

Python Programming

Important Python Practical Programming Questions

1. Write python program to check given number is odd and even number.

```
# Prompt the user to enter a number
number = int(input("Enter a number: "))

# Check if the number is even or odd
if number % 2 == 0:
    print(f"{number} is an even number.")
else:
    print(f"{number} is an odd number.")
```

2. Write a python program to find factorial of positive number.

```
#factorial no
no=int(input("enter any no"))
print(no)
fact=1
temp=no
while no>0:
    fact=fact*no
    no=no-1
print(fact,"is factorial of ",temp)
```

Python Programming

3. Write a python program to find area / Circumference of Circle.

```
import math

# Taking radius input from the user
radius = float(input("Enter the radius of the circle: "))
area = math.pi * radius**2
circumference = 2 * math.pi * radius

print(f"Area of the circle: {area:.2f}")
print(f"Circumference of the circle: {circumference:.2f}")
```

4. Write a python program to print the calendar of January month of the year 2024.

```
import calendar

# Set the desired year and month
year = 2024
month = 1 # January

# Print the calendar
print(calendar.month(year, month))
```

Python Programming

5. Write a python program to calculate number of days between two days.

```
from datetime import date

def numOfDay(date1, date2):
    if date2 > date1:
        return (date2 - date1).days
    else:
        return (date1 - date2).days

date1 = date(2018, 12, 13)
date2 = date(2015, 2, 25)
print(numOfDay(date1, date2), "days")
```

6. Write a python program to enter a number in python and print its Octal and Hexadecimal equivalent.

```
# Input a number
number = int(input("Enter a number: "))

# Convert to octal and hexadecimal
octal_value = oct(number)
hexadecimal_value = hex(number)

print("Octal equivalent: ", octal_value)
print("Hexadecimal equivalent: ", hexadecimal_value)
```

Python Programming

7. Write a python program to find the sum of all odd number till 100.

```
# Initialize the sum
total_sum = 0

# Iterate through numbers from 1 to 100
for num in range(1, 101):
    if num % 2 != 0: # Check if the number is odd
        total_sum += num

print("Sum of odd numbers from 1 to 100: ",total_sum)
```

8. Write a python program to find the sum of any five integers.

```
# Initialize the sum
total_sum = 0

# Input five integers
for _ in range(5):
    num = int(input("Enter an integer: "))
    total_sum += num

print("Sum of the entered integers: ",total_sum)
```

Python Programming

9. Write a python program to find the sum of square of root of any three numbers.

```
import math

# Input three numbers
num1 = float(input("Enter the first number: "))
num2 = float(input("Enter the second number: "))
num3 = float(input("Enter the third number: "))

# Calculate the square root of each number
sqrt1 = math.sqrt(num1)
sqrt2 = math.sqrt(num2)
sqrt3 = math.sqrt(num3)

# Sum the results
sqrt_sum = sqrt1 + sqrt2 + sqrt3

print("The sum of square root of numbers is ",sqrt_sum)
```

10. Write a python program to check the given number is Palindrome number or not.

```
N=int(input("Enter a Number: "))
temp=N
rev=0
while N>0:
    digit=N%10
    rev=rev*10+digit
    N=N//10
if (temp==rev):
    print(temp," is an Palindram Number")
else:
    print(temp," is not Palindram Number")
```

Python Programming

11. Write a python program to check weather a given number is Armstrong number or not.

```
Num=int(input("Enter a Number: "))
Len=len(str(Num))
sum=0
temp=Num
while temp>0:
    digit=temp%10
    sum=sum+digit**Len
    temp=temp//10
if Num==sum:
    print(Num," is an Armstong Number")
else:
    print(Num," is not Armtrong Number")
```

12. Write a python program to print all Armstrong numbers in given range.

```
Num1=int(input("Enter a lower Number: "))
Num2=int(input("Enter a upper Number: "))
for Num in range(Num1,Num2+1):
    Len=len(str(Num))
    sum=0
    temp=Num
    while temp>0:
        digit=temp%10
        sum=sum+digit**Len
        temp=temp//10
    if Num==sum:
        print(Num," is an Armstong Number")
```

Python Programming

13. Write a python program to find factorial of a number.

```
#factorial no
no=int(input("enter any no"))
print(no)
fact=1
temp=no
while no>0:
    fact=fact*no
    no=no-1
print(fact,"is factorial of ",temp)
```

