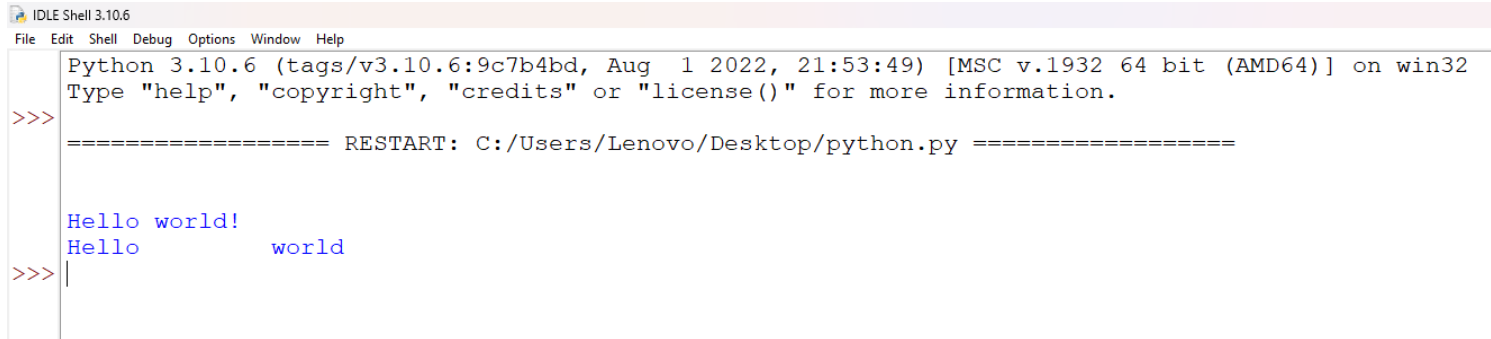


```
1. print(' ')
print(" ")
print("Hello world!")
print("Hello    world")
```

Op:



```
IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Lenovo/Desktop/python.py =====
Hello world!
Hello    world
>>> |
```

```
2
# it will print new line after the messages
print("Hello")
print("World")
```

```
# it will print new line
print()
```

```
# it will print new line after printing "Hello"
print("Hello",end="\n")
# it will print new line after printing "World"
print("World")
```

```
# it will print new line
print()
```

```
# it will not print new line after printing "Hello"
# it will print space " "
print("Hello",end=" ")
# it will print new line after printing "World"
print("World")
```

```
IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/py2.py =====
Hello
World

Hello
World

Hello World
>>> |
```

3.# Python code to print text/string using  
# user-defined method

```
# function definition of "putMe()"
# it will accepts a string and print it on the screen
def putMe(text):
    # printing text on the screen
    print(text)
```

```
# main code
if __name__ == '__main__':
    putMe('Hello world!')
    putMe('Welcome @')
    putMe('IncludeHelp')
    putMe('The world of programming')
    putMe('A programming community for developer and students')
```

```
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/py3.py =====
Hello world!
Welcome @
IncludeHelp
The world of programming
A programming community for developer and students
>>> |
```

4.  
#Printing different values  
# printing integer value  
print(15)  
# printing float value  
print(121.599)  
# printing string value  
print("RIT")  
# printing boolean value

```
print(True)
```

```
#Arithmetic operations inside the print() on values
```

```
# adding and printing integer value
```

```
print(12+36)
```

```
# adding and printing float value
```

```
print(12.56+12.45)
```

```
# adding and printing string value
```

```
print("RIT"+"CSE")
```

```
# adding and printing boolean value
```

```
print(True+False)
```

```
#Printing different types of variables
```

```
# variable with integer value
```

```
a=122
```

```
# variable with float value
```

```
b=121.563
```

```
# variable with string value
```

```
c="CSE"
```

```
# variable with Boolean value
```

```
d=True
```

```
# printing all variables
```

```
print(a)
```

```
print(b)
```

```
print(c)
```

```
print(d)
```

```
#Printing different types of variables along with the messages
```

```
# variable with integer value
```

```
a=1211
```

```
# variable with float value
```

```
b=1222.56
```

```
# variable with string value
```

```
c="RIT"
```

```
# variable with Boolean value
```

```
d=True
```

```
# printing values with messages
```

```
print("Integer\t:",a)
```

```
print("Float\t:",b)
```

```
print("String\t:",c)
```

```
print("Boolean\t:",d)
```

#Printing different types of variables with messages and converting them to string using "str()" function

# variable with integer value

a=12

# variable with float value

b=12.56

# variable with string value

c="Hello"

# variable with Boolean value

d=True

# printing values with messages

print("Integer\t:"+str(a))

print("Float\t:"+str(b))

print("String\t:"+str(c))

print("Boolean\t:"+str(d))

```
IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/py4.py =====
15
121.599
RIT
True
48
25.009999999999998
RITCSE
1
122
121.563
CSE
True
Integer : 1211
Float : 1222.56
String : RIT
Boolean : True
Integer :12
Float :12.56
String :Hello
Boolean :True
>>>
```

5.

