18CSC207J Advanced Programming Practice

CT 1   Total marks : 25

PART A   05 X 01 = 05

1 . What do we use to define a block of code in Python language?
        a. Key
        b. Square Brackets
        c. Indentation
        d. Round brackets

2. Which of the following declarations is incorrect in python language?
        a. xyzp = 5,000,000
        b. x y z p = 5000 6000 7000 8000
        c. x,y,z,p = 5000, 6000, 7000, 8000
        d. x_y_z_p = 5,000,000

3. Choose the correct option with respect to Python.

        A. Both tuples and lists are immutable.
        B. Tuples are immutable while lists are mutable.
        C. Both tuples and lists are mutable.
        D. Tuples are mutable while lists are immutable.

4. What will be the output of below Python code?
        tuple1=(5,1,7,6,2)
        tuple1.pop(2)
        print(tuple1)

        A. (5,1,6,2)
        B. (5,1,7,6)
        C. (5,1,7,6,2)
        D. Error

5. Which of the following is False with respect Python code?

class Student:

    def __init__(self,id,age):

        self.id=id

        self.age=age

std=Student(1,20)

        A. "std" is the reference variable for object Student(1,20)
        B. id and age are called the parameters.
        C. Every class must have a constructor.
        D. Both A & B

PART-B

6)

```
n=int(input("Enter the number "))
for i in range (1,n+1):
    print(i*i*i,end=" ")
```

```
Enter the number 5
1 8 27 64 125
```

7)

```
x=input("Enter the string ")
y=x[1::2]
print("Characters present at odd index position are :")
for i in y:
    print(i)
```

```
Enter the string Someone
Characters present at odd index position are :
o
e
n
```

8)

```
def func(name,age,regno,dept):
    print("Name is",name)
    print("Age is",age)
    print("Reg no is",regno)
    print("Department is",dept)
Name="Someone"
Age=19
Regno=420
Dept="Cse"
func(Name,Age,Regno,Dept)
```

```
Name is Someone
Age is 19
Reg no is 420
Department is Cse
```

9)
```python
class Vehicle:
    def __init__(self,max_speed,mileage):
        self.max_speed=max_speed
        self.milegae=mileage
    def display(self):
        print("Maximum speed is : ",self.max_speed,"Km/hr")
        print("Milegae is :",self.milegae,"km/ltr")
obj=Vehicle(200,20)
obj.display()
```

```
Maximum speed is :  200 Km/hr
Milegae is : 20 km/ltr
```

10)
```python
#Here branch is defualt argument as it takes defualt value if value not passed in function call
def printinfo(name,branch="CSE"):
    print("Name is :",name)
    print("Branch is :",branch)
printinfo("Someone","ECE")
printinfo("Anyone")
```

```
Name is : Someone
Branch is : ECE
Name is : Anyone
Branch is : CSE
```

PART-C

11)
```python
def recfun(n):
    if n==0:
        return 0
    else:
        return n+recfun(n-1)
print("Sum of number from 0 to 10 is :",recfun(10))
```

```
Sum of number from 0 to 10 is : 55
```

12)
```python
class Vehicle:
    def __init__(self, capacity):
        self.capacity = capacity
    def fare(self):
        return self.capacity * 100
class Bus(Vehicle):
    def fare(self):
        amt=super().fare()
        amt+=amt* 10 / 100
        return amt
School_bus = Bus(50)
print("Total Bus fare is:", School_bus.fare())
```

```
Total Bus fare is: 5500.0
```

13)
```python
def is_prime(n):
    if n==2 or n==3 or n==5:
        return True
    for i in range(2,n//2):
        if n%i==0:
            return False
        else:
            return True
x=int(input("Enter the Number : "))
if is_prime(x):
    print("It is a Prime Number ")
else:
    print("It is not a Prime Number ")
```

```
Enter the Number : 31
It is a Prime Number
```

14)
```python
class SRMIST:
    school = 'School of computing'
    dep1= 'cse'
    dep2="ai"
    dep3="csbs"
print("Original attributes and their values of the Student class:")
for attr, value in SRMIST.__dict__.items():
    if not attr.startswith('_'):
        print(f'{attr} -> {value}')
print("\nAfter adding the student_class, attributes and their values with the said class:")
SRMIST.specializtion  = 'CSE CORE'
for attr, value in SRMIST.__dict__.items():
    if not attr.startswith('_'):
        print(f'{attr} -> {value}')
print("\nAfter removing the student_name, attributes and their values from the said class:")
del SRMIST.dep1
del SRMIST.dep2
for attr, value in SRMIST.__dict__.items():
    if not attr.startswith('_'):
        print(f'{attr} -> {value}')
```

```
Original attributes and their values of the Student class:
school -> School of computing
dep1 -> cse
dep2 -> ai
dep3 -> csbs

After adding the student_class, attributes and their values with the said class:
school -> School of computing
dep1 -> cse
dep2 -> ai
dep3 -> csbs
specializtion -> CSE CORE

After removing the student_name, attributes and their values from the said class:
school -> School of computing
dep3 -> csbs
specializtion -> CSE CORE
```

