 Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Write a program to demonstrate different number datatypes in python.	
Experiment No: 01	Date:07/07/2025	Enrollment No: 92400133037

Aim: Write a program to demonstrate different number datatypes in python.

IDE:

Data types in Python refer to classifying or categorizing data objects based on their characteristics and behavior. They determine the type of values variables can hold and specify the operations that can be performed on those values. For instance, Python has several built-in data types, including numeric types (int, float, complex), string (str), Boolean (bool), and collection types (list, tuple, dict, set). Moreover, each data type has its own set of properties, methods, and behaviors that allow programmers to manipulate and process data effectively in their programs.

Built-in Data Types in Python

Built-in data types in Python are fundamental data structures provided by the Python programming language. Pre-defined and available for use without requiring any additional libraries or modules. Python offers several built-in data types, including:

Numeric Data Types: Numeric data types in Python are used to represent numerical values. Python provides three primary numeric datatypes in python:



- **Integer (int):** Integers are whole numbers without any decimal points. They can be positive or negative.
- **Floating-Point (float):** Floating-point numbers represent decimal values. They can be positive or negative and may contain a decimal point.
- **Complex (complex):** People use complex numbers to represent numbers with a real and imaginary part. You write them in the form of $a + bj$, where a is the real part and b is the imaginary part.

String Data Type(str): Represents a sequence of characters enclosed in single quotes (‘ ’) or double quotes (“ ”), such as “Hello, World!”, ‘Python’.

Boolean Data Type(bool): Represents either True or False, used for logical operations and conditions.

Collection Data Types:

- **list:** Represents an ordered and mutable collection of items, enclosed in square brackets [].
- **tuple:** Represents an ordered and immutable collection of items, enclosed in parentheses ()

<div><div>Marwadi University <small>Marwadi Chandarana Group</small></div></div> <div></div>		Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)		Aim: Write a program to demonstrate different number datatypes in python.	
Experiment No: 01	Date:07/07/2025	Enrollment No: 92400133037	

- dict: Represents a collection of key-value pairs enclosed in curly braces {} with unique keys.
- set: Represents an unordered and mutable collection of unique elements, enclosed in curly braces {} or using the set() function.

Numeric Data Types

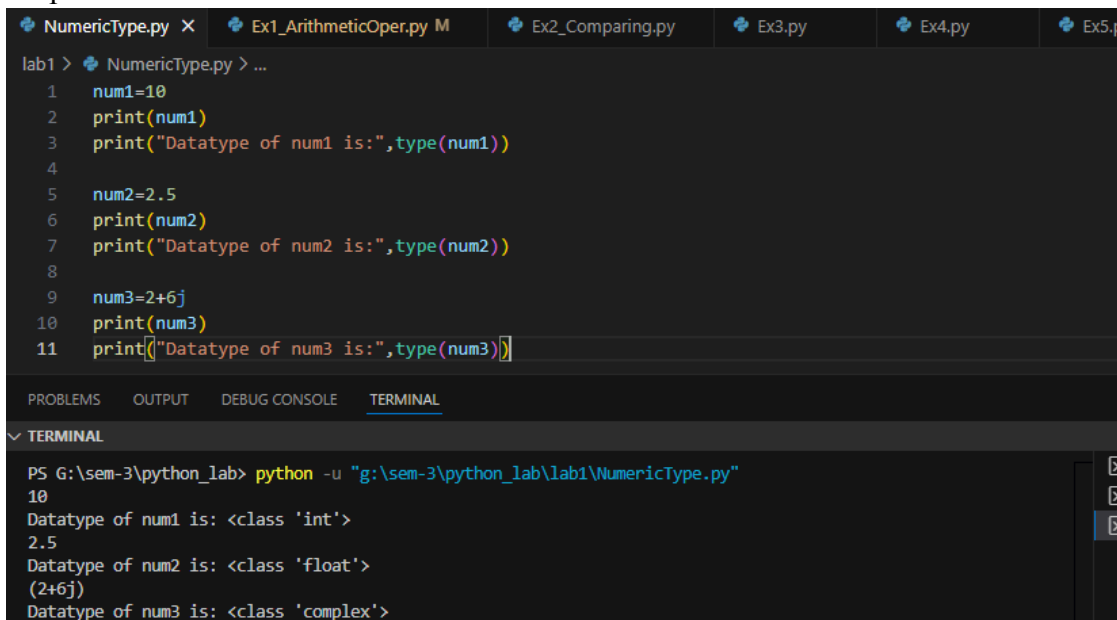
Python Code:

```
num1 = 10
print(num1)
print("Datatype of num1 is", type(num1))
```

```
num2 = 2.5
print(num2)
print("Datatype of num1 is", type(num2))
```

```
num3 = 2+6j
print(num3)
print("Datatype of num1 is", type(num3))
```

