linspace(0,10,30)
y=np.sin(x)
plt.plot(x,y,'-s',color='red',markersize=10,linewidth='2'
,markerfacecolor='yellow',markeredgewidth='1.2')
plt.ylim(-2,2,1,1)
plt.show()
```



174)

```
x9=np.linspace(0,10,30)
y9=np.sin(x9)
plt.scatter(x9,y9,marker='o')
plt.show()
```

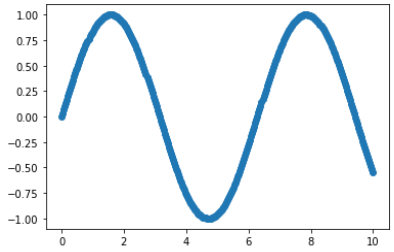
```
2sd19ec009
9=np.linspace(0,10,30)
9=np.sin(x9)
lt.scatter(x9,y9,marker='o')
lt.show()
```



175

```
x9=np.linspace(0,10,300)
y9=np.sin(x9)
plt.scatter(x9,y9,marker='o')
plt.show()
```

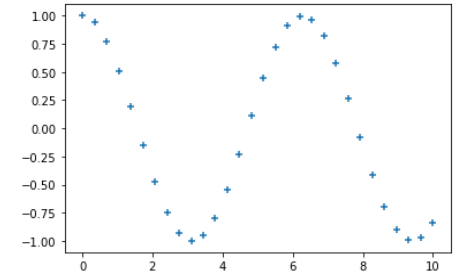
```
#2sd19ec009
x9=np.linspace(0,10,300)
y9=np.sin(x9)
plt.scatter(x9,y9,marker='o')
plt.show()
```



176)

```
x9=np.linspace(0,10,30)
y9=np.cos(x9)
plt.scatter(x9,y9,marker='+')
plt.show()
```

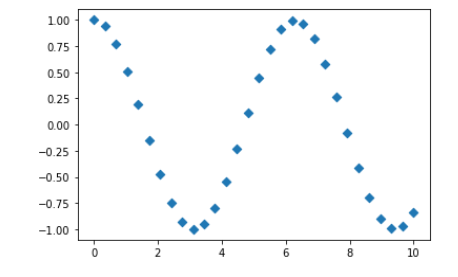
```
] #2sd19ec009
  x9=np.linspace(0,10,30)
  y9=np.cos(x9)
  plt.scatter(x9,y9,marker='+')
  plt.show()
```



177)

```
x9=np.linspace(0,10,30)
y9=np.cos(x9)
plt.scatter(x9,y9,marker='D')
plt.show()
```

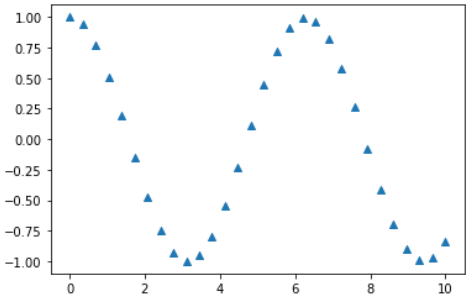
```
89] #2sd19ec009
    x9=np.linspace(0,10,30)
    y9=np.cos(x9)
    plt.scatter(x9,y9,marker='D')
    plt.show()
```



178)

```
x9=np.linspace(0,10,30)
y9=np.cos(x9)
plt.scatter(x9,y9,marker='^')
plt.show()
```

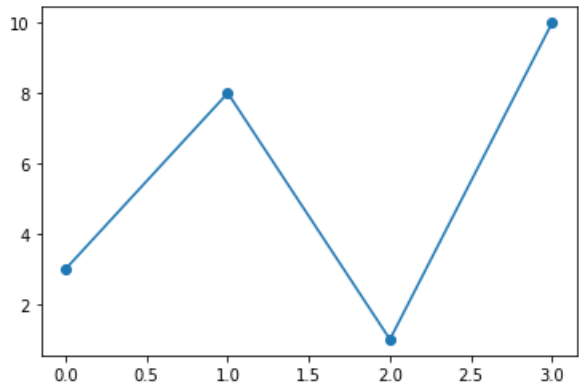
```
#2sd19ec009
x9=np.linspace(0,10,30)
y9=np.cos(x9)
plt.scatter(x9,y9,marker='^')
plt.show()
```



179)

```
import matplotlib.pyplot as plt
import numpy as np
ypoints = np.array([3, 8, 1, 10])
plt.plot(ypoints, marker = 'o')
plt.show()
```

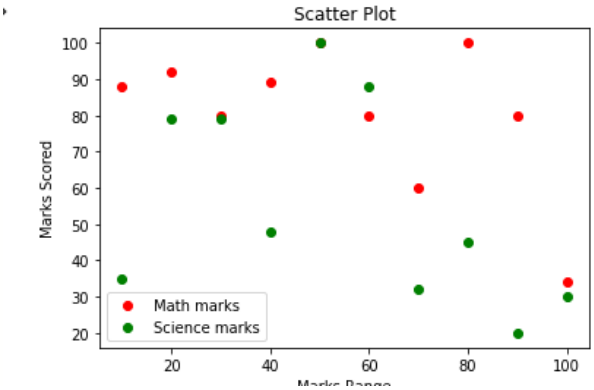
```
#2sd19ec009
import matplotlib.pyplot as plt
import numpy as np
ypoints = np.array([3, 8, 1, 10])
plt.plot(ypoints, marker = 'o')
plt.show()
```



180)