14. Write a python program to find factors of a given number.

```
n=int(input("Enter a number which you want a Factor: "))
print("The Factors of", n,"are: ")
for i in range(1,n+1):
    if n%i==0:
        print(i)
```

Python Programming

15. Write a python program to input two numbers as input and compute the greatest common divisor (GCD).

```
import math

# Input two numbers
num1 = int(input("Enter the first number: "))
num2 = int(input("Enter the second number: "))

# Calculate the GCD
gcd = math.gcd(num1, num2)

print("Greatest Common Divisor (GCD) of ", num1, "and" , num2, "is" , gcd)
```

16. Write a python program to print factorial of a given number by Recursion.

```
def factorial(n):
    if n == 0:
        return 1
    else:
        return n * factorial(n - 1)

# Input from the user
number = int(input("Enter a positive integer: "))

if number < 0:
    print("Factorial is not defined for negative numbers.")
else:
    result = factorial(number)
    print("The factorial of ", number, "is" , result)
```


Python Programming

17. Write a python program to find power of given number by Recursion.

```
def power(base, exponent):
    if exponent == 0:
        return 1
    else:
        return base * power(base, exponent - 1)

# Input from the user
base = float(input("Enter the base: "))
exponent = int(input("Enter the exponent: "))

result = power(base, exponent)
print(base, "raised to the power of ", exponent, "is" , result)
```

18. Write a python program to find the sum of even number between 100 to 500.

```
# Initialize the sum
total_sum = 0

# Iterate through numbers from 100 to 500
for num in range(100, 501):
    if num % 2 == 0: # Check if the number is even
        total_sum += num

print("Sum of odd numbers from 1 to 100: ", total_sum)
```

Python Programming

19. Write a python program to sort the dictionary by keys.

```
# Sample dictionary
my_dict = {'banana': 3, 'apple': 4, 'pear': 1, 'orange': 2}

# Sort the dictionary by keys using sorted() and a for loop
sorted_keys = sorted(my_dict.keys())

sorted_dict = {}
for key in sorted_keys:
    sorted_dict[key] = my_dict[key]

print("Sorted dictionary:", sorted_dict)
```

20. Write a python program to find the sum of values of a dictionary.

```
# Sample dictionary
my_dict = {'banana': 5, 'apple': 4, 'pear': 1, 'orange': 2}

# Use the sum() function to find the sum of all the values in the dictionary
total_sum = sum(my_dict.values())

# Print the result
print("Sum of values:", total_sum)
```

Python Programming

21. Write a python program to test whether a passed letter is vowel or not.

```
# Define a set of vowels
vowels = 'aeiouAEIOU'

# Input from the user
letter = input("Enter a letter: ")

# Check if the input is a single letter
if len(letter) != 1 or not letter.isalpha():
    print("Please enter a single letter.")
else:
    # Check if the letter is in the set of vowels
    if letter in vowels:
        print(f"The letter {letter} is a vowel.")
    else:
        print(f"The letter {letter} is not a vowel.")
```

22. Write a python program to reverse a list of
For example- Input list = [10,11,12,13,14,15]

```
# Sample list
my_list = [10, 11, 12, 13, 14, 15]

# Method 1: Using the reverse() method
my_list_copy1 = my_list.copy() # Creating a copy to maintain original list
my_list_copy1.reverse()
print("Reversed list using reverse():", my_list_copy1)

# Method 2: Using list slicing
reversed_list_slicing = my_list[::-1]
print("Reversed list using slicing:", reversed_list_slicing)
```

Python Programming

23. Write a python program to calculate area of triangle by user input base & height.

```
# Input from the user
base = float(input("Enter the base of the triangle: "))
height = float(input("Enter the height of the triangle: "))

# Calculate the area of the triangle
area = 0.5 * base * height

# Output the result
print(f"The area of the triangle with base {base} and height {height} is {area}")
```

24. Write a python program to count the number of digits in entered number.

```
a=int(input("Enter Any Three or Four digit Number: "))
count=0
while(a>0):
    count=count+1
    a=a//10
print("The Number of digit number: ",count)
```