Output:




The screenshot shows a Python IDE with a file named 'NumericType.py' open. The code in the editor is as follows:

```
1 num1=10
2 print(num1)
3 print("Datatype of num1 is:",type(num1))
4
5 num2=2.5
6 print(num2)
7 print("Datatype of num2 is:",type(num2))
8
9 num3=2+6j
10 print(num3)
11 print("Datatype of num3 is:",type(num3))
```

The output window shows the following results:

```
PS G:\sem-3\python_lab> python -u "g:\sem-3\python_lab\lab1\NumericType.py"
10
Datatype of num1 is: <class 'int'>
2.5
Datatype of num2 is: <class 'float'>
(2+6j)
Datatype of num3 is: <class 'complex'>
```

 Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Write a program to demonstrate different number datatypes in python.	
Experiment No: 01	Date:07/07/2025	Enrollment No: 92400133037

Example 1

x = 5

y = -6

Performing arithmetic operations

sum_result = x + y

difference_result = x - y

multiplication_result = x * y

division_result = x / y

Printing the results

print("Sum:", sum_result)

print("Difference:", difference_result)

print("Multiplication:", multiplication_result)

print("Division:", division_result)


Output:

```
lab1 > Ex1_ArithmeticOper.py > ...
1  x,y=5,-6
2  print("Sum:",(x+y))
3  print("Difference:",(x-y))
4  print("Multiplication:",(x*y))
5  print("Division:",(x/y))
6  print("Division:",(x//y))#integer ans
7  print("Modulus:",(x%y))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

▼ **TERMINAL**

```
(2, 3, 4, 2, 3, 4, 5, 6, 7)
PS G:\sem-3\python_lab> python -u "g:\sem-3\python_lab\lab1\Ex1_ArithmeticOper.py"
Sum: -1
Difference: 11
Multiplication: -30
Division: -0.8333333333333334
Division: -1
Modulus: -1
PS G:\sem-3\python_lab>
```

<div><div><div>Marwadi University</div><div>Marwadi Chandarana Group</div></div><div><div>NAAC</div><div>A+</div></div></div>		<div>Marwadi University</div> <div>Faculty of Engineering & Technology</div> <div>Department of Information and Communication Technology</div>	
<div>Subject: Programming With Python (01CT1309)</div>		<div>Aim: Write a program to demonstrate different number datatypes in python.</div>	
<div>Experiment No: 01</div>	<div>Date:07/07/2025</div>	<div>Enrollment No: 92400133037</div>	

Example 2:

```

a = 10
b = 20
# Comparing the values
greater_than = a > b
less_than_or_equal = a <= b
equal_to = a == b
not_equal_to = a != b
# Printing the results
print("Greater than:", greater_than)
print("Less than or equal to:", less_than_or_equal)
print("Equal to:", equal_to)
print("Not equal to:", not_equal_to)

```

Output

```

lab1 > Ex2_Comparing.py > ...
1  a,b=10,20
2  print("Greater than:",(a>b))
3  print("less than or equal to:",(a<=b))
4  print("Equal:",(a==b))
5  print("Not equal to:",(a!=b))
6  |

```


PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

✓ **TERMINAL**