#Declare different types of variables, print their values, types and Ids

print("Numbers")

print("-----")

a=19

print(type(a),id(a),a)

a=25.9

print(type(a),id(a),a)

a=7+7j

print(type(a),id(a),a)

```
print("\nText")
print("-----")
a='n' #single quotes
print(type(a),id(a),a)
a="n" #double quotes
print(type(a),id(a),a)
a='hi' #single"
print(type(a),id(a),a)
a="hi" #double""
print(type(a),id(a),a)
```

```
print("\nBoolean")
print("-----")
a=True
print(type(a),id(a),a)
```

```
print("\nFunction")
print("-----")
def fun1():
    return "I am from RIT"
a=fun1
print(type(a),id(a),a())
```

```
print("\nObjects")
print("-----")
class Demo:
    def hello(self):
        return "Hi"
a=Demo()
print(type(a),id(a),a.hello())
```

```
print("\nCollections")
print("-----")
a=[5,6,7]
print(type(a),id(a),a)
a=[]
print(type(a),id(a),a)
a=(5,6,7)
print(type(a),id(a),a)
a=()
print(type(a),id(a),a)
a=5,6,7
print(type(a),id(a),a)
a={5,6,7}
```

```
print(type(a),id(a),a)
a={}
print(type(a),id(a),a)
a={"id":5,"name":"computer"}
print(type(a),id(a),a)
```

```
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/103_5.py =====
Numbers
-----
<class 'int'> 2385424155440 19
<class 'float'> 2385463781232 25.9
<class 'complex'> 2385463782544 (7+7j)

Text
-----
<class 'str'> 2385429314992 n
<class 'str'> 2385429314992 n
<class 'str'> 2385432689968 hi
<class 'str'> 2385432689968 hi

Boolean
-----
<class 'bool'> 140711482641256 True

Function
-----
<class 'function'> 2385465229664 I am from RIT

Objects
-----
<class '__main__.Demo'> 2385465092624 Hi

Collections
-----
<class 'list'> 2385465169088 [5, 6, 7]
<class 'list'> 2385465171200 []
<class 'tuple'> 2385432745920 (5, 6, 7)
<class 'tuple'> 2385424171120 ()
<class 'tuple'> 2385432745920 (5, 6, 7)
<class 'set'> 2385465066880 {5, 6, 7}
<class 'dict'> 2385424975360 {}
<class 'dict'> 2385464813824 {'id': 5, 'name': 'computer'}
>>>
```

6.

#Python program to demonstrate variables scope

# Python code to demonstrate example

# of variable scopes

# global variable

a = 200

# defining a function to test scopes

def func():

# local variable

b = 100

# printing the value of global variable (a)

# and, local variable (b)

print("a: ", a, "b: ", b)

# main code

if \_\_name\_\_ == '\_\_main\_\_':

# local variable of main

c = 200

# printing values of a, b and c

print("a: ", a) #global

# print("a: ", b) #local of text \*\*\* will give an error

print("c: ", c) # local to main

# calling the function

func()

# updating the value of global variable 'a'

a = a+10

# printing 'a' again

print("a: ", a) #global

```
IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/103_6.py =====
a: 200
c: 200
a: 200 b: 100
a: 210
>>> |
```

7.

#Determine the type of an object in Python

# Python code to determine the type of objects

# declaring objects and assigning values

a = 19

b = 19.23

c = "Hello"

d = (100, 200, 300, 400)

e = [100, 200, 300, 400]

# printing types of the objects

# using type() function

print("type(a): ", type(a))

print("type(b): ", type(b))

print("type(c): ", type(c))

print("type(d): ", type(d))

print("type(e): ", type(e))

# printing the type of the value

# using type() function

print("type(19): ", type(19))

print("type(19.23): ", type(19.23))

print("type('Nikitha'): ", type("Nikitha"))

print("type((100, 200, 300, 400)): ", type((100, 200, 300, 400)))

print("type([100, 200, 300, 400]): ", type([100, 200, 300, 400]))

```
IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/103_7.py =====
type(a): <class 'int'>
type(b): <class 'float'>
type(c): <class 'str'>
type(d): <class 'tuple'>
type(e): <class 'list'>
type(19): <class 'int'>
type(19.23): <class 'float'>
type("Nikitha"): <class 'str'>
type((100, 200, 300, 400)): <class 'tuple'>
type([100, 200, 300, 400]): <class 'list'>
>>> |
```



8.