```
import matplotlib.pyplot as plt
math_marks = [88, 92, 80, 89, 100, 80, 60, 100, 80, 34]
science_marks = [35, 79, 79, 48, 100, 88, 32, 45, 20, 30]
marks_range = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]
plt.scatter(marks_range, math_marks, label='Math marks', color='r')
plt.scatter(marks_range, science_marks, label='Science marks', color='g
')
plt.title('Scatter Plot')
plt.xlabel('Marks Range')
plt.ylabel('Marks Scored')
plt.legend()
plt.show()
```

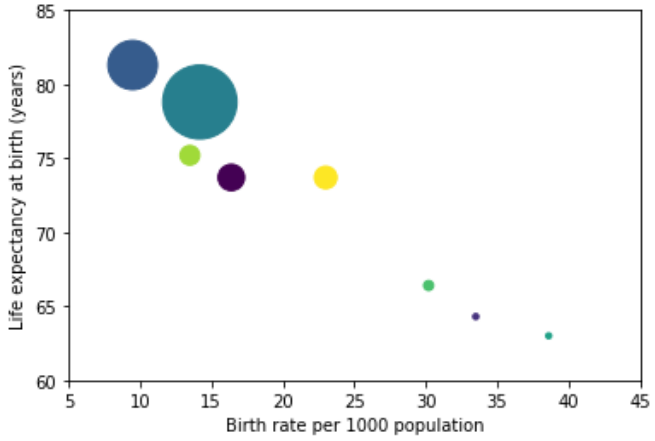
```
#2sd19ec009
import matplotlib.pyplot as plt
math_marks = [88, 92, 80, 89, 100, 80, 60, 100, 80, 34]
science_marks = [35, 79, 79, 48, 100, 88, 32, 45, 20, 30]
marks_range = [10, 20, 30, 40, 50, 60, 70, 80, 90, 100]
plt.scatter(marks_range, math_marks, label='Math marks', color='r')
plt.scatter(marks_range, science_marks, label='Science marks', color='g')
plt.title('Scatter Plot')
plt.xlabel('Marks Range')
plt.ylabel('Marks Scored')
plt.legend()
plt.show()
```



181)

```
import numpy as np
import matplotlib.pyplot as plt
countries = ['Brazil', 'Madagascar', 'S. Korea', 'United States',
             'Ethiopia', 'Pakistan', 'China', 'Belize']
birth_rate = [16.4, 33.5, 9.5, 14.2, 38.6, 30.2, 13.5, 23.0]
life_expectancy = [73.7, 64.3, 81.3, 78.8, 63.0, 66.4, 75.2, 73.7]
GDP = np.array([4800, 240, 16700, 37700, 230, 670, 2640, 3490])

fig = plt.figure()
ax = fig.add_subplot(111)
colours = range(len(countries))
ax.scatter(birth_rate, life_expectancy, c=colours, s=GDP/20)
ax.set_xlim(5, 45)
ax.set_ylim(60, 85)
ax.set_xlabel('Birth rate per 1000 population')
ax.set_ylabel('Life expectancy at birth (years)')
plt.show()
```

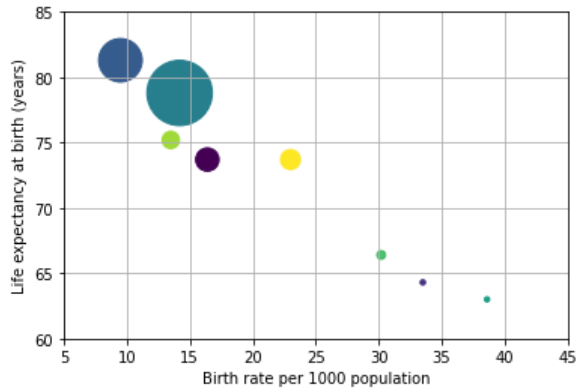
```
#2sd19ec009
import numpy as np
import matplotlib.pyplot as plt
countries = ['Brazil', 'Madagascar', 'S. Korea', 'United States',
             'Ethiopia', 'Pakistan', 'China', 'Belize']
birth_rate = [16.4, 33.5, 9.5, 14.2, 38.6, 30.2, 13.5, 23.0]
life_expectancy = [73.7, 64.3, 81.3, 78.8, 63.0, 66.4, 75.2, 73.7]
GDP = np.array([4800, 240, 16700, 37700, 230, 670, 2640, 3490])

fig = plt.figure()
ax = fig.add_subplot(111)
colours = range(len(countries))
ax.scatter(birth_rate, life_expectancy, c=colours, s=GDP/20)
ax.set_xlim(5, 45)
ax.set_ylim(60, 85)
ax.set_xlabel('Birth rate per 1000 population')
ax.set_ylabel('Life expectancy at birth (years)')
plt.show()
```



181)

```
import numpy as np
import matplotlib.pyplot as plt
countries = ['Brazil', 'Madagascar', 'S. Korea', 'United States',
             'Ethiopia', 'Pakistan', 'China', 'Belize']
birth_rate = [16.4, 33.5, 9.5, 14.2, 38.6, 30.2, 13.5, 23.0]
life_expectancy = [73.7, 64.3, 81.3, 78.8, 63.0, 66.4, 75.2, 73.7]
GDP = np.array([4800, 240, 16700, 37700, 230, 670, 2640, 3490])

fig = plt.figure()
ax = fig.add_subplot(111)
colours = range(len(countries))
ax.scatter(birth_rate, life_expectancy, c=colours, s=GDP/20)
ax.set_xlim(5, 45)
ax.set_ylim(60, 85)
ax.set_xlabel('Birth rate per 1000 population')
ax.set_ylabel('Life expectancy at birth (years)')
ax.grid()
plt.show()
```

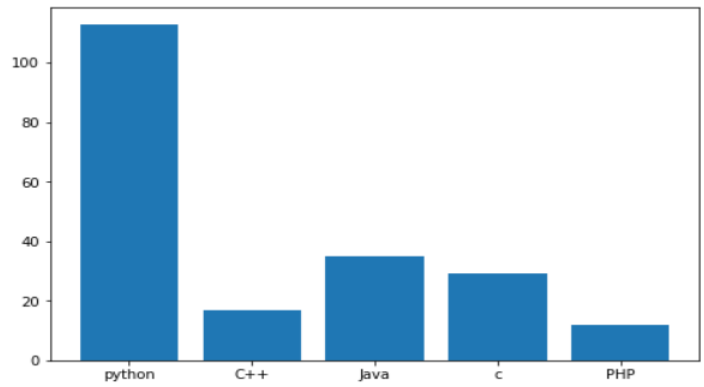
```
] #2sd19ec009
  import numpy as np
  import matplotlib.pyplot as plt
  countries = ['Brazil', 'Madagascar', 'S. Korea', 'United States',
               'Ethiopia', 'Pakistan', 'China', 'Belize']
  birth_rate = [16.4, 33.5, 9.5, 14.2, 38.6, 30.2, 13.5, 23.0]
  life_expectancy = [73.7, 64.3, 81.3, 78.8, 63.0, 66.4, 75.2, 73.7]
  GDP = np.array([4800, 240, 16700, 37700, 230, 670, 2640, 3490])