```

(2, 3, 4, 2, 3, 4, 5, 6, 7)
PS G:\sem-3\python_lab> python -u "g:\sem-3\python_lab\lab1\Ex1_ArithmeticOper.py"
Sum: -1
Difference: 11
Multiplication: -30
Division: -0.8333333333333334
Division: -1
Modulus: -1
PS G:\sem-3\python_lab> python -u "g:\sem-3\python_lab\lab1\Ex2_Comparing.py"
Greater than: False
less than or equal to: True
Equal: False
Not equal to: True
PS G:\sem-3\python_lab>

```

 Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Write a program to demonstrate different number datatypes in python.	
Experiment No: 01	Date:07/07/2025	Enrollment No: 92400133037

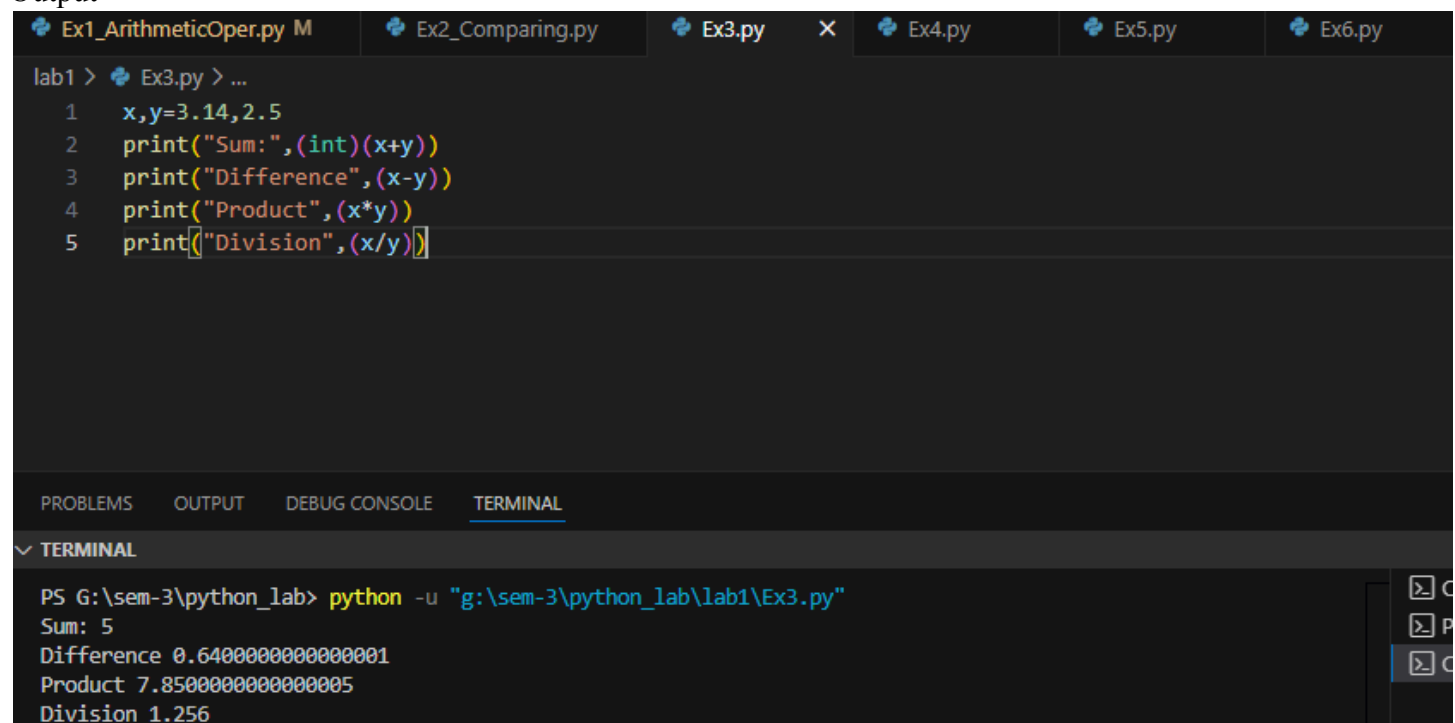
Example 3

```

x = 3.14
y = 2.5
# Performing arithmetic operations
sum_result = x + y
difference_result = x - y
multiplication_result = x * y
division_result = x / y
# Printing the results
print("Sum:", sum_result)
print("Difference:", difference_result)
print("Multiplication:", multiplication_result)
print("Division:", division_result)

```

Output



The screenshot shows a code editor with several tabs: Ex1_ArithmeticOper.py, Ex2_Comparing.py, Ex3.py, Ex4.py, Ex5.py, and Ex6.py. The active tab is Ex3.py, which contains the following Python code:

```

lab1 > Ex3.py > ...
1  x,y=3.14,2.5
2  print("Sum:",(int)(x+y))
3  print("Difference",(x-y))
4  print("Product",(x*y))
5  print("Division",(x/y))


```

Below the code editor, the terminal window is open, showing the command to run the script and its output:

```

PS G:\sem-3\python_lab> python -u "g:\sem-3\python_lab\lab1\Ex3.py"
Sum: 5
Difference 0.6400000000000001
Product 7.8500000000000005
Division 1.256

```

 Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Write a program to demonstrate different number datatypes in python.	
Experiment No: 01	Date:07/07/2025	Enrollment No: 92400133037

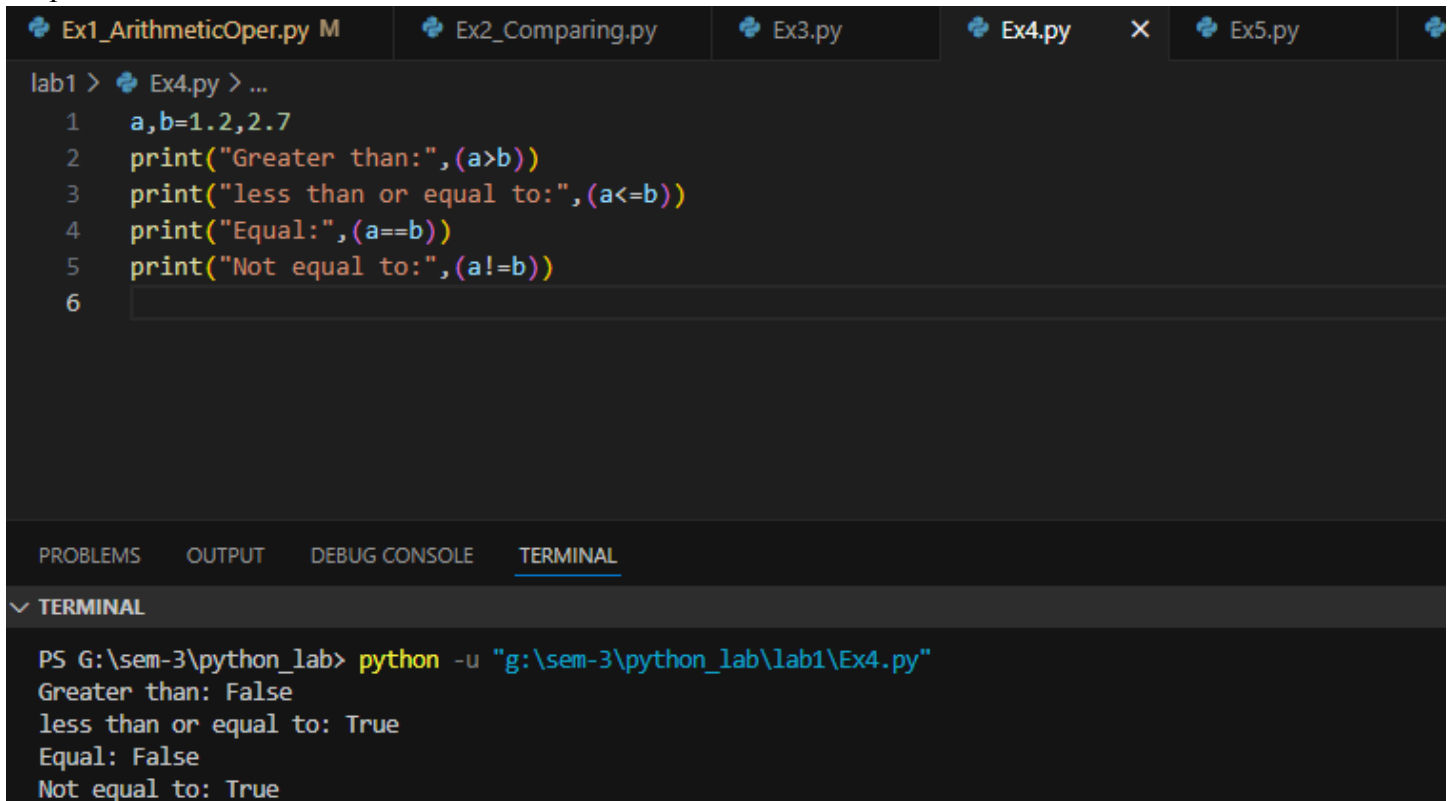
Example 4