#Create number variables (int, float and complex) and print their types and values in Python

#Python code to create number variables, print types and values

# creating number variables and assigning values

a = 15    # integer

b = 105.23    # float

c = 15+5j    # complex

# printing types

print("type(a): ", type(a))

print("type(b): ", type(b))

print("type(c): ", type(c))

# printing values

print("value of a: ", a)

print("value of b: ", b)

print("value of c: ", c)

#Assigning integer number in binary, decimal, octal, and hexadecimal format

# creating integer variables and assigning values

# in different format

a = 567    # integer (decimal format)

b = 0o567    # integer (octal format)

c = 0x567AF    # integer (hexadecimal format)

d = 0b10101    # integer binary format

# printing types

print("type(a): ", type(a))

print("type(b): ", type(b))

print("type(c): ", type(c))

print("type(d): ", type(d))

# printing values

print("value of a: ", a)

print("value of b: ", b)

print("value of c: ", c)

print("value of d: ", d)

```
IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help

Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.

>>>
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/103_8.py =====
type(a): <class 'int'>
type(b): <class 'float'>
type(c): <class 'complex'>
value of a: 15
value of b: 105.23
value of c: (15+5j)
type(a): <class 'int'>
type(b): <class 'int'>
type(c): <class 'int'>
type(d): <class 'int'>
value of a: 567
value of b: 375
value of c: 354223
value of d: 21

>>>
```

9.

#Create integer variable by assigning binary value in Python

# Python code to create variable

# by assigning binary value

# creating number variable

# and, assigning binary value

a = 0b1110

b = 0b00000000

c = 0b11111111

d = 0b10101010

e = 0b11110000

# printing types

print("type of the variables...")

print("type of a: ", type(a))

print("type of b: ", type(b))

print("type of c: ", type(c))

print("type of d: ", type(d))

print("type of e: ", type(e))

# printing values in decimal format

print("value of the variables in decimal format...")

```

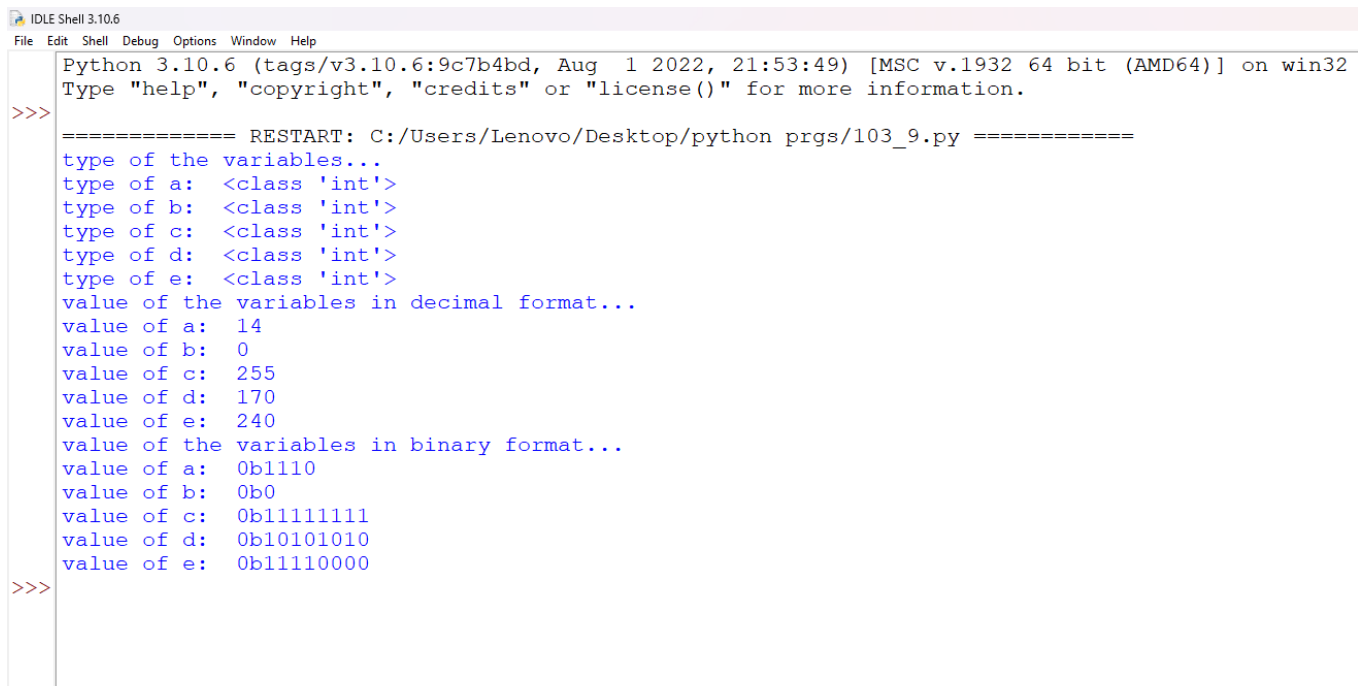
print("value of a: ", a)
print("value of b: ", b)
print("value of c: ", c)
print("value of d: ", d)
print("value of e: ", e)

