### Python:

What is the data type of the result in the following expression: 10 / 2?

* a. int  
  b. float  
  c. str  
  d. bool

Which data type is used to represent a sequence of characters in Python?

* a. int  
  b. float  
  c. str  
  d. list

What is the output of bool("False")?

* a. False  
  b. True  
  c. TypeError  
  d. None

In Python, which data type is used to store an ordered collection of elements with no duplicate values?

* a. tuple  
  b. list  
  c. set  
  d. dictionary

What is the result of the expression 3 \*\* 2?

* a. 5  
  b. 6  
  c. 9  
  d. 27

What does the % operator do in Python?

* a. Exponentiation  
  b. Floor division  
  c. Modulus  
  d. Multiplication

What is the result of the expression 5 // 2?

* a. 2.5  
  b. 2  
  c. 3  
  d. 2.0

In Python, how is a block of code inside an if statement defined?

* a. By indentation  
  b. By braces {}  
  c. By parentheses ()  
  d. By square brackets []

What is the purpose of the elif keyword in Python?

* a. It signifies the end of an if statement.  
  b. It is used to catch exceptions.  
  c. It is an abbreviation for "else if" and is used for multiple conditions.  
  d. It represents a loop in Python.

In a for loop in Python, what does the range(5) represent?

* a. The numbers 0 to 5 (inclusive)  
  b. The numbers 1 to 5 (inclusive)  
  c. The numbers 0 to 4 (inclusive)  
  d. The numbers 1 to 4 (inclusive)

What is the output of int("10")?

* a. 10  
  b. "10"  
  c. TypeError  
  d. None

Which data type is mutable (can be modified after creation) in Python?

* a. int  
  b. float  
  c. str  
  d. list

In Python, how do you check the type of a variable?

* a. typeof(var)  
  b. typeOf(var)  
  c. type(var)  
  d. var.type()

What is the result of the expression 5 != 5?

* a. True  
  b. False  
  c. Error  
  d. None

What is the purpose of the and operator in Python?

* a. Logical AND  
  b. Bitwise AND  
  c. Assignment AND  
  d. Concatenation

Which operator is used for string concatenation in Python?

* a. +  
  b. -  
  c. \*  
  d. /

What does the in operator do in Python?

* a. Membership test for lists and strings  
  b. Exponentiation  
  c. Bitwise AND  
  d. Modulus

In Python, what is the purpose of the else clause in an if statement?

* a. It is used for error handling.  
  b. It contains the main block of code.  
  c. It is executed when the if condition is True.  
  d. It is executed when the if condition is False.

How can you terminate a loop prematurely in Python?

* a. stop statement  
  b. end statement  
  c. break statement  
  d. terminate statement

What is the purpose of the pass statement in Python?

* a. It indicates the end of a code block.  
  b. It is a placeholder and does nothing.  
  c. It is used to define a function.  
  d. It is a comment.

What is the result of the expression 8 // 3?

* a. 2.67  
  b. 2.0  
  c. 2  
  d. 2.5

What is the purpose of the or operator in Python?

* a. Logical OR  
  b. Bitwise OR  
  c. Assignment OR  
  d. Concatenation

Which operator is used for exponentiation in Python?

* a. \*\*  
  b. ^  
  c. //  
  d. %

In Python, what is the purpose of the elif clause in an if statement?

* a. It is executed when the if condition is True.  
  b. It is used for error handling.  
  c. It is a short form of "else if" and is used for multiple conditions.  
  d. It contains the main block of code.

What is the output of the following code snippet?

num = 7

if num % 2 == 0:

print("Even")

elif num % 3 == 0:

print("Divisible by 3")

else:

print("Odd")

* a. Even  
  b. Divisible by 3  
  c. Odd  
  d. None

num = 7

if num % 2 == 0:

print("Even")

elif num % 3 == 0:

print("Divisible by 3")

else:

print("Odd")

programs:

Task 1: Arithmetic Operators

# Write a program that takes two numbers from the user and performs the following operations:

# - Addition

# - Subtraction

# - Multiplication

# - Division

**Syntax:**

a=30

b=20

print(a+b)

print(a-b)

print(a\*b)

print(a/b)

**output:**

50

10

600

1.5

# Task 2: Logical Operators

# Write a program that asks the user for their age.

# - If the age is less than 18, print "You are a minor."

# - If the age is 18 or older, print "You are an adult."

**Syntax:**

age=26

if age<=18:

print("you are minor")

else:

print("you are adult")

**output:**

you are adult

# Task 3: Comparison Operators

# Write a program that compares two strings entered by the user.

# - If the strings are equal, print "Strings are equal."

# - If not, print "Strings are not equal."

**Syntax:**

str1=30

str2=25

if str1==str2:

print("strings are equal")

else:

print("strings are not equal")

**output:**

strings are not equal

# Task 4: While Loop

# Write a program that uses a while loop to print the numbers from 1 to 5.

**Syntax:**

x = 1

while x<= 10:

print(x)

x+= 1

**output:**

1

2

3

4

5

6

7

8

9

10

# Task 5: For Loop

# Write a program that uses a for loop to iterate over a list of fruits and print each fruit.

**Syntax:**

fruits={"apple","orange","mango","banana","sapota","papaya"}

for fruits in fruits:

print(fruits)

**output:**

papaya

apple

mango

banana

sapota

orange

# Task 6: Lists

# Create a list of numbers and perform the following operations:

# - Add a new number to the list.

# - Remove an existing number from the list.

**Syntax:**

x = [100,101,103,104,105,106]

print("Original list of numbers:", x)

y= 108

x.append(y)

print("List after adding", y, ":", x)

number\_to\_remove = 102

if number\_to\_remove in x:

x.remove(number\_to\_remove)

print("List after removing", number\_to\_remove, ":", x)

else:

print(number\_to\_remove, "is not in the list.")

**output:**

Original list of numbers: [100, 101, 103, 104, 105, 106]

List after adding 108 : [100, 101, 103, 104, 105, 106, 108]

102 is not in the list.

# Task 7: Dictionaries

# Create a dictionary representing a person with attributes like name, age, and city.

# - Print the person's information.

# - Add a new attribute (e.g., occupation) to the dictionary.

**Syntax:**

person = {

"name": "Sasi",

"age": 27,

"city": "Hyderabad"

}

print("Person's Information:")

for key, value in person.items():

print(f"{key.capitalize()}: {value}")

person["occupation"] = "Engineer"

print("\nPerson's Information after adding occupation:")

for key, value in person.items():

print(f"{key.capitalize()}: {value}")

**output:**

Person's Information:

Name: Sasi

Age: 27

City: Hyderabad

Person's Information after adding occupation:

Name: Sasi

Age: 27

City: Hyderabad

Occupation: Engineer

# Bonus Task: Combine Control Statements and Operators

# Write a program that asks the user to enter two numbers.

# - If the sum of the numbers is greater than 10, print "Sum is greater than 10."

# - If the sum is less than or equal to 10, print "Sum is less than or equal to 10."

**Syntax:**

x = 5

y = 4

sum\_of\_numbers = x+ y

if sum\_of\_numbers > 10:

print("Sum is greater than 10.")

else:

print("Sum is less than or equal to 10.")

**output:**

Sum is less than or equal to 10.