  fig = plt.figure()
  ax = fig.add_subplot(111)
  colours = range(len(countries))
  ax.scatter(birth_rate, life_expectancy, c=colours, s=GDP/20)
  ax.set_xlim(5, 45)
  ax.set_ylim(60, 85)
  ax.set_xlabel('Birth rate per 1000 population')
  ax.set_ylabel('Life expectancy at birth (years)')
  ax.grid()
  plt.show()
```



182)

```
import matplotlib.pyplot as plt
fig = plt.figure()
ax = fig.add_axes([0,0,1,1])
langs = ['python', 'C++', 'Java', 'c', 'PHP']
students = [113,17,35,29,12]
ax.bar(langs,students)
plt.show()
```

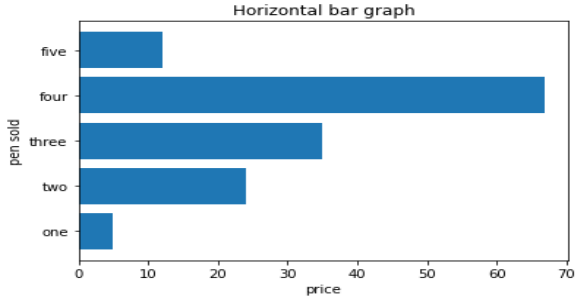
```
#2sd19ec009
import matplotlib.pyplot as plt
fig = plt.figure()
ax = fig.add_axes([0,0,1,1])
langs = ['python', 'C++', 'Java', 'c', 'PHP']
students = [113,17,35,29,12]
ax.bar(langs,students)
plt.show()
```



183)

```
import matplotlib.pyplot as plt
y=['one', 'two', 'three', 'four', 'five']
x=[5,24,35,67,12]
plt.barh(y, x)
plt.ylabel("pen sold")
plt.xlabel("price")
plt.title("Horizontal bar graph")
plt.show()
```

```
#2sd19ec009
import matplotlib.pyplot as plt
y=['one', 'two', 'three', 'four', 'five']
x=[5,24,35,67,12]
plt.barh(y, x)
plt.ylabel("pen sold")
plt.xlabel("price")
plt.title("Horizontal bar graph")
plt.show()
```

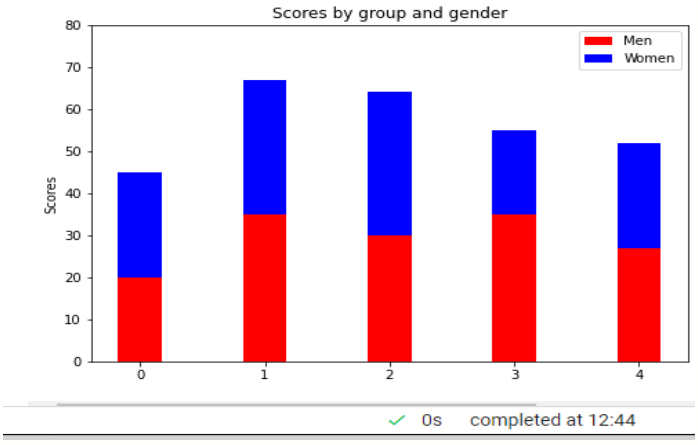


184)

```
import numpy as np
import matplotlib.pyplot as plt
N = 5
menMeans = (20, 35, 30, 35, 27)
womenMeans = (25, 32, 34, 20, 25)
ind = np.arange(N) # the x locations for the groups
width = 0.35
fig = plt.figure()
ax = fig.add_axes([0,0,1,1])
ax.bar(ind, menMeans, width, color='r')
ax.bar(ind, womenMeans, width,bottom=menMeans, color='b')
ax.set_ylabel('Scores')
ax.set_title('Scores by group and gender')
ax.set_xticks(ind, ('G1', 'G2', 'G3', 'G4', 'G5'))
ax.set_yticks(np.arange(0, 81, 10))
ax.legend(labels=['Men', 'Women'])
plt.show()
```

```
#2sd19ec009
import numpy as np
import matplotlib.pyplot as plt
N = 5
menMeans = (20, 35, 30, 35, 27)
womenMeans = (25, 32, 34, 20, 25)
ind = np.arange(N) # the x locations for the groups
width = 0.35
fig = plt.figure()
ax = fig.add_axes([0,0,1,1])
ax.bar(ind, menMeans, width, color='r')
ax.bar(ind, womenMeans, width,bottom=menMeans, color='b')
ax.set_ylabel('Scores')
ax.set_title('Scores by group and gender')
ax.set_xticks(ind, ('G1', 'G2', 'G3', 'G4', 'G5'))
ax.set_yticks(np.arange(0, 81, 10))
ax.legend(labels=['Men', 'Women'])
plt.show()
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1
```



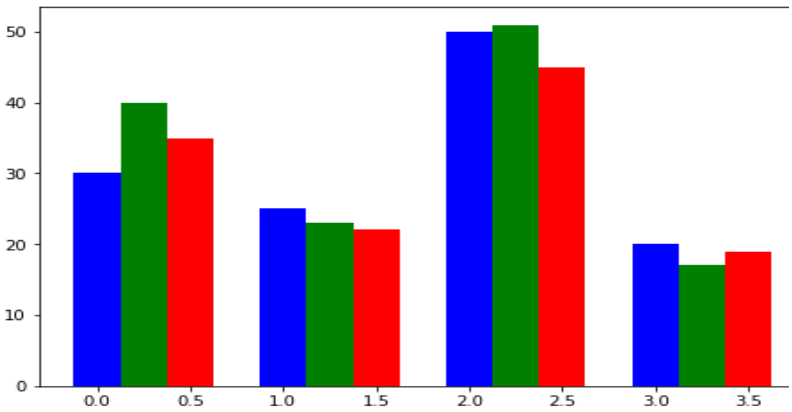