```

```

# printing values in binary format
print("value of the variables in binary format...")
print("value of a: ", bin(a))
print("value of b: ", bin(b))
print("value of c: ", bin(c))
print("value of d: ", bin(d))
print("value of e: ", bin(e))

```



```

IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/103_9.py =====
type of the variables...
type of a: <class 'int'>
type of b: <class 'int'>
type of c: <class 'int'>
type of d: <class 'int'>
type of e: <class 'int'>
value of the variables in decimal format...
value of a: 14
value of b: 0
value of c: 255
value of d: 170
value of e: 240
value of the variables in binary format...
value of a: 0b1110
value of b: 0b0
value of c: 0b11111111
value of d: 0b10101010
value of e: 0b11110000
>>>

```

10.

#Create integer variable by assigning octal value in Python

# Python code to create variable

# by assigning octal value

# creating number variable

# and, assigning octal value

a = 0o1234567

b = 0o7654321

c = 0o1234

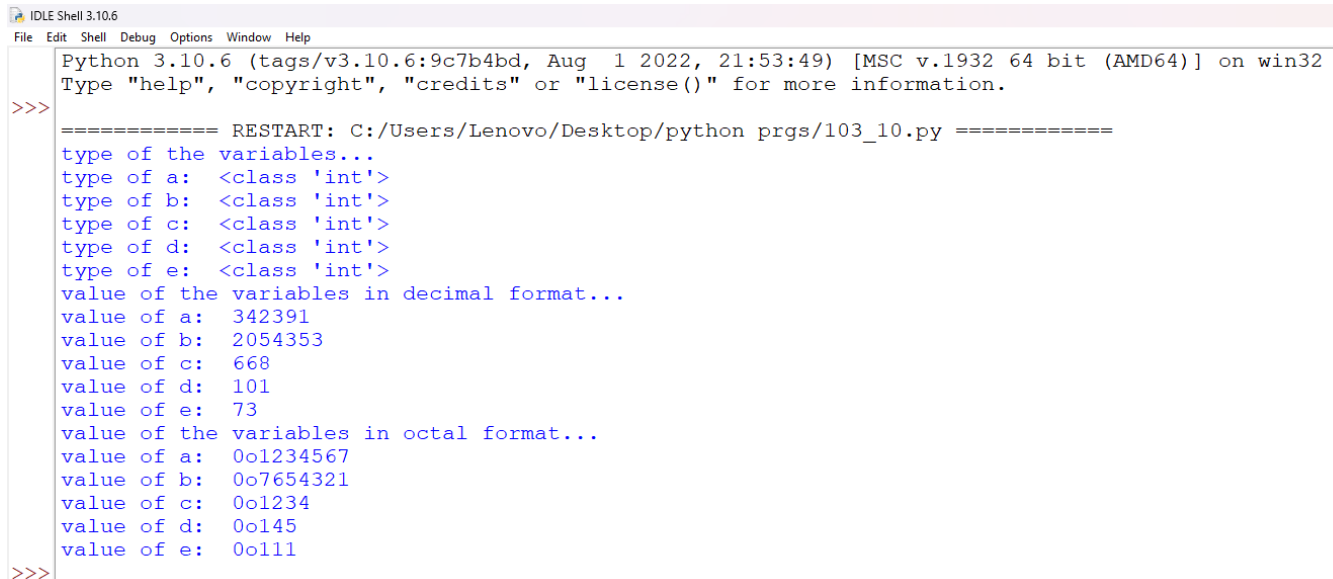
d = 0o145

e = 0o111

```
# printing types
print("type of the variables...")
print("type of a: ", type(a))
print("type of b: ", type(b))
print("type of c: ", type(c))
print("type of d: ", type(d))
print("type of e: ", type(e))
```

```
# printing values in decimal format
print("value of the variables in decimal format...")
print("value of a: ", a)
print("value of b: ", b)
print("value of c: ", c)
print("value of d: ", d)
print("value of e: ", e)
```

```
# printing values in octal format
print("value of the variables in octal format...")
print("value of a: ", oct(a))
print("value of b: ", oct(b))
print("value of c: ", oct(c))
print("value of d: ", oct(d))
print("value of e: ", oct(e))
```



```
IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/103_10.py =====
type of the variables...
type of a: <class 'int'>
type of b: <class 'int'>
type of c: <class 'int'>
type of d: <class 'int'>
type of e: <class 'int'>
value of the variables in decimal format...
value of a: 342391
value of b: 2054353
value of c: 668
value of d: 101
value of e: 73
value of the variables in octal format...
value of a: 0o1234567
value of b: 0o7654321
value of c: 0o1234
value of d: 0o145
value of e: 0o111
>>>
```

11.

#Create integer variable by assigning hexadecimal value in Python

