```
###Q1.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
```

```
n = 10
r = 5
p = 100
```

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

```
162.8894626777442
```

```
###Q2. 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
```

```
A = 10
B = 20
string = "There are {} students in the class, with {} who play at least one sport."
```

```
print(string.format(B,A))
```

```
There are 20 students in the class, with 10 who play at least one sport.
```

```
###Q3. 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.
```

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

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

```
Start coding or generate with AI.
```

```
##Q4. What will be the output of the following code? ans: b-3
x = lambda a,b: a//b
x(10,3)
```

```
3
```

```
##Q5. What will be the output of the following code? ans: c-Greater
A = 10
B = 12
print("Smaller") if A == B else print("Greater") if A < B else print("True")
```

```
Greater
```

```
##Q6. What will be the output of the following code? ans: a
import os
import numpy as np
my_list1 = [2,7,3,5,4,6]
```

```
print(my_list1)
```

```
[2, 7, 3, 5, 4, 6]
```

```
arr_1 = np.array(my_list1, dtype = int)
print(arr_1)
```

```
[2 7 3 5 4 6]
```

##Q7. Create a string called ‘string’ with the value as “Machine Learning”. Which code(s) is/are appropriate to slice the substring “Learn”? ans: d

```
string = "Machine Learning"
```

```
string[slice(8,13,1)]
```

```
'Learn'
```

##Q8. Create a sequence of numbers from 10 to 25 and increment by 4. What is the index of the value 18? ans:b-2

```
# Create a sequence of numbers from 10 to 25
Lst = [i for i in range(10, 26,4)]

indx = Lst.index(18)
print("The index of value 18 is {}".format(indx))
```

```
The index of value 18 is 2
```

Start coding or generate with AI.

##Q9. Which of the following is true with respect to the below codes? ans: a- num1 = num2

```
num1 = 5**4
num2 = pow(5,4)
print(num1, num2)
```

```
625 625
```

##Q10.A Python NameError exception is raised when: -
ans: a. Trying to access a variable which has not been defined

Start coding or generate with AI.

##Q11.What type of exception will be raised for the code given below? ans: c. ValueError

```
x = "string"
int(x)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-5-6107b025c2b2> in <cell line: 4>()
      2
      3 x = "string"
----> 4 int(x)

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

##Q12.A FileNotFoundError exception is raised by operating system errors when:
ans:b. A file or directory is requested but does not exist in the working directory

Start coding or generate with AI.

##Q13.Consider a variable Z. The value of Z is "ID-5632". Data type of Z is: -
ans: b. Character

```
import numpy as np
Z = "ID-5632"
print(type(Z))
```

```
<class 'str'>
```

```
##Q14.Which of the following variable(s) are character data type?
ans: d. All of the above
```

```
K = "4"
J = "Welcome"
L = "?"
```

```
print(type(K), type(J), type(L))
```

```
<class 'str'> <class 'str'> <class 'str'>
```

```
##Q15.Choose the symbol/s that does not have the ability to convert any values to string?
ans: d. #
```

```
Start coding or generate with AI.
```

```
##Q16.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
```

```
Country={"India":"Delhi","China":"Beijing","Japan":"Tokyo","Qatar":"Doha","France":"Marseilles"}
Country
```

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

```
varr=Country["France"].replace("Marseilles","Paris")
Country["France"]=varr
```

```
Country
```

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

```
Start coding or generate with AI.
```

```
##Q17. 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. ans:d.tuple_1[3] = 45
```

```
sum(tuple_1)
```

```
27
```

```
len(tuple_2)
```

```
3
```

```
tuple_2 + tuple_1
```

```
(8, 9, 4, 1, 5, 6, 7, 8)
```

```
tuple_1[3] = 45
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-35-bf6065b4ce4b> in <cell line: 1>()
----> 1 tuple_1[3] = 45

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

##Q18. How many elements in the following data structure?
s = {1,2,3,4,4,4,5,6}

```
len(s)
```

```
6
```

##Q19.Write a function which finds all pythagorean triplets of triangles whose sides are no greater than a natural number N

```
def pythagoreanTriplets(limits) :
    c, m = 0, 2

    while c < limits :
      for n in range(1, m) :
            a = m * m - n * n
            b = 2 * m * n
            c = m * m + n * n
            if c > limits :
                break
            print(a, b, c)
      m = m + 1

if __name__ == '__main__' :
  limit = 20
  pythagoreanTriplets(limit)
```

```
3 4 5
8 6 10
5 12 13
15 8 17
12 16 20
```