**Assignment - Python Basic Constructs**

**Question 1**

1. Using Python script as a calculator Create the variables n, r, p and assign them values 10, 5, and 100 respectively. Then evaluate the following expression in the Python console.

𝐴 = 𝑝 (1 + 𝑟/ 100)n

1. 100
2. **162.89**
3. 189
4. None of the above

**Code:**

# Assigning values to variables

n = 10

r = 5

p = 100

# Calculating A

A = p \* (1 + r / 100) \*\* n

# Printing the result

print("The result of the expression is:", A)

**Answer:**

The result of the expression is: 162.8894626777442

**Question 2**

1. In a given string format operation, how will you print the given string.

A = 10

B = 20

Str = "There are {} students in the class, with {} who play at least one sport."

a. print(string.format(a,b))

b. print(string+a+b)

c. print(string.format(b,a))

**d. None of the above**

**Code:**

A = 10

B = 20

string = "There are {} students in the class, with {} who play at least one sport."

print(string.format(a, b))

**Answer:**

**d. None of the above**

**Question 3**

1. In a given sample string, How do you print a double quoted string in between a regular string using the escape character?

Sample output = It goes without saying, “Time is Money”, and none can deny it.

**a. print(“It goes without saying, \“Time is Money\”, and none can deny it.”)**

b. print(“It goes without saying, \Time is Money, and none can deny it.”)

c. print(“It goes without saying” + “Time is Money” + “and none can deny it.”)

d. None of the above.

**Code:**

print("It goes without saying, \"Time is Money\", and none can deny it.")

**Answer:**

It goes without saying, "Time is Money", and none can deny it.

**Question 4**

1. What will be the output of the following code?

x = lambda a,b: a//b x(10,3)

a. 3.3333333333

**b. 3**

c. 30

d. 1000

**Code:**

x = lambda a, b: a // b

print(x(10, 3))

**Answer:**

3

**Question 5**

1. What will be the output of the following code?

A = 10

B = 12

print("Smaller") if A == B else print("Greater") if A < B else print("True")

a. True

b. Smaller

**c. Greater**

d. None of the above

**Code:**

A = 10

B = 12

print("Smaller") if A == B else print("Greater") if A < B else print("True")

**Answer:**

Greater

**Question 6**

1. What will be the output of the following code?

import os

import numpy as np

mylist1 = [2, 7, 3, 5, 4, 6]

print(mylist1)

arr\_1 = numpy.array(mylist1, dtype = int) print(arr\_1)

a. [2 7 3 5 4 6]

b. TypeError

**c. NameError: name 'numpy' is not defined**

d. None of the above

**Code:**

import os

  import numpy as np

  mylist1 = [2, 7, 3, 5, 4, 6]

  print(mylist1)

  arr\_1 = numpy.array(mylist1, dtype = int)

  print(arr\_1)

**Answer:**

NameError: name 'numpy' is not defined

**Question 7**

1. Create a string called ‘string’ with the value as “Machine Learning”. Which code(s) is/are appropriate to slice the substring “Learn”?

a. string[slice(13,8,1)]

b. string[slice(1,8,1)]

**c. string[8:14]**

**d. string[slice(8,13,1)]**

**Code:**

# Create the string

string = "Machine Learning"

# Option a

substring\_a = string[slice(13, 8, 1)]

print("Option a:", substring\_a)

# Option b

substring\_b = string[slice(1, 8, 1)]

print("Option b:", substring\_b)

# Option c

substring\_c = string[8:14]

print("Option c:", substring\_c)

# Option d

substring\_d = string[slice(8, 13, 1)]

print("Option d:", substring\_d)

**Answer:**

Option a:

Option b: achine

Option c: Learni

Option d: Learn

**Question 8**

1. Create a sequence of numbers from 10 to 25 and increment by 4. What is the index of the value 18?

a. 3

**b. 2**

c. 0

**Code:**

# Create the sequence of numbers

sequence = list(range(10, 26, 4))  # [10, 14, 18, 22]

# Find the index of the value 18

index\_of\_18 = sequence.index(18)

print("Index of value 18 is:", index\_of\_18)

**Answer:**

Index of value 18 is: 2

**Question 9**

1. Which of the following is true with respect to the below codes?

num1 = 5\*\*4

num2 = pow(5,4)

print(num1, num2)

**a. num1 = num2**

b. num1 ≠ num2

c. num1 < num2

d. num1 > num2

**Code:**

num1 = 5\*\*4

num2 = pow(5,4)

print(num1, num2)

**Answer:**

625 625

**Question 10**

1. A Python NameError exception is raised when: -
2. **Trying to access a variable which has not been defined**
3. Trying to access a key in a dictionary that does not exist
4. Accessing a column with misspelled column name
5. Accessing the function from a module that has not been imported