Python Programming

25. Write a python program to print ASCII code for entered string or message.

```
string=input("Enter String to encode: ")
print("ASCII code for string: ")
for ch in string:
    print(ch,"ASCII code is: ",ord(ch),end="\n")
```

26. Write a python program to check entered number is a perfect or not.

```
n=int(input("Enter a number: "))
sum=0
for i in range(1,n):
    if(n%i==0):
        sum=sum+i
if(sum==n):
    print("The Number is Perfect Number")
else:
    print("The Number is not a Perfect Number")
```

Python Programming

27. Write a python program to print the Fibonacci series.

```
no=int(input("enter a no"))
no1,no2=0,1
count=0
if no<=0:
    print("plaese enter a positive no ")
elif no==1:
    print("plaese enter a upto 1 no ")
else:
    print("fibonacci series",count)
    while(count<no):
        print(no1)
        t=no1+no2
        no1=no2
        no2=t
        count=count+1
```

Using function :-

```
def recur_fibo(n):
    if n<=1:
        return n
    else:
        return(recur_fibo(n-1)+recur_fibo(n-2))
nterms=int(input("Enter the number of terms: "))
if nterms<=0:
    print("Plese Enter a Positive Integer")
else:
    print("Fibonacci series: ")
    for i in range(nterms):
        print(recur_fibo(i))
```

Python Programming

28. Write a python program to print inverted star pattern.

```
# Input from the user
rows = int(input("Enter the number of rows: "))

# Outer loop for each row
for i in range(rows, 0, -1):
    # Inner loop for each star in the row
    for j in range(i):
        print("*", end="")
    # Print a newline after each row
    print()
```

29. Write a python program to print all the prime number within the given range.

```
n=int(input("Enter the number for Prime Number limit: "))
for i in range(2,n+1):
    c=0
    for j in range(2,i//2+1):
        if (i%j==0):
            c=c+1
    if (c==0):
        print(i, end=" ")
```

Python Programming

30. Write a python program to input principal amount, rate and time and calculate the compound interest amount.

```
p=float(input("Enter The Principal Amount: "))
t=float(input("Enter The Number of Years: "))
r=float(input("Enter The Rate of interest: "))
CI=p*(1+r/100)**t
print("Compount Interest: ",CI)
```

31. Write a python program to find cumulative sum of a list.

```
def Cumulative(l):
    new=[]
    cumsum=0
    for element in l:
        cumsum=cumsum+element
        new.append(cumsum)
    return new
lists=[10,20,30,40,50]
print("New list: ", Cumulative(lists))
```


Python Programming

32. Write a python function that takes two lists and returns *True* if they have at least one common items.

```
def common_data(list1,list2):
    result=False
    for x in list1:
        for y in list2:
            if x==y:
                result=True
                return result
print(common_data([1,2,3,4,5],[5,6,7,8,9]))
print(common_data([1,2,3,4,5],[6,7,8,9]))
```

33. Write a python program to find product of two number with repetitive addition by using recursion.

```
def recursive_multiplication(a, b):
    # Base case: if b is 0, the product is 0
    if b == 0:
        return 0
    # If b is positive, add a one less time and add a one more time
    elif b > 0:
        return a + recursive_multiplication(a, b - 1)
    # If b is negative, make it positive and negate the result
    else:
        return -recursive_multiplication(a, -b)

# Input from the user
num1 = int(input("Enter the first number: "))
num2 = int(input("Enter the second number: "))

# Calculate the product using the recursive function
product = recursive_multiplication(num1, num2)

# Output the result
print("The product of ",num1, "and", num2, "is:", product)
```

Python Programming

34. Write a python program to print Fibonacci series up to term n by using recursion.

```
def fibonacci(n):
    # Base case:
    if n <= 1:
        return n
    else:
        # Recursive case:
        return fibonacci(n-1) + fibonacci(n-2)

# Input from the user
terms = int(input("Enter the number of terms for Fibonacci series: "))

# Check if the input is valid
if terms <= 0:
    print("Please enter a positive integer.")
else:
    print("Fibonacci series up to", terms, "terms:")
    for i in range(terms):
        print(fibonacci(i), end=" ")
```

35. Write a python program to compute H.C.F.

```
def compute_hcf(x,y):
    if x>y:
        smaller=y
    else:
        smaller=x
    for i in range(1,smaller+1):
        if ((x%i==0)and(y%i==0)):
            hcf=i
    return hcf

num1=int(input("Enter a first number: "))
num2=int(input("Enter a second number: "))
print("The H.C.F. is: ",compute_hcf(num1,num2))
```