```

a = 1.2
b = 2.7
# Comparing the values
greater_than = a > b
less_than_or_equal = a <= b
equal_to = a == b
not_equal_to = a != b
# Printing the results
print("Greater than:", greater_than)
print("Less than or equal to:", less_than_or_equal)
print("Equal to:", equal_to)
print("Not equal to:", not_equal_to)

```

Output



```

lab1 > Ex4.py > ...
1  a,b=1.2,2.7
2  print("Greater than:",(a>b))
3  print("less than or equal to:",(a<=b))
4  print("Equal:",(a==b))
5  print("Not equal to:",(a!=b))
6

```


PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

✓ **TERMINAL**

```

PS G:\sem-3\python_lab> python -u "g:\sem-3\python_lab\lab1\Ex4.py"
Greater than: False
less than or equal to: True
Equal: False
Not equal to: True

```

 Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Write a program to demonstrate different number datatypes in python.	
Experiment No: 01	Date:07/07/2025	Enrollment No: 92400133037

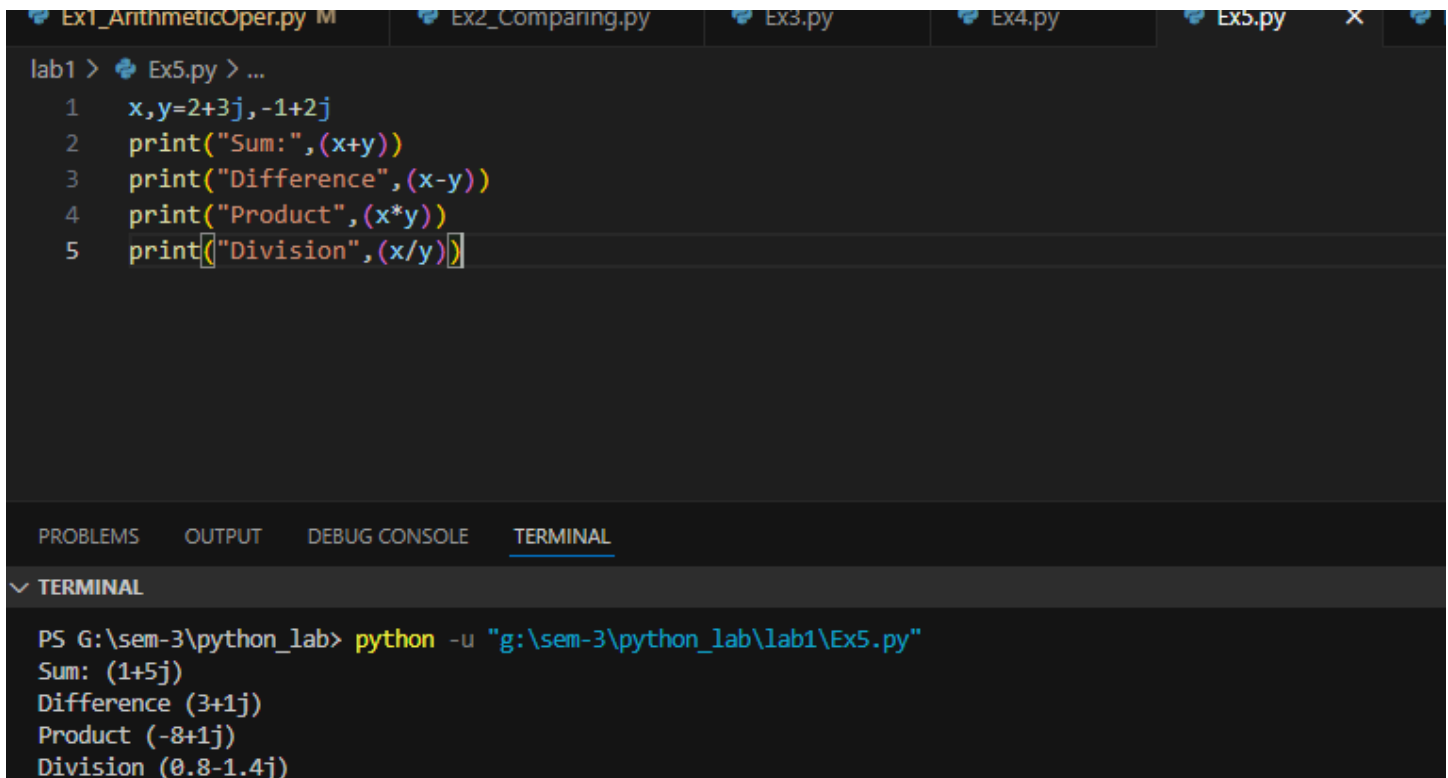
Example 5

