

Name: Taha Arshad

Class: BSIT

Section: B

Lab title : Python Lab Task

1. Numeric Types

Program 1: int Example

```
a = 5
b = 10
sum_result = a + b
print("The sum of", a, "and", b, "is:", sum_result)
```

Output:

The sum of 5 and 10 is: 15

Program 2: float Example

```
radius = 3.5
pi = 3.14
area = pi * radius * radius
print("The area of the circle is:", area)
```

Output:

The area of the circle is: 38.465

2. Sequence Types

Program 1: str Example

```
name = "John"
greeting = "Hello " + name + "!"
print(greeting)
```

Output:

Hello John!

Program 2: list Example

```
fruits = ['apple', 'banana', 'cherry']
print("Fruits:", fruits)
fruits.append('orange')
print("Updated fruits:", fruits)
```

Output:

Fruits: ['apple', 'banana', 'cherry']
Updated fruits: ['apple', 'banana', 'cherry', 'orange']

Program 3: tuple Example

```
coordinates = (10, 20)
print("Coordinates:", coordinates)
# Tuples are immutable, so trying to modify them will raise an error
```

Output:

Coordinates: (10, 20)

3. Mapping Type

Program 1: dict Example

```
person = {'name': 'Alice', 'age': 25}
print("Name:", person['name'])
print("Age:", person['age'])
```

Output:

Copy code
Name: Alice
Age: 25

Program 2: Adding a new key-value pair to the dictionary

```
person['city'] = 'New York'
print("Updated person:", person)
```

Output:

Updated person: {'name': 'Alice', 'age': 25, 'city': 'New York'}

4. Set Types

Program 1: set Example

```
numbers = {1, 2, 3, 4}
print("Numbers:", numbers)
numbers.add(5)
print("Updated set:", numbers)
```

Output:

Numbers: {1, 2, 3, 4}
Updated set: {1, 2, 3, 4, 5}

Program 2: Removing an element from a set

```
numbers.remove(2)  
print("After removing 2:", numbers)
```

Output:

After removing 2: {1, 3, 4, 5}

5. Boolean Type

Program 1: bool Example

```
is_sunny = True  
if is_sunny:  
    print("It's a sunny day!")  
else:  
    print("It's not sunny today.")
```

Output:

It's a sunny day!

Program 2: Using a comparison to generate a boolean

```
a = 10  
b = 5  
result = a > b  
print("Is a greater than b?", result)
```

Output:

Is a greater than b? True

Shapes using Strings

1. Square Shape

```
print("*" * 5)  
print("*" * 5)  
print("*" * 5)  
print("*" * 5)  
print("*" * 5)
```

Output:

```
*****
*****
*****
*****
*****
```

2. Right Triangle

```
print("*" * 1)
print("*" * 2)
print("*" * 3)
print("*" * 4)
print("*" * 5)
```

Output:

```
*
**
***
****
*****
```

3. Left-Aligned Triangle

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

Output:

```
  *
  **
 ***
****
*****
```

4. Pyramid

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

Output:

```
  *
 ***
*****
*****
*****
```

5. Diamond Shape

```
print("  *")
print(" ***")
print("*****")
print(" ***")
print("  *")
```

Output:

```
  *
 ***
*****
 ***
  *
```

6. Hourglass Shape

```
print("*****")
print(" ***** ")
print("  *** ")
print("   * ")
```

Output:

```
*****
*****
***
*
```

7. Hollow Square

```
print("*****")
print("*  *")
print("*  *")
print("*  *")
print("*****")
```

Output:

```
*****
*  *
*  *
*  *
```

```
*****
print("*" * 10)
print("*" * 10)
print("*" * 10)
```

Output:

```
*****
*****
*****
```

9. Double Triangle

```
print(" * ")
print(" *** ")
print(" ***** ")
print("*****")
print(" * ")
print(" *** ")
print(" ***** ")
print("*****")
```

Output:

```
 *
***
*****
*****
 *
***
*****
*****
```

10. Zigzag

```
print(" *")
print(" *")
print("")
print(" *")
print(" *")
```

Output:

```
 *
 *
 *
 *
 *
```