# Python code to create variable

# by assigning hexadecimal value

# creating number variable

# and, assigning hexadecimal value

a = 0x111 #lowercase x

b = 0X111 #uppercase x

c = 0xCDCD

d = 0Xcdcd

e = 0x1234abcdef

# printing types

print("type of the variables...")

print("type of a: ", type(a))

print("type of b: ", type(b))

print("type of c: ", type(c))

print("type of d: ", type(d))

print("type of e: ", type(e))

# printing values in decimal format

print("value of the variables in decimal format...")

print("value of a: ", a)

print("value of b: ", b)

print("value of c: ", c)

print("value of d: ", d)

print("value of e: ", e)

# printing values in hexadecimal format

print("value of the variables in hexadecimal format...")

print("value of a: ", hex(a))

print("value of b: ", hex(b))

print("value of c: ", hex(c))

print("value of d: ", hex(d))

print("value of e: ", hex(e))

```
IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/103_11.py =====
type of the variables...
type of a: <class 'int'>
type of b: <class 'int'>
type of c: <class 'int'>
type of d: <class 'int'>
type of e: <class 'int'>
value of the variables in decimal format...
value of a: 273
value of b: 273
value of c: 52685
value of d: 52685
value of e: 78193085935
value of the variables in hexadecimal format...
value of a: 0x111
value of b: 0x111
value of c: 0xcdcd
value of d: 0xcdcd
value of e: 0x1234abcdef
>>>
```

12.

#Python | Typecasting Input to Integer, Float

# input to integer

```
num = int(input("Input a value: "))
```

# printing input value

```
print("num = ", num)
```

# input to float

```
num = float(input("Input a value: "))
```

# printing input value

```
print("num = ", num)
```

```
IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/103_12.py =====
>>> Input a value: 5
num = 5
Input a value: 5
num = 5.0
>>>
```

13.

#How to check multiple variables against a value in Python?

a = 14

b = 26

c = 33

# method 1

if a == 14 or b == 14 or c == 14:

print("True")

else:

print("False")

# method 2

if 14 in (a, b, c):

print("True")

else:

print("False")

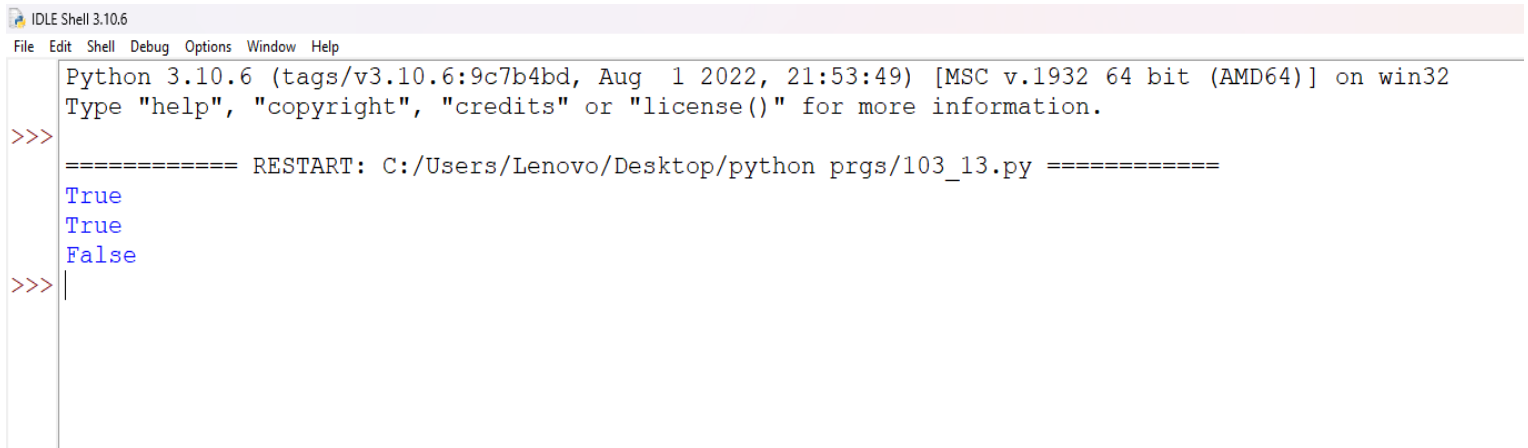
# method 3

if 10 in {a, b, c}:

print("True")

else:

print("False")



```
IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/103_13.py =====
True
True
False
>>> |
```

14.