```
✓ 0s completed at 12:44
```

185)

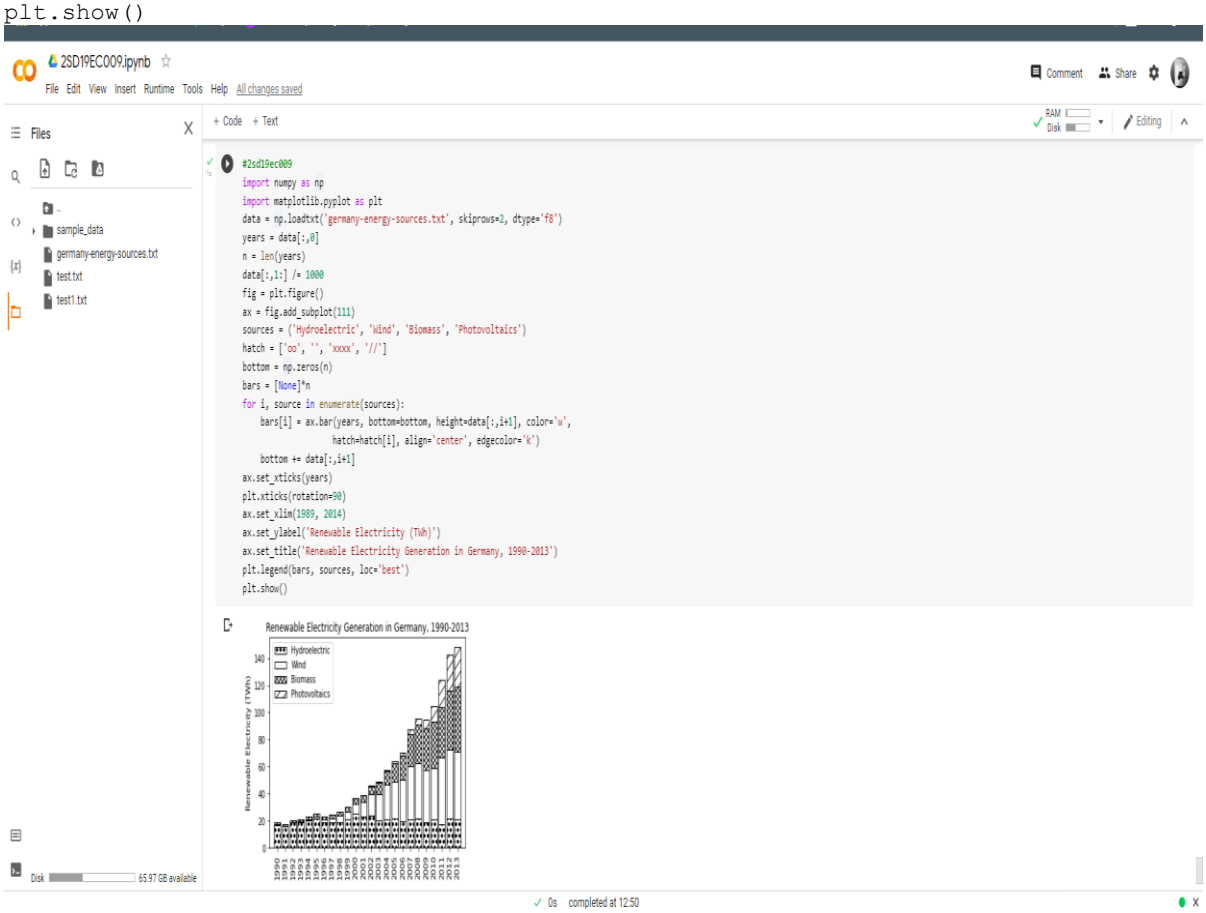
```
import matplotlib.pyplot as plt
data = [[30, 25, 50, 20],
[40, 23, 51, 17],
[35, 22, 45, 19]]
X = np.arange(4)
fig = plt.figure()
ax = fig.add_axes([0,0,1,1])
ax.bar(X + 0.00, data[0], color = 'b', width = 0.25)
ax.bar(X + 0.25, data[1], color = 'g', width = 0.25)
ax.bar(X + 0.50, data[2], color = 'r', width = 0.25)
```

```
2sd19ec009
import matplotlib.pyplot as plt
data = [[30, 25, 50, 20],
40, 23, 51, 17],
35, 22, 45, 19]]
 = np.arange(4)
ig = plt.figure()
x = fig.add_axes([0,0,1,1])
x.bar(X + 0.00, data[0], color = 'b', width = 0.25)
x.bar(X + 0.25, data[1], color = 'g', width = 0.25)
x.bar(X + 0.50, data[2], color = 'r', width = 0.25)
```