18CSC207J Advanced Programming Practice

CT 1

PART A   **Total marks : 25**

05 X 01 = 05

1. Which of the following statements is correct for variable names in Python language?
   a. All variable names must begin with an underscore.
   b. Unlimited length
   c. The variable name length is a maximum of 2.
   d. Both a & b

2. What is the output of the following code

   aList = ["SRMIST", [4, 8, 12, 16]]
   print(aList[0][1],aList[1][3])

   a) **R 16**
   b) M 12
   c) I 4
   d) S 16

2. Choose the correct option.

   A. In Python, a tuple can contain only integers as its elements.
   B. In Python, a tuple can contain only strings as its elements.
   C. In Python, a tuple can contain both integers and strings as its elements.
   D. In Python, a tuple can contain either string or integer but not both at a time.

3. What will be the output of the following Python code?
   str1 = 'SOC'
   str2 = ','
   str3 = 'CTech'
   str1 = str1+str2+str3
   print(str1[-1:])

   A. C
   B. c
   C. T
   D. H

4. How many objects and reference variables are there for the given Python code?

   class A:
           print("Inside class")
   A()
   A()
   obj=A()
   A. 2 and 1

B. 3 and 3
C. 3 and 1
D. 3 and 2

PARTB

6)

```
x=input("Enter the string ")
y=x[0::2]
print("Characters at odd position in string are : ")
for i in y:
    print(i)
```

```
Enter the string Someone
Characters at odd position in string are
S
m
o
e
```

7)

```
rows = int(input("Enter the no of rows "))
cols = int(input("Enter the no of columns "))
arr=[]
for i in range(rows):
  col = []
  for j in range(cols):
    col.append(i*j)
  arr.append(col)
print(arr)
```

```
Enter the no of rows 3

Enter the no of columns 4
[[0, 0, 0, 0], [0, 1, 2, 3], [0, 2, 4, 6]]
```

8)

```
ls=[None]*10
for i in range(10):
    ls[i]=i
for i in range(10):
    ls[i]=ls[i]*ls[i]
print(ls)
```

```
[0, 1, 4, 9, 16, 25, 36, 49, 64, 81]
```

9)

```
dictio={1:"One",2:"two"}
if 1 in dictio:
    print("Exists")
else:
    print("Doesn't Exists")
```

```
Exists
```

10)

Find the error and explain :
```
class Student:
     schoolName='XYZ School'
    def _init_(self,name,age):
            self.name=name
            self.age=age
```
1) Two double underscores ('_ _') should be there in there in constructor declaration statement .
2)It should not be neglected because underscore changes the scope of the attribute for eg _ _ for private and _
for protected .

PART C

11)
```
x=input("Enter the Password : ")
y=len(x)
up=low=spl=dg=0
for ch in x:
    if ch.isupper():
        up=up+1
    if ch.islower():
        low=low+1
    if ch in "$#@":
        spl=spl+1
    if ch.isdigit():
        dg=dg+1
if y>=6 and y<=16 and up>=1 and low>=1 and spl>=1 and dg>=1:
    print("Valid Password")
else:
    print("Invalid Password")
```

```
Enter the Password : Srmist@2022
Valid Password
```

12)
```python
a = input("Enter the String : ")
alpha=dig=spl= 0
for i in a:
    if i.isalpha():
        alpha=alpha+1
    elif i.isdigit():
        dig=dig+1
    else:
        spl=spl+1
print("\nTotal Number of Alphabets in this String :  ", alpha)
print("Total Number of Digits in this String :  ", dig)
print("Total Number of Special Characters in this String :  ", spl)
```

```
Enter the String : Srmist@2017

Total Number of Alphabets in this String :    6
Total Number of Digits in this String :    4
Total Number of Special Characters in this String :    1
```

13)
```python
class Banks_SRMIST():
    def getBalance(self):
        return 0
class CUB(Banks_SRMIST):
    def getBalance(self):
        return 15000
class HDFC(Banks_SRMIST):
    def getBalance(self):
        return 30000
class Indian_Bank(Banks_SRMIST):
    def getBalance(self):
        return 45000
obj1_499=CUB()
obj2_499=HDFC()
obj3_499=Indian_Bank()
print(obj1_499.getBalance())
print(obj2_499.getBalance())
print(obj3_499.getBalance())
```

```
15000
30000
45000
```

14)
```python
def fun(n):
  while True:
    k = str(n)
    if k == k[::-1]:
      break
    else:
      m = int(k[::-1])
      n += m
  return n
num = int(input("Enter the number : "))
print(fun(num))
```

```
Enter the number : 1234
5555
```