#Program to define an integer value and print it

# Python program to define an

# integer value and print it

# declare and assign an integer value

num = 15

# print num

print("num =",num )

# print using format

print("num = {0}".format(num))

# assign another value

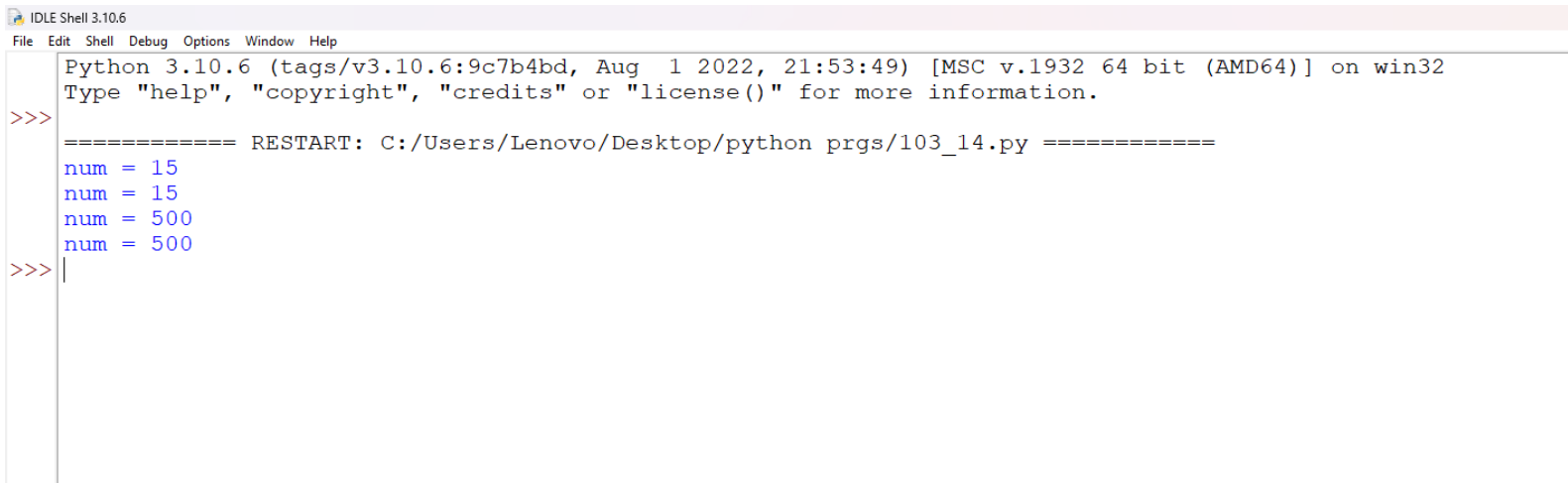
num = 500

# print num

print("num =",num)

# print using format

print("num = {0}".format(num))



```
IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/103_14.py =====
num = 15
num = 15
num = 500
num = 500
>>> |
```

15.

#Input two integers and find their addition

# input two numbers: value of a and b

a = int(input("Enter A: "))

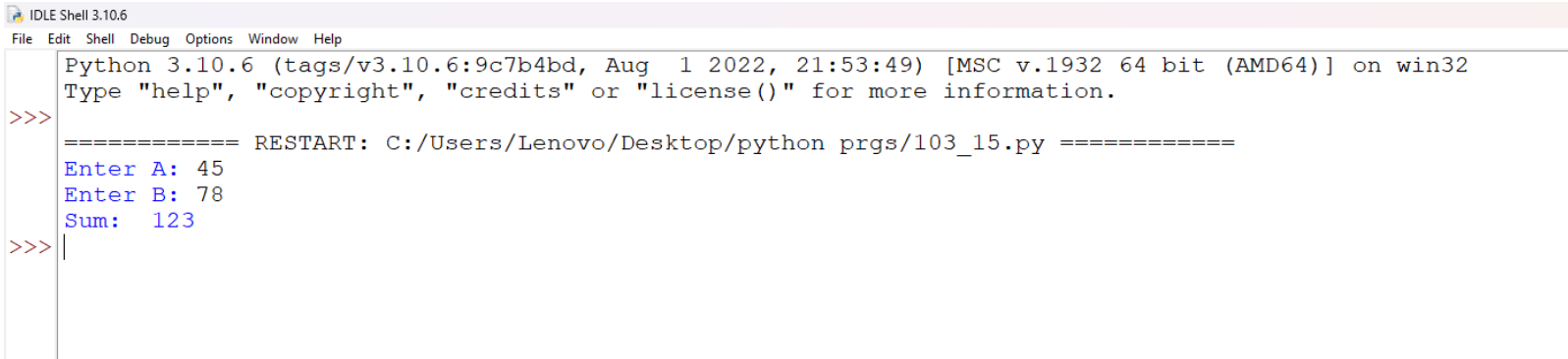
b = int(input("Enter B: "))