```
BarContainer object of 4 artists>
```



```
186)
import numpy as np
import matplotlib.pyplot as plt
data = np.loadtxt('germany-energy-sources.txt', skiprows=2, dtype='f8')
years = data[:,0]
n = len(years)
data[:,1:] /= 1000
fig = plt.figure()
ax = fig.add_subplot(111)
sources = ('Hydroelectric', 'Wind', 'Biomass', 'Photovoltaics')
hatch = ['oo', '', 'xxxx', '//']
bottom = np.zeros(n)
bars = [None]*n
for i, source in enumerate(sources):
    bars[i] = ax.bar(years, bottom=bottom, height=data[:,i+1], color='w',
                     hatch=hatch[i], align='center', edgecolor='k')
    bottom += data[:,i+1]
ax.set_xticks(years)
plt.xticks(rotation=90)
ax.set_xlim(1989, 2014)
ax.set_ylabel('Renewable Electricity (TWh)')
ax.set_title('Renewable Electricity Generation in Germany, 1990-2013')
plt.legend(bars, sources, loc='best')
plt.show()
```



186)

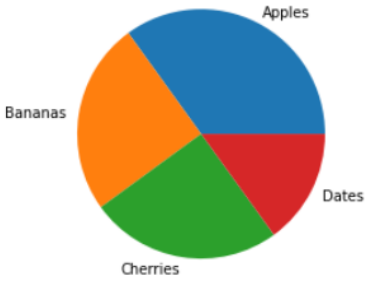
```
import matplotlib.pyplot as plt
import numpy as np
y = np.array([35, 25, 25, 15])
plt.pie(y)
plt.show()
```

```
02] #2sd19ec009
    import matplotlib.pyplot as plt
    import numpy as np
    y = np.array([35, 25, 25, 15])
    plt.pie(y)
    plt.show()
```

187)

```
y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]
plt.pie(y, labels = mylabels)
plt.show()
```

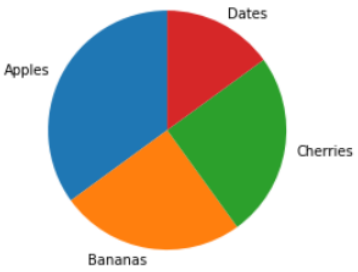
```
[103] #2sd19ec009
      y = np.array([35, 25, 25, 15])
      mylabels = ["Apples", "Bananas", "Cherries", "Dates"]
      plt.pie(y, labels = mylabels)
      plt.show()
```



188)

```
y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]
plt.pie(y, labels = mylabels, startangle = 90)
plt.show()
```

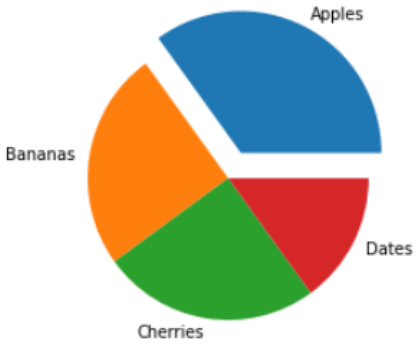
```
[105] #2sd19ec009
      y = np.array([35, 25, 25, 15])
      mylabels = ["Apples", "Bananas", "Cherries", "Dates"]
      plt.pie(y, labels = mylabels, startangle = 90)
      plt.show()
```



189)

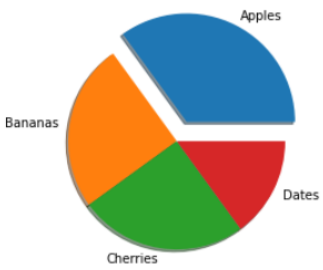
```
y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]
myexplode = [0.2, 0, 0, 0]
plt.pie(y, labels = mylabels, explode = myexplode)
plt.show()
```

```
#2sd19ec009
y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]
myexplode = [0.2, 0, 0, 0]
plt.pie(y, labels = mylabels, explode = myexplode)
plt.show()
```



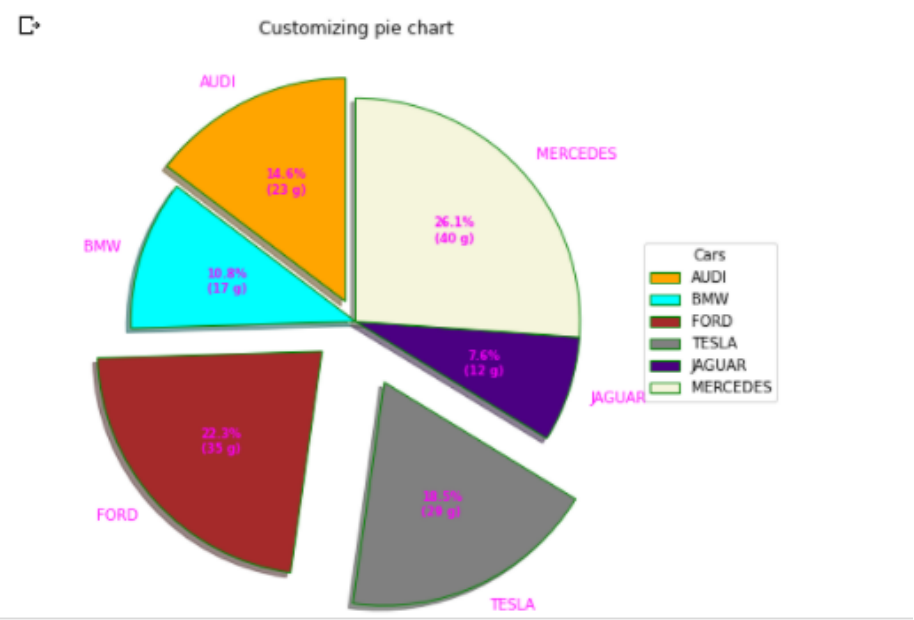
```
190)
y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]
myexplode = [0.2, 0, 0, 0]
plt.pie(y, labels = mylabels, explode = myexplode, shadow = True)
plt.show()
```