```

x = 2 + 3j
y = -1 + 2j
# Performing arithmetic operations
sum_result = x + y
difference_result = x - y
multiplication_result = x * y
division_result = x / y
# Printing the results
print("Sum:", sum_result)
print("Difference:", difference_result)
print("Multiplication:", multiplication_result)
print("Division:", division_result)

```

Output



The screenshot shows a Python IDE with a dark theme. The top bar displays several open files: Ex1_ArithmeticOper.py, Ex2_Comparing.py, Ex3.py, Ex4.py, and Ex5.py. The main editor window shows the code from Example 5. Below the editor, there are tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL. The TERMINAL tab is active, showing the command prompt and the output of the program.

```

lab1 > Ex5.py > ...
1  x,y=2+3j,-1+2j
2  print("Sum:",(x+y))
3  print("Difference",(x-y))
4  print("Product",(x*y))
5  print(["Division",(x/y)])


```

TERMINAL

```

PS G:\sem-3\python_lab> python -u "g:\sem-3\python_lab\lab1\Ex5.py"
Sum: (1+5j)
Difference (3+1j)
Product (-8+1j)
Division (0.8-1.4j)

```

 Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Write a program to demonstrate different number datatypes in python.	
Experiment No: 01	Date:07/07/2025	Enrollment No: 92400133037

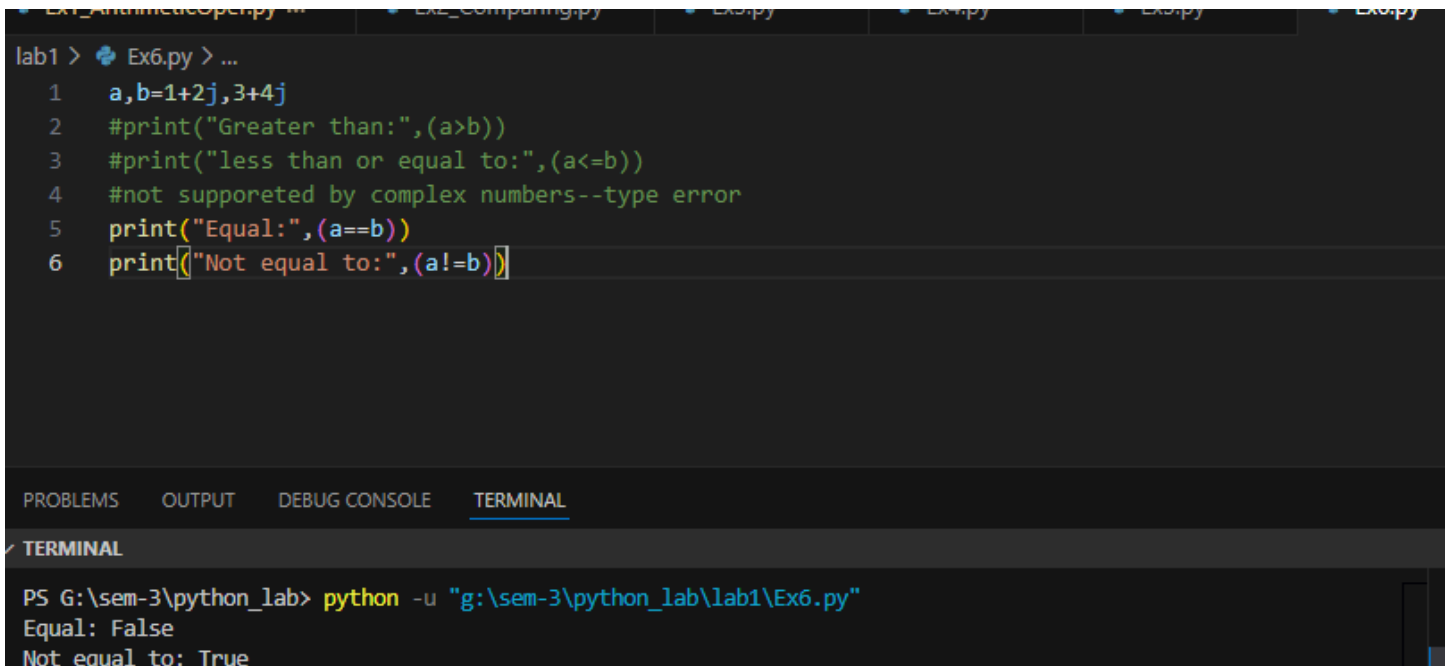
Example 6