# find sum of a and b and assign to c

c = a+b



```
# print sum (c)
print("Sum: ",c)
```

A screenshot of the IDLE Shell 3.10.6 window. The title bar reads 'IDLE Shell 3.10.6'. The menu bar includes 'File', 'Edit', 'Shell', 'Debug', 'Options', 'Window', and 'Help'. The shell area shows the following text: 'Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32' followed by 'Type "help", "copyright", "credits" or "license()" for more information.' Below this, there are three lines of input: '>>>' followed by 'Enter A: 45', 'Enter B: 78', and 'Sum: 123'. The prompt '>>>' is shown again on the next line. Above the input, there is a line of text: '===== RESTART: C:/Users/Lenovo/Desktop/python prgs/103\_15.py ====='.

16.

```
# program to find sum of two numbers
# python program to find sum of
# two numbers
```

```
num1 = 45
num2 = 76
```

```
# finding sum
sum = num1 + num2
```

```
# printing sum
print("sum of ", num1, " and ", num2, " is = ", sum)
```

```
# taking input from user
num1 = input("Enter first number: ")
num2 = input("Enter second number: ")
```

```
# finding sum
sum = int(num1) + int(num2)
```

```
# printing sum
print("sum of ", num1, " and ", num2, " is = ", sum)
```

```
IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/103_16.py =====
sum of 45 and 76 is = 121
Enter first number: 45
Enter second number: 45
sum of 45 and 45 is = 90
>>>
```

17.

#program to find addition of two numbers (4 different ways)

""" 1) Simply take input from the user and typecast to an integer  
at the same time after that  
performing addition operation on the both number """

```
if __name__ == "__main__":
```

```
    # take input from user
```

```
    a = int(input("a:"))
```

```
    b = int(input("b:"))
```

```
    # addition operation perform
```

```
    sum_num = a + b
```

```
    print("sum of two number is: ",sum_num)
```

""" 2) Using a user-defined function for  
doing the sum of two numbers. """

```
# define a function for performing
```

```
# addition of number
```

```
def sum_num(a,b):
```

```
    return a + b
```

```
# Main code
```

```
if __name__ == "__main__":
```

```
    a = int(input("a:"))
```

```
    b = int(input("b:"))
```

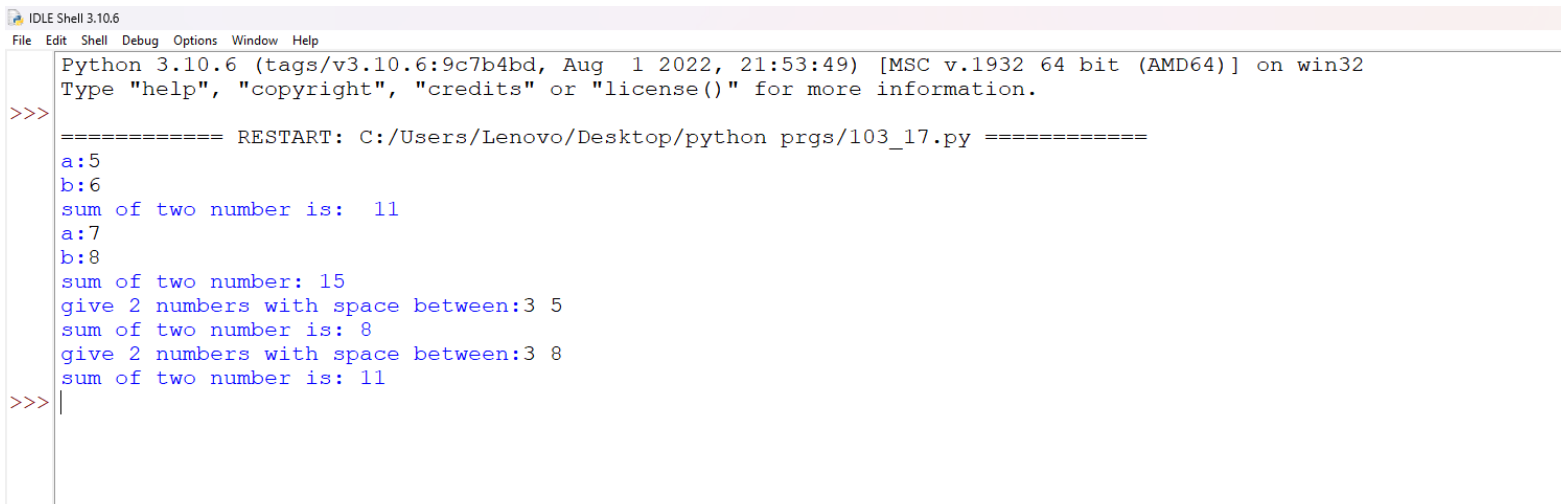
```
print("sum of two number:",sum_num(a,b))
```