```
#2sd19ec009
y = np.array([35, 25, 25, 15])
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]
myexplode = [0.2, 0, 0, 0]
plt.pie(y, labels = mylabels, explode = myexplode, shadow = True)
plt.show()
```



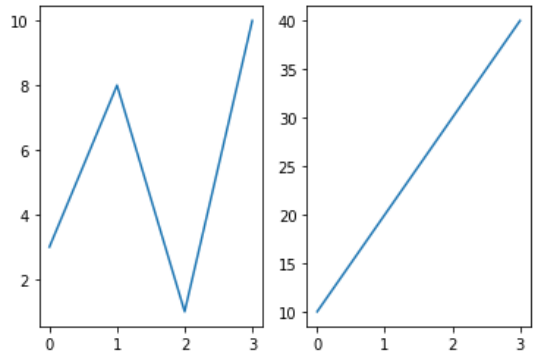

```
191) #2sd193c009
cars = ['AUDI', 'BMW', 'FORD',
    'TESLA', 'JAGUAR', 'MERCEDES']
data = [23, 17, 35, 29, 12, 41]
explode = (0.1, 0.0, 0.2, 0.3, 0.0, 0.0)
colors = ( "orange", "cyan", "brown",
    "grey", "indigo", "beige")
wp = { 'linewidth' : 1, 'edgecolor' : "green" }
def func(pct, allvalues):
  absolute = int(pct / 100.*np.sum(allvalues))
  return "{:.1f}%\n({:d} g)".format(pct, absolute)
fig, ax = plt.subplots(figsize =(10, 7))
wedges, texts, autotexts = ax.pie(data,
                 autopct = lambda pct: func(pct, data),
                 explode = explode,
                 labels = cars,
                 shadow = True,
                 colors = colors,
                 startangle = 90,
                 wedgeprops = wp,
                 textprops = dict(color ="magenta"))
ax.legend(wedges, cars,
    title ="Cars",
    loc ="center left",
    bbox_to_anchor =(1, 0, 0.5, 1))
plt.setp(autotexts, size = 8, weight ="bold")
ax.set_title("Customizing pie chart")
plt.show()
```

```
#2sd193c009
cars = ['AUDI', 'BMW', 'FORD',
    'TESLA', 'JAGUAR', 'MERCEDES']
data = [23, 17, 35, 29, 12, 41]
explode = (0.1, 0.0, 0.2, 0.3, 0.0, 0.0)
colors = ( "orange", "cyan", "brown",
    "grey", "indigo", "beige")
wp = { 'linewidth' : 1, 'edgecolor' : "green" }
def func(pct, allvalues):
  absolute = int(pct / 100.*np.sum(allvalues))
  return "{:.1f}%\n({:d} g)".format(pct, absolute)
fig, ax = plt.subplots(figsize =(10, 7))
wedges, texts, autotexts = ax.pie(data,
                autopct = lambda pct: func(pct, data),
                explode = explode,
                labels = cars,
                shadow = True,
                colors = colors,
                startangle = 90,
                wedgeprops = wp,
                textprops = dict(color ="magenta"))
ax.legend(wedges, cars,
    title ="Cars",
    loc ="center left",
    bbox_to_anchor =(1, 0, 0.5, 1))
plt.setp(autotexts, size = 8, weight ="bold")
ax.set_title("Customizing pie chart")
plt.show()
```



```
191)
x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])
plt.subplot(1, 2, 1)
plt.plot(x,y)
x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])
plt.subplot(1, 2, 2)
plt.plot(x,y)
plt.show()
```

```
#2sd19ec009
x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])
plt.subplot(1, 2, 1)
plt.plot(x,y)
x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])
plt.subplot(1, 2, 2)
plt.plot(x,y)
plt.show()
```



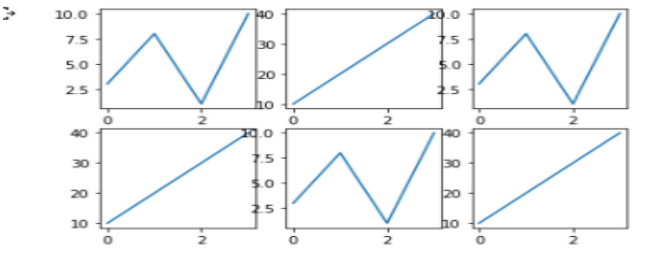

```
192)
a9=np.array([[0,0.5],[-1,2]])
vals,vecs=np.linalg.eig(a9)
print('vals= ',vals)
print('vecs= ',vecs)
print(np.isclose(np.sum(vals),a9.trace()))
```

```
sd19ec009
=np.array([[0,0.5],[-1,2]])
ls,vecs=np.linalg.eig(a9)
int('vals= ',vals)
int('vecs= ',vecs)
int(np.isclose(np.sum(vals),a9.trace()))
```

```
ls=  [0.29289322 1.70710678]
cs=  [[-0.86285621 -0.28108464]
-0.50544947 -0.95968298]]
ue
```

```
193)
x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])
plt.subplot(2, 3, 1)
plt.plot(x,y)
x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])
plt.subplot(2, 3, 2)
plt.plot(x,y)
x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])
plt.subplot(2, 3, 3)
plt.plot(x,y)
x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])
plt.subplot(2, 3, 4)
plt.plot(x,y)
x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])
plt.subplot(2, 3, 5)
plt.plot(x,y)
x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])
plt.subplot(2, 3, 6)
plt.plot(x,y)
plt.show()
```

```
#2sd19ec009
x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])
plt.subplot(2, 3, 1)
plt.plot(x,y)
x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])
plt.subplot(2, 3, 2)
plt.plot(x,y)
x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])
plt.subplot(2, 3, 3)
plt.plot(x,y)
x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])
plt.subplot(2, 3, 4)
plt.plot(x,y)
x = np.array([0, 1, 2, 3])
y = np.array([3, 8, 1, 10])
plt.subplot(2, 3, 5)
plt.plot(x,y)
x = np.array([0, 1, 2, 3])
y = np.array([10, 20, 30, 40])
plt.subplot(2, 3, 6)
plt.plot(x,y)
plt.show()
```



194)

