OOPS – 1 Deadline Sep 12

**1.Operator Used**

**Send Feedback**

Which operator in python is used to access properties and methods of an object?

**Options**

1. /
2. ->
3. . answer

Iv. None of the Above

**2. Predict The Output**

What will be the output of following code?

class Student:

name = “Rohan”

age = 16

s1 = Student()

s2 = Student()

print(s1.name,end=” “)

print(s2.name,end=” “)

**i.None None**

**ii.Rohan Rohan answer**

**iii.Rohan None**

**iv.None of the Above**

**3.Predict The Output**

**Send Feedback**

What will be the output of following code?

class Student:

pp = 50

s1 = Student()

s1.pp= 58

s2 = Student()

s2.pp = 60

print(s1.pp)

i.50

ii.58 answer

iii.60

iv.None of the Above

**4.Predict The Output**

**Send Feedback**

What will be the output of following code?

class Student:

name = “Parikh”

def store\_details(self):

self.age = 60

def print\_details(self):

print(self.name, end=” ”)

print(self.age)

s = Student()

s.store\_details()

s.print\_details()

1. Error
2. Parikh 60 answer
3. Parikh None
4. None of the above

**5.Predict The Output**

**Send Feedback**

What will be the output of following code?

class Student:

name = “Parikh”

def store\_details(self):

self.age = 60

def print\_age(self):

print(self.age)

s = Student()

s.store\_details()

s1 = Student()

s1.print\_age()

1. **Error answer**
2. **60**
3. **None**
4. **Parikh 60**

**6. Predict The Output**

**Send Feedback**

What will be the output of following code?

class Student:

def \_\_init\_\_(self,name,age):

self.name = “Rohan”

self.age = 20

def print\_student\_details():

print(self.name, end= “ “)

print(self.age)

s = Student()

s.print\_student\_details()

1. **Rohan 20**
2. **Error answer**
3. **Rohan**
4. **None of the Above**

**7.Predict The Output**

**Send Feedback**

What will be the output of following code?

class Student:

def \_\_init\_\_(self,name,age):

self.name = “Rohan”

self.age = 20

def print\_student\_details():

print(self.name, end= “ “)

print(self.age)

s = Student(“Parikh”,25)

s.print\_student\_details()

1. **Parikh 25**
2. **Rohan 20**
3. **Parikh 20**
4. **Error answer**

**8. Predict The Output**

**Send Feedback**

What will be the output of following code?

class Student:

def \_\_init\_\_(self,name,age):

self.name = name

self.age = age

def print\_student\_details(self):

print(self.name, end= “ “)

print(self.age)

s = Student(“Rohan”,60)

s.print\_student\_details()

1. **Error**
2. **Rohan 60 answer**
3. **Rohan**
4. **None of the Above**

**9. Predict The Output**

**Send Feedback**

What will be the output of this code?

class Student:

def \_\_init\_\_(self,name,age):

self.name = name

self.age = age

def print\_student\_details():

print(self.name, end= “ “)

print(self.age)

@staticmethod

def isTeen(age):

return age>16

a = Student.isTeen(18)

print(a)

1. **True error**
2. **False**
3. **Error**
4. **None of the Above**

**10. Predict The Output**

**Send Feedback**

What will be the output of this code?

class Student:

def \_\_init\_\_(self,name,age):

self.\_\_name = name

self.age = age

def print\_student\_details():

print(self.\_\_name, end= “ “)

print(self.age)

s = Student(“Rohan”,20)

print(s.name)

1. **Rohan**
2. **None**
3. **Error answer**
4. **None of the Above**

**12. Predict The Output**

**Send Feedback**

What will be the output of this code?

class Student:

def \_\_init\_\_(self,name,age):

self.\_\_name = name

self.age = age

def print\_student\_details(self):

print(self.\_\_name, end= “ “)

print(self.age)

s = Student(“Rohan”,20)

s.print\_student\_details()

1. **Rohan 20 answer**
2. **None**
3. **Error**
4. **None of the Above**

**Complex Number**

**Send Feedback**

#### A ComplexNumber class contains two data members : one is the real part (R) and the other is imaginary (I) (both integers).

#### Implement the Complex numbers class that contains following functions -

##### 1. constructor

#### You need to create the appropriate constructor.

##### 2. plus -

#### This function adds two given complex numbers and updates the first complex number.

#### e.g.

if C1 = 4 + i5 and C2 = 3 +i1

C1.plus(C2) results in:

C1 = 7 + i6 and C2 = 3 + i1

##### 3. multiply -

#### This function multiplies two given complex numbers and updates the first complex number.

#### e.g.

if C1 = 4 + i5 and C2 = 1 + i2

C1.multiply(C2) results in:

C1 = -6 + i13 and C2 = 1 + i2

##### 4. print -

#### This function prints the given complex number in the following format :

a + ib

#### Note : There is space before and after '+' (plus sign) and no space between 'i' (iota symbol) and b.

##### Input Format :

Line 1 : Two integers - real and imaginary part of 1st complex number

Line 2 : Two integers - real and imaginary part of 2nd complex number

Line 3 : An integer representing choice (1 or 2) (1 represents plus function will be called and 2 represents multiply function will be called)

##### Output format :

Check details of 'print' function given above.

##### Sample Input 1 :

4 5

6 7

1

##### Sample Output 1 :

10 + i12

##### Sample Input 2 :

4 5

6 7

2

##### Sample Output 2 :

-11 + i58