''' 3) We are taking the input from user in one line after that typecast into an integer and stored them in the list then use sum() inbuilt function which returns the sum of elements of the list. '''

```
if __name__ == "__main__":  
    # take input from the user in list  
    a = list(map(int,input("give 2 numbers with space between:").split()))  
  
    # sum function return sum of elements  
    # present in the list  
    print("sum of two number is:",sum(a))
```

''' 4) We are taking the input from user in one line and store them in two different variables then typecast both into an integer at the time of addition operation.'''

```
if __name__ == "__main__":  
  
    # take input from the user in a and b variables  
    a,b = input("give 2 numbers with space between:").split()  
  
    # perform addition operation  
    r = int(a) + int(b)  
  
    print("sum of two number is:",r)
```



```
IDLE Shell 3.10.6  
File Edit Shell Debug Options Window Help  
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license()" for more information.  
>>>=====  
a:5  
b:6  
sum of two number is: 11  
a:7  
b:8  
sum of two number: 15  
give 2 numbers with space between:3 5  
sum of two number is: 8  
give 2 numbers with space between:3 8  
sum of two number is: 11  
>>>
```

#Python program to demonstrate the example for arithmetic operators

a = 18

b = 3

# addition

result = a+b

print("a+b :", result)

# subtraction

result = a-b

print("a-b :", result)

# division

result = a/b

print("a/b :", result)

# modulus

result = a%b

print("a%b :", result)

# exponent

result = a\*\*b

print("a\*\*b :", result)

# floor division

result = a//b

print("a//b :", result)

# updating the values of a & b

a = -19

b = 3

print("a:", a, "b:", b)

result = a//b

print("a//b :", result)

```
IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/103_18.py =====
a+b : 21
a-b : 15
a/b : 6.0
a%b : 0
a**b : 5832
a//b : 6
a: -19 b: 3
a//b : -7
>>> |
```

19.

# python program to print ASCII

# value of a given character

# Assigning character to a variable

char\_var = 'A'

# printing ASCII code

print("ASCII value of " + char\_var + " is = ", ord(char\_var))

char\_var = 'a'

# printing ASCII code

print("ASCII value of " + char\_var + " is = ", ord(char\_var))

char\_var = '8'

# printing ASCII code

print("ASCII value of " + char\_var + " is = ", ord(char\_var))

```
IDLE Shell 3.10.6
File Edit Shell Debug Options Window Help
Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/Lenovo/Desktop/python prgs/103_19.py =====
ASCII value of A is = 65
ASCII value of a is = 97
ASCII value of 8 is = 56
>>>
```

20.

# Python program to find simple interest

```
p = float(input("Enter the principle amount : "))
```

```
r = float(input("Enter the rate of interest : "))
```

```
t = float(input("Enter the time in the years: "))
```

# calculating simple interest

```
si = (p*r*t)/100
```

# printing the values

```
print("Principle amount: ", p)
```

```
print("Interest rate  : ", r)
```

```
print("Time in years  : ", t)
```

```
print("Simple Interest : ", si)
```

File Edit Shell Debug Options Window Help

Python 3.10.6 (tags/v3.10.6:9c7b4bd, Aug 1 2022, 21:53:49) [MSC v.1932 64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license()" for more information.

>>>

===== RESTART: C:/Users/Lenovo/Desktop/python prgs/103\_20.py =====

Enter the principle amount : 500

Enter the rate of interest : 4

Enter the time in the years: 2

Principle amount: 500.0

Interest rate : 4.0

Time in years : 2.0

Simple Interest : 40.0

>>>