```
from scipy import constants
print(constants.kibi)
print(constants.mebi)
print(constants.gibi)
print(constants.gram)
print(constants.metric_ton)
print(constants.degree)
print(constants.minute)
print(constants.hour)
print(constants.day)
```

115]

```
#2sd19ec009
from scipy import constants
print(constants.kibi)
print(constants.mebi)
print(constants.gibi)
print(constants.gram)
print(constants.metric_ton)
print(constants.degree)
print(constants.minute)
print(constants.hour)
print(constants.day)
```

```
1024
1048576
1073741824
0.001
1000.0
0.017453292519943295
60.0
3600.0
86400.0
```

195)

```
from scipy.integrate import quad
f9=lambda x:1/x**2
print(quad(f9,1,4))
```

```
#2sd19ec009
from scipy.integrate import quad
f9=lambda x:1/x**2
print(quad(f9,1,4))
```

```
(0.7500000000000002, 1.913234548258995e-09
```

196)

```
from scipy.integrate import quad
def integrand(x):
  return x**2
ans,err=quad(integrand,0,1)
print(ans)
```

```
18] #2sd19ec009
    from scipy.integrate import quad
    def integrand(x):
      return x**2
    ans,err=quad(integrand,0,1)
    print(ans)
```

```
0.33333333333333337
```

197)

```
def f(x,n,m):
  return np.sin(x)**n*np.cos(x)**m
n,m=2,1
print(quad(f, -np.pi/2,np.pi/2,args=(n,m)))
```

```
#2sd19ec009
def f(x,n,m):
  return np.sin(x)**n*np.cos(x)**m
n,m=2,1
print(quad(f, -np.pi/2,np.pi/2,args=(n,m)))
```

```
(0.6666666666666666, 1.6257269518146785e-13)
```

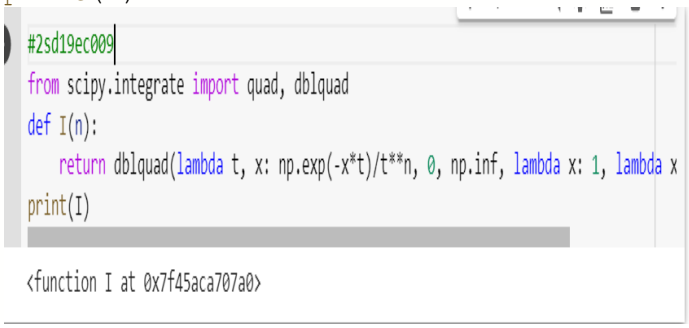
198)

```
from scipy import integrate
f=lambda y,x:x**2*y
a,b=1,4
gfun=lambda x:0
hfun=lambda x:2
print(integrate.dblquad(f,a,b,gfun,hfun))
```

```
[123] #2sd19ec009
      from scipy import integrate
      f=lambda y,x:x**2*y
      a,b=1,4
      gfun=lambda x:0
      hfun=lambda x:2
      print(integrate.dblquad(f,a,b,gfun,hfun))

      (42.00000000000001, 4.662936703425658e-13)
```

199)

```
from scipy.integrate import quad, dblquad
def I(n):
    return dblquad(lambda t, x: np.exp(-
x*t)/t**n, 0, np.inf, lambda x: 1, lambda x: np.inf)
print(I)
```



200)

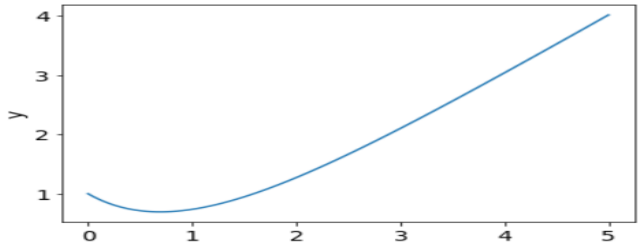
```
from scipy.integrate import tplquad
import numpy as np
def integrand(z,y,x):
  return y*np.sin(x)+z*np.cos(x)
ans,err=tplquad(integrand,0,np.pi,
                lambda x:0,
                lambda x:1,
                lambda x,y:-1,
                lambda x,y:1)
print(ans)
```

```
[132] #2sd19ec009
      from scipy.integrate import tplquad
      import numpy as np
      def integrand(z,y,x):
        return y*np.sin(x)+z*np.cos(x)
      ans,err=tplquad(integrand,0,np.pi,
                      lambda x:0,
                      lambda x:1,
                      lambda x,y:-1,
                      lambda x,y:1)
      print(ans)

      1.9999999999999998
```

201)

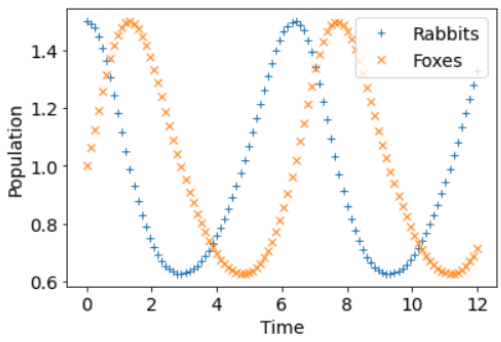
```
from scipy.integrate import odeint
def dy_dx(y, x):
    return x - y
xs = np.linspace(0,5,100)
y0 = 1.0
ys = odeint(dy_dx, y0, xs)
ys = np.array(ys).flatten()
plt.rcParams.update({'font.size': 14})
plt.xlabel("x")
plt.ylabel("y")
plt.plot(xs, ys);
```

```
#2sd19ec009
from scipy.integrate import odeint
def dy_dx(y, x):
    return x - y
xs = np.linspace(0,5,100)
y0 = 1.0
ys = odeint(dy_dx, y0, xs)
ys = np.array(ys).flatten()
plt.rcParams.update({'font.size': 14})
plt.xlabel("x")
plt.ylabel("y")
plt.plot(xs, ys);
```