```

a = 1 + 2j
b = 3 + 4j
# Comparing the values
equal_to = a == b
not_equal_to = a != b
# Printing the results
print("Equal to:", equal_to)
print("Not equal to:", not_equal_to)

```

Output



The screenshot shows a Python IDE with a file named 'Ex6.py' open. The code in the editor is as follows:

```

1  a,b=1+2j,3+4j
2  #print("Greater than:",(a>b))
3  #print("less than or equal to:",(a<=b))
4  #not supported by complex numbers--type error
5  print("Equal:",(a==b))
6  print("Not equal to:",(a!=b))

```

The terminal output shows the execution of the script:

```

PS G:\sem-3\python_lab> python -u "g:\sem-3\python_lab\lab1\Ex6.py"
Equal: False
Not equal to: True


```

Example 7

```

st1 = "ICT Department 3EK1"
print(st1)
print(st1[0])
print(st1[0:4])

```


 Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Write a program to demonstrate different number datatypes in python.	
Experiment No: 01	Date:07/07/2025	Enrollment No: 92400133037

Output

```

lab1 > Ex7_String.py > ...
1  st1="ICT Department 3ek1"
2  print(st1)
3  print(st1[0])
4  print(st1[0:5])
5

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

✓ **TERMINAL**

```

PS G:\sem-3\python_lab> python -u "g:\sem-3\python_lab\lab1\Ex7_String.py"
ICT Department 3ek1
I
ICT D

```

Example 8

```


st1 = "ICT"
st2 = "Department"
st3 = "3EK1"
print(st1+st2+st3)

```

Repetitions: Python allows us to repeat a given string with the help of ‘ * ‘ operator.

```
print(4*st1)
```

Output

 Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Write a program to demonstrate different number datatypes in python.	
Experiment No: 01	Date:07/07/2025	Enrollment No: 92400133037

```
lab1 > Ex8.py > ...
1  st1="ICT"
2  st2="Department"
3  st3="3EK1"
4  print(st1+" "+st2+" "+st3)
5
6  #repetitions
7  print(4*(st1+" "))
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

✓ TERMINAL

```
PS G:\sem-3\python_lab> python -u "g:\sem-3\python_lab\lab1\Ex8.py"
ICT Department 3EK1
ICT ICT ICT ICT
```

Membership: The Membership operator helps to check whether a given character is present in the string or not with the help of two operators in and not in. In and not in operator returns the Boolean value True or False.

```
st1 = "ICT Department 3EK1"
print("p" in st1)
```

Output

```
Print("f" not in st1)
```


Output

```
lab1 > MembershipOp.py > ...
1  st1="ICT Department 3EK1"
2  print("p" in st1)
3  print("f" not in st1)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

✓ TERMINAL

```
PS G:\sem-3\python_lab> python -u "g:\sem-3\python_lab\lab1\MembershipOp.py"
True
True
```

 Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Write a program to demonstrate different number datatypes in python.	
Experiment No: 01	Date:07/07/2025	Enrollment No: 92400133037

Collection Data Types

Collection data types in Python are used to store and organize multiple values into a single entity. Python provides several built-in collection data types, including lists, tuples, dictionaries, and sets.

```
list1=[123,567,89]
print(list1)
```

```
list2=["hello","how","are"]
print(list2)
```

```
list3= ["hey",1223,"hello"]
print(list3)
```