Python Programming

36. Write a program that takes in a sentence as input and displays the number of words, number of capital letters, no. of small letters and number of special symbols.

```
str1=input("Enter a Sentence: ")
word=len(str1.split())
u,l,n,s=0,0,0,0
for i in range(len(str1)):
    if str1[i]>='A' and str1[i]<='Z': u=u+1
    elif str1[i]>='a' and str1[i]<='z': l=l+1
    elif str1[i]>='0' and str1[i]<='9': n=n+1
    else: s=s+1
print("Original Strings: ",str1)
print("The Number of words in string are: "+str(word))
print('\nUpper case character: ',u)
print('Lower case character: ',l)
print('Number case: ',n)
print('Special case character: ',s)
```

37. Write a program which takes list of numbers as input and finds:
- a) The largest number in the list

```
def find_largest_number(numbers):
    if not numbers:
        return None
    max_number = numbers[0]
    for num in numbers[1:]:
        if num > max_number:
            max_number = num
    return max_number
num_list = [5, 2, 8, 1, 3, 9, 4]
largest_num = find_largest_number(num_list)
print("The largest number in the list is: ",largest_num)
```

Python Programming

b) The smallest number in the list

```
def find_smallest_number(numbers):
    if not numbers:
        return None
    min_number = numbers[0]
    for num in numbers[1:]:
        if num < min_number:
            min_number = num
    return min_number
num_list = [5, 2, 8, 1, 3, 9, 4]
smallest_num = find_smallest_number(num_list)
print("The largest number in the list is: ",smallest_num)
```

c) Product of all the items in the list

```
def calculate_product(numbers):
    if not numbers:
        return None
    product = 1
    for num in numbers:
        product *= num
    return product
num_list = [5, 2, 8, 1, 3, 9, 4]
product_of_num = calculate_product(num_list)
print("The largest number in the list is: ",product_of_num)
```

Python Programming

38. Write a Python program to combine two dictionary adding values for common keys.

```
d1 = {'a': 100, 'b': 200, 'c': 300}
```

```
d2 = {'a': 300, 'b': 200, 'd': 400}
```

Sample output: Counter({'a': 400, 'b': 400, 'd': 400, 'c': 300})

```
# Sample dictionaries
dict1 = {'a': 100, 'b': 200, 'c': 300}
dict2 = {'a': 300, 'b': 400, 'd': 500}

# Combine dictionaries and add values for common keys
combined_dict = {}

# Add values from dict1
for key, value in dict1.items():
    combined_dict[key] = combined_dict.get(key, 0) + value

# Add values from dict2
for key, value in dict2.items():
    combined_dict[key] = combined_dict.get(key, 0) + value

# Print combined dictionary
print("Combined Dictionary:")
print(combined_dict)
```

Python Programming

39. Write a program to replace 'a' with 'b', 'b' with 'c', ..., 'z' with 'a' and similarly for 'A' with 'B', 'B' with 'C', ..., 'Z' with 'A' in a file. The other characters should remain unchanged.

```
def isVowel(ch):
    if (ch != 'a' and ch != 'e' and ch != 'i' and ch != 'o' and ch != 'u'):
        return False
    return True
def replaceConsonants(s):
    for i in range(len(s)):
        if (isVowel(s[i])==False):
            if (s[i]=='z'):
                s[i]='b'
            else:
                s[i]=chr(ord(s[i])+1)
        if (isVowel(s[i])==True):
            s[i]=chr(ord(s[i])+1)
    return ''.join(s)
s="YAMA Publication"
print(replaceConsonants(list(s)))
```

40. Take two NumPy arrays having two dimensions. Concatenate the arrays on axis 1.

```
import numpy as np

# Example NumPy arrays
✓ array1 = np.array([[1, 2, 3],
                    [4, 5, 6]])
✓ array2 = np.array([[7, 8],
                    [9, 10]])

# Concatenate arrays along axis 1 (horizontally)
concatenated_array = np.concatenate((array1, array2), axis=1)

# Print the concatenated array
print("Concatenated Array along axis 1:")
print(concatenated_array)
```