```
202)
a,b,c,d = 1,1,1,1
def dP_dt(P, t):
    return [P[0]*(a - b*P[1]), -P[1]*(c - d*P[0])]

ts = np.linspace(0, 12, 100)
P0 = [1.5, 1.0]
Ps = odeint(dP_dt, P0, ts)
prey = Ps[:,0]
predators = Ps[:,1]
plt.plot(ts, prey, "+", label="Rabbits")
plt.plot(ts, predators, "x", label="Foxes")
plt.xlabel("Time")
plt.ylabel("Population")
plt.legend();
```

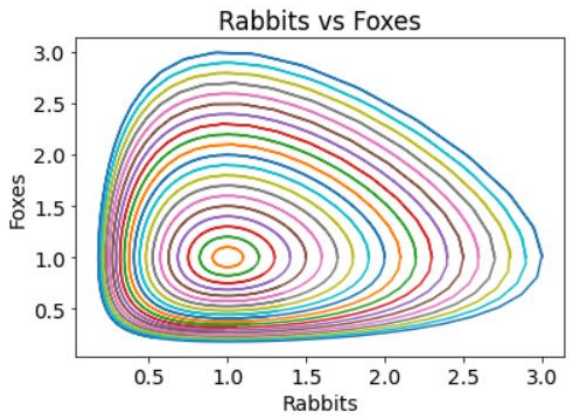
```
#2sd19ec009
a,b,c,d = 1,1,1,1
def dP_dt(P, t):
    return [P[0]*(a - b*P[1]), -P[1]*(c - d*P[0])]

ts = np.linspace(0, 12, 100)
P0 = [1.5, 1.0]
Ps = odeint(dP_dt, P0, ts)
prey = Ps[:,0]
predators = Ps[:,1]
plt.plot(ts, prey, "+", label="Rabbits")
plt.plot(ts, predators, "x", label="Foxes")
plt.xlabel("Time")
plt.ylabel("Population")
plt.legend();
```



203)

```
ic = np.linspace(1.0, 3.0, 21)
for r in ic:
    P0 = [r, 1.0]
    Ps = odeint(dP_dt, P0, ts)
    plt.plot(Ps[:,0], Ps[:,1], "-")
plt.xlabel("Rabbits")
plt.ylabel("Foxes")
plt.title("Rabbits vs Foxes");
```

```
#2sd19ec009
ic = np.linspace(1.0, 3.0, 21)
for r in ic:
    P0 = [r, 1.0]
    Ps = odeint(dP_dt, P0, ts)
    plt.plot(Ps[:,0], Ps[:,1], "-")
plt.xlabel("Rabbits")
plt.ylabel("Foxes")
plt.title("Rabbits vs Foxes");
```



204)

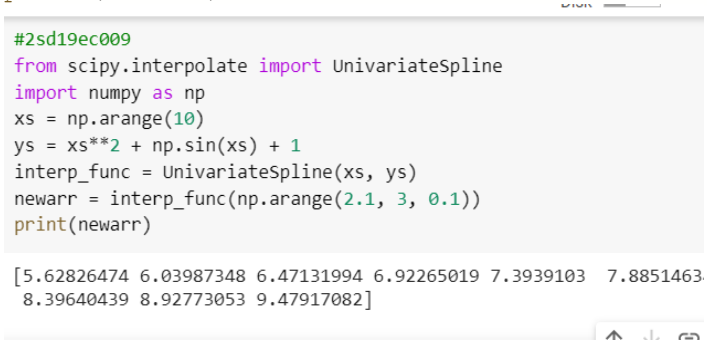
```
#2s19ec009
from scipy.interpolate import interp1d
import numpy as np
xs = np.arange(10)
ys = 2*xs + 1
interp_func = interp1d(xs, ys)
newarr = interp_func(np.arange(2.1, 3, 0.1))
print(newarr)
```

```
#2s19ec009
from scipy.interpolate import interp1d
import numpy as np
xs = np.arange(10)
ys = 2*xs + 1
interp_func = interp1d(xs, ys)
newarr = interp_func(np.arange(2.1, 3, 0.1))
print(newarr)
```

```
[5.2 5.4 5.6 5.8 6.  6.2 6.4 6.6 6.8]
```

205)

```
from scipy.interpolate import UnivariateSpline
import numpy as np
xs = np.arange(10)
ys = xs**2 + np.sin(xs) + 1
interp_func = UnivariateSpline(xs, ys)
newarr = interp_func(np.arange(2.1, 3, 0.1))
print(newarr)
```



```
#2sd19ec009
from scipy.interpolate import UnivariateSpline
import numpy as np
xs = np.arange(10)
ys = xs**2 + np.sin(xs) + 1
interp_func = UnivariateSpline(xs, ys)
newarr = interp_func(np.arange(2.1, 3, 0.1))
print(newarr)
```

```
[5.62826474 6.03987348 6.47131994 6.92265019 7.3939103  7.8851463
 8.39640439 8.92773053 9.47917082]
```

206)

```
from scipy.interpolate import Rbf
import numpy as np
xs = np.arange(10)
ys = xs**2 + np.sin(xs) + 1
interp_func = Rbf(xs, ys)
newarr = interp_func(np.arange(2.1, 3, 0.1))
print(newarr)
```

```
#2sd19ec009
from scipy.interpolate import Rbf
import numpy as np
xs = np.arange(10)
ys = xs**2 + np.sin(xs) + 1
interp_func = Rbf(xs, ys)
newarr = interp_func(np.arange(2.1, 3, 0.1))
print(newarr)
```

```
[6.25748981 6.62190817 7.00310702 7.40121814 7.8161443  8.247734
 8.69590519 9.16070828 9.64233874]
```