**Answer:**

1. Trying to access a variable which has not been defined

**Question 11**

1. What type of exception will be raised for the code given below?

x= "string"

int(x)

a. NameError

b. KeyError

**c. ValueError**

d. AttributeError

Answer:

ValueError: invalid literal for int() with base 10: 'string'

**Question 12**

1. A FileNotFoundError exception is raised by operating system errors when: -
2. Trying to create a file or directory which already exists
3. **A file or directory is requested but does not exist in the working directory**
4. Trying to run an operation without the adequate access rights
5. A directory operation, os.listdir() is requested on something which is not a directory

**Answer:**

b. A file or directory is requested but does not exist in the working directory

**Question 13**

1. Consider a variable Z. The value of Z is "ID-5632". Data type of Z is: -

a. Complex

**b. Character**

c. Integer

d. Boolean

**Answer:**

b. Character

**Question 14**

1. Which of the following variable(s) are character data type?

a. K= “4”

b. J= “Welcome”

c. L= “?”

**d. All of the above**

**Answer:**

d. All of the above

**Question 15**

1. Choose the symbol/s that does not have the ability to convert any values to string?

a. ( )

b. “ ”

c. {}

**d. #**

**Answer:**

d. #

**Question 16**

1. Create a dictionary ‘Country’ that maps the following countries to their capitals respectively:

Country India China Japan Qatar France State Delhi Beijing Tokyo Doha Marseilles

Find 2 commands to replace “Marseilles” with “Paris” is:

**Code:**

# Create the dictionary Country

Country = {

    'India': 'Delhi',

    'China': 'Beijing',

    'Japan': 'Tokyo',

    'Qatar': 'Doha',

    'France': 'Marseilles'

}

# Replace "Marseilles" with "Paris"

Country['France'] = 'Paris'

# Print the updated dictionary

print(Country)

**Answer:**

{'India': 'Delhi', 'China': 'Beijing', 'Japan': 'Tokyo', 'Qatar': 'Doha', 'France': 'Paris'}

**Code:**

Country.update({'France': 'Paris'})

print(Country)

**Answer:**

{'India': 'Delhi', 'China': 'Beijing', 'Japan': 'Tokyo', 'Qatar': 'Doha', 'France': 'Paris'}

**Question 17**

1. Create the tuples given below

tuple\_1 = (1,5,6,7,8)

tuple\_2 = (8,9,4)

Identify which of the following code does not work on a tuple.

a. sum(tuple\_1)

b. len(tuple\_2)

c. tuple\_2 + tuple\_1

**d. tuple\_1[3] = 45**

Code:

# Create the tuples

tuple\_1 = (1, 5, 6, 7, 8)

tuple\_2 = (8, 9, 4)

# Option a: sum(tuple\_1)

result\_a = sum(tuple\_1)

print("Result of sum(tuple\_1):", result\_a)

# Option b: len(tuple\_2)

result\_b = len(tuple\_2)

print("Length of tuple\_2:", result\_b)

# Option c: tuple\_2 + tuple\_1

result\_c = tuple\_2 + tuple\_1

print("Concatenation of tuple\_2 and tuple\_1:", result\_c)

# Option d: tuple\_1[3] = 45

result\_d = tuple\_1[3] = 45

print("The output will be:", result\_d)

**Answer:**

Result of sum(tuple\_1): 27

Length of tuple\_2: 3

Concatenation of tuple\_2 and tuple\_1: (8, 9, 4, 1, 5, 6, 7**, 8)**

**TypeError: 'tuple' object does not support item assignment**

**Question 18**

1. How many elements in the following data structure?

s = {1, 2, 3, 4, 4, 4, 5, 6}

**Code:**

s = {1, 2, 3, 4, 4, 4, 5, 6}

print("Number of elements in the set:", len(s))

**Answer:**

Number of elements in the set: 6

**Question 19**

1. Write a function which finds all pythagorean triplets of triangles whose sides are no greater than a natural number N.

**Code:**

def find\_pythagorean\_triplets(N):

    triplets = []

    for a in range(1, N+1):

        for b in range(a, N+1):

            c = (a\*\*2 + b\*\*2) \*\* 0.5

            if c.is\_integer() and c <= N:

                triplets.append((a, b, int(c)))

    return triplets

# Example usage:

N = 10

triplets = find\_pythagorean\_triplets(N)

print("Pythagorean triplets with sides no greater than", N, ":", triplets)

**Answer:**

Pythagorean triplets with sides no greater than 10 : [(3, 4, 5), (6, 8, 10)]