

# COMPUTER ASSINGMENT-1

**Q1. Create one variable containing following type of data:**

(i) string

(ii) list

(iii) float

(iv) tuple

Ans . # (i) String

```
my_string = "Hello, World!"
```

# (ii) List

```
my_list = [1, 2, 3, 4, 5]
```

# (iii) Float

```
my_float = 3.14
```

# (iv) Tuple

```
my_tuple = (10, 20, 30)
```

**Q2. Given are some following variables containing data:**

(i) `var1 = ' '`

(ii) `var2 = '[ DS , ML , Python]'`

(iii) `var3 = [ 'DS' , 'ML' , 'Python' ]`

(iv) `var4 = 1.`

**What will be the data type of the above given variable.**

ANS . (i) `var1 = ' '` - This contains a string with a single space character. The data type is a string.

(ii) `var2 = '[ DS , ML , Python]'` - This contains a string with square brackets and elements separated by commas. It looks like a string representation of a list, but since it's within quotes, it's considered as a string data type.

(iii) `var3 = [ 'DS' , 'ML' , 'Python' ]` - This contains a list with three string elements. The data type is a list.

(iv) `var4 = 1.` - This contains a floating-point number. The data type is a float. Note that the presence of a decimal point makes Python interpret it as a float, not an integer.

**Q3. Explain the use of the following operators using an example:**

(i) /

(ii) %

(iii) //

(iv) \*\*

ANS . i) / (Division Operator):

- The division operator (/) is used to perform division between two numbers.
- It returns the quotient as a floating-point number.
- Example:

```
result = 10 / 3
```

```
print(result) # Output: 3.3333333333333335
```

(ii) % (Modulus Operator):

- The modulus operator (%) returns the remainder when one number is divided by another.
- Example:

```
remainder = 10 % 3
```

```
print(remainder) # Output: 1
```

(iii) // (Floor Division Operator):

- The floor division operator (//) returns the quotient of the division, rounded down to the nearest integer.
- It performs integer division, discarding any fractional part.

- Example:

```
result = 10 // 3
```

```
print(result) # Output: 3
```

(iv) **\*\*** (Exponentiation Operator):

- The exponentiation operator (\*\*) raises the first operand to the power of the second operand.
- It performs exponentiation.
- Example:

```
result = 2 ** 3
```

```
print(result) # Output: 8
```

**Q4. Create a list of length 10 of your choice containing multiple types of data. Using for loop print the element and its data type.**

ANS . my\_list = [10, 3.14, 'hello', True, [1, 2, 3], {'a': 1, 'b': 2}, (4, 5), None, False, 5+3j]

for element in my\_list:

```
    print(f"Element: {element}, Type: {type(element)}")
```

**Q5. Using a while loop, verify if the number A is purely divisible by number B and if so then how many times it can be divisible.**

ANS . # Taking input for numbers A and B

```
A = int(input("Enter the number A: "))
```

```
B = int(input("Enter the number B: "))
```

```
# Initializing a variable to count how many times A can be divided by B
```

```
divisions = 0
```

```
# Verifying if A is purely divisible by B and counting the divisions
```

```
while A % B == 0:
```

```
    A = A / B
```

```
    divisions += 1
```

```
if divisions > 0:
```

```
print(f"{A} is divisible by {B} {divisions} times.")
```

else:

```
print(f"{A} is not divisible by {B}.")
```

**Q7. What do you understand about mutable and immutable data types? Give examples for both showing this property.**

**ANS** **Immutable Data Types:**

Immutable data types are those whose values cannot be changed after they are created. If you modify an immutable object, you're actually creating a new object in memory. Examples of immutable data types include integers, floats, strings, tuples, and frozensets.

**Mutable Data Types:**

Mutable data types, on the other hand, are those whose values can be changed after they are created. This means that you can modify the object without creating a new one. Examples of mutable data types include lists, dictionaries, sets, and byte arrays.

**Q6. Create a list containing 25 int type data. Using for loop and if-else condition print if the element is divisible by 3 or not.**

**ANS** . # Create a list containing 25 integers

```
int_list = [7, 12, 4, 9, 15, 28, 33, 21, 18, 5, 30, 11, 6, 20, 27, 8, 3, 17, 22, 13, 25, 10, 36, 14, 19]
```

```
# Using for loop and if-else condition to check divisibility by 3
```

```
for num in int_list:
```

```
    if num % 3 == 0:
```

```
        print(f"{num} is divisible by 3.")
```

```
    else:
```

```
        print(f"{num} is not divisible by 3.")
```