Output

```
lab1 > collection.py > ...
1 #lists,tuples,dictionaries,sets
2 list1=[123,567,89]
3 print(list1)
4
5 list2=["hello","how","are"]
6 print(list2)
7
8 list3=["hey","123","hello"]
9 print(list3)
10


▼ TERMINAL

PS G:\sem-3\python_lab> python -u "g:\sem-3\python_lab\lab1\collection.py"
[123, 567, 89]
['hello', 'how', 'are']
['hey', '123', 'hello']
```

```
list1=["apple","mango",123,345]
list2 = ["grapes"]
print(list1+ list2)
```

Output

```
11 list4=["apple","mango",123,345]
12 list5=["grapes"]
13 print(list4+list5)
14
['apple', 'mango', 123, 345, 'grapes']
```

 Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Write a program to demonstrate different number datatypes in python.	
Experiment No: 01	Date:07/07/2025	Enrollment No: 92400133037

```
dict1={"comp": "computer" , "sci" : "science"}
print(dict1["comp"])
dict2={"123": "computer", 456 : "maths"}
print(dict2["123"])
print(dict1["comp"]+ dict2["123"])
```

Check

```
print(dict1+ dict2)
print(dict1["computer"]+ dict2["computer"])
```

Output


```
15 print("\n\nDictionaries")
16 dict1={"comp": "computer", "sci": "science"}
17 print(dict1["comp"])
18 dict2={"123": "computer", "456": "maths"}
19 print(dict2["123"])
20 print(dict1["comp"]+dict2["123"])
21 #print(dict1+dict2)--type error
22 #print(dict1["computer"]+dict2["computer"])--key error
23
24 print("\n\nSets")
25 my_set={1,2,3,4,5}
26 print(my_set)
27
```

```
Dictionaries
computer
computer
computercomputer
```

```
my_set = {1, 2, 3, 4, 5}
print(my_set)
```

```
set1 = {1, 2, 3, 4, 5}
set2 = {4, 5, 6, 7, 8}
check
print(set1 + set2)
```

Output

 Marwadi University Marwadi Chandarana Group	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Write a program to demonstrate different number datatypes in python.	
Experiment No: 01	Date:07/07/2025	Enrollment No: 92400133037

```

24  print("\n\nSets")
25  my_set={1,2,3,4,5}
26  print(my_set)
27

```

```

28  set1={1,2,3,4,5}
29  set2={4,5,6,7,8}
30  #print(set1+set2)--type error

```

```

Sets
{1, 2, 3, 4, 5}

```

```

my_tuple = (1, 2, 3, 4, 5)
t1 = (2,3,4)
t2 = (5,6,7)
print(t1+t2)

```

Output

```

32  print("\n\nTuples")
33  my_tuple=(1,2,3,4,5)
34  t1=(2,3,4)
35  t2=(5,6,7)
36  t2=t1+t2
37  print(t2)
38  print(t1+t2)

```

```



Tuples
(2, 3, 4, 5, 6, 7)
(2, 3, 4, 2, 3, 4, 5, 6, 7)
PS G:\sem-3\python_lab>

```

Post Lab Exercise:

- Write a program that displays "Welcome to Python" five times.
- Write a program that displays the following table:

a	a ²	a ³
1	1	1
2	4	8
3	9	27
4	16	64

 Marwadi University Marwadi Chandarana Group 	Marwadi University Faculty of Engineering & Technology Department of Information and Communication Technology	
Subject: Programming With Python (01CT1309)	Aim: Write a program to demonstrate different number datatypes in python.	
Experiment No: 01	Date:07/07/2025	Enrollment No: 92400133037

c. Write a program that displays the result of

$$\frac{9.5 \times 4.5 - 2.5 \times 3}{45.5 - 3.5}$$

```
lab1 > PostLab.py > ...
1  print("a:")
2  print(["Welcome to Python\n"*5])
3
4  print("b:")
5  print("a    a^2   a^3")
6  print("1    1    1")
7  print("2    4    8")
8  print("3    9    27")
9  print("4    16   64")
10
11 print("c:")
12 num=(9.5*4.5)-(2.5*3)
13 den=45.5-3.5
14 print(num/den)
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

✓ TERMINAL

```
PS G:\sem-3\python_lab> python -u "g:\sem-3\python_lab\lab1\PostLab.py"
a:
Welcome to Python
Welcome to Python
Welcome to Python
Welcome to Python
Welcome to Python

b:
a    a^2   a^3
1    1    1
2    4    8
3    9    27
4    16   64

c:
0.8392857142857143
```