Module-(5)

Solved Question & Answers on Python Programming

1] Define a function which takes two objects representing complex numbers & returns new complex number with addition of two complex numbers. Define a suitable class 'Complex' to represent the complex number. Develop a program to read N (N>=2) complex numbers & to compute the addition of N complex numbers. [-8M-]

Ans Refer pg no 1.-2.

```
class Complex:
    def __init__(self, real, imag):
        self.real, self.imag = real, imag

    def __add__(self, other):
        return Complex(self.real + other.real, self.imag +
                       other.imag)

    def __str__(self):
        return "{self.real} + {self.imag}"

def add_complex_numbers(complex_numbers):
    result = Complex(0,0)
    for num in complex_numbers:
        result += num  → Add each complex no to the result.
    return result
```

```
# Program to read N complex numbers & compute their addition
N = int (input ("Enter the no of complex numbers (N >= 2) : "))
if N < 2 :
    print ("N must be at least 2")
else :
    complex_numbers = []
    for i in range (N):
        real = float (input ("Enter the real part of complex
                              number (i+1) : "))
        imag = float (input ("Enter the imaginary part of
                              complex number {i+1} : "))
        complex_numbers.append (Complex (real, imag))
    # Compute the addition of all complex numbers
    result = add_complex_number (complex_numbers)
    print ("The sum of the complex number is : {result}")
```

Complex Class :- Represents a complex no with real & imaginary parts

Addition Method :- The _add_ method adds two complex numbers

String Representation :- The _str_ method returns a readable string format like a+bi

User Input :- the program takes the no of complex no 'N' & their values as i/p.

Addition Function :- computes the sum of all entered complex no.

```
Enter the number of complex numbers (N>=2) : 3
Enter the real part of complex numbers 1 : 2
Enter the imaginary part of complex number 1 : 3
Enter the real part of complex number 2 : 1
Enter the imaginary part of complex number 2 : 4
Enter the real part of complex number 3 : 3
Enter the imaginary part of complex number 3 : -1
The sum of the complex number is : 6.0 + 6.0i
```

3] Explain __init__() and __str__() method with examples [6M]

Ans Refer pg no 23 → __init__

Refer pg no 25 → __str__

3] Briefly explain the printing of objects with an examples [6M]

(P) V.V.V Imp

Ans Refer pg no 21 - 23 //

4] Define pure function & modifier. Explain the role of pure functions & modifiers in applications development with suitable python programs. [10M] V.V.V Imp

(P)

Ans Refer pg no 16 - 17

Example for modifiers :-

```
def increment ( time , seconds ) :
    time.second + = seconds.
```

```
    if time.second >= 60 : time.second = 60:
        time.minute += 1

    if time.minute >= 60 : time.minute = 60:
        time.hours += 1.
```

Output :

```
>>> start = Time()
>>> start.hour = 9
>>> start.minute = 45
>>> start.second = 0

>>> done = increment (start, 10)
>>> print_time(done)

    9:45:10
```

Pure Functions :- Refer pg no 15- 17

Example :

```
def add_time (t1, t2):
    sum = Time()
        sum.hour = t1.hour + t2.hour
        sum.minute = t1.minute + t2.minute
        sum.second = t1.second + t2.second.

    if sum.second >= 60 : sum.second = 60
        sum.minute += 1

    if sum.minute >= 60 : sum.minute = 60
        sum.hour += 1
        return sum.
```

Output :-

```
>>> start = Time()
>>> start.hour = 9
>>> start.minute = 45
>>> stard.second = 0

>>> duration = Time()
>>> duration.hour = 1
>>> duration.minute = 35
>>> duration.second = 0

>>> done = add_time(start, duration)
>>> print_time(done)

11:20:00
```

5] What is Polymorphism ? Demonstrate polymorphism with function to find histogram to count the numbers of times each letters appears in a word and in sentence [-10M-].

Ans Refer page no 29 – 30

```
def find_histogram(tex):
    histogram = {}
    for char in text.replace(" ", "").lower(): ↪ Remove spaces for sentences.
        histogram[char] = histogram.get(char,0) + 1
    return histogram

# Example Usage
word = "hello"
sentence = "hello world"
print("Histogram for word", find_histogram(word))
print("Histogram for sentence", find_histogram(sentence))
```

↳ Polymorphism : The same find_histogram function works for both a word & a sentence bcz both are treated as strings

↳ Histogram Calculation :- It counts occurences of each ~~item~~ letter ignoring spaces for sentences.

Example :-

Histogram for word : {'h': 1, 'e': 1, 'l': 2, 'o': 1}

Histogram for sentence : {'h': 1, 'e': 1, 'l': 3, 'o': 2, 'w': 1, 'r': 1, 'd': 1}

6] Write Deck methods to add, remove shuffle and sort cards with illustrating the problems [-10M-]

Ans

```
import random

Class Deck :
    def __init__ (self): self.cards = []
    def add (self, card): self.cards.append (card)
    def remove (self, card): self.cards.remove (card) if card
                in self.cards else print ('{cards}'
                not in deck")

    def shuffle (self): random.shuffle (self.cards)
    def sort (self): self.cards.sort()
    def show (self): print (self.cards)

# Example usage.
    deck = Deck()
    deck.add ("Ace of Spades")
```

```
deck.add("King of Hearts")
deck.add("Queen of diamonds")
deck.add("10 of clubs")
deck.show()

deck.remove("King of Hearts")
deck.show()
deck.shuffle()
deck.show()
deck.sort()
deck.show()
```

Output :-

['Ace of Spades', 'King of Hearts', 'Queen of Diamonds', '10 of clubs']

['Ace of Spades', 'Queen of Diamonds', '10 of clubs']

['Queen of Diamonds', '10 of clubs', 'Ace of Spades']

['10 of clubs', 'Ace of Spades', 'Queen of Diamonds']

7] Write a note on Attributes [-6M-]

Ans Refer page no 2-4

8] Create Rectangle as an object with example [-6M-]

Ans Refer page no 4-5

9] How to write Instances as return values [-8M-]

Ans Refer page no 5-6

10] Write a note on Copying of an object with examples.

Ans Refer page 9-11

11] Give a note on Time Class :-

Ans Refer page no 13-15

12] Write a short note on Debugging.

Ans Refer page no 11-13

13] Give diffce blw Prototyping v/s Planning.

Ans Refer page no 17-19

14] Write a note on OOP.

Ans Refer page no 19-20