**Type - 1 :- Without Parameter, Without Return type.**

**# 1. Write a program to calculate area of rectangle.**

**def rectangle\_area():**

**length = 4**

**breadth = 3**

**area = length \* breadth**

**print("1. Area of the rectangle =",area)**

**# 2. Write a program to calculate area of circle.**

**def circle\_area():**

**radius = 7**

**area = 3.14 \* radius\*\*2**

**print("2. Area of circle = ",area)**

**# 3. Write a program to find sum of following series using functions:**

**# a. 1 + 2 + 3 + 4 +..... + n**

**# b. 1! + 2! + 3! + 4! +..... + n!**

**# c. 1^1 + 2^2 + 3^3 + ...... + n^n**

**def various\_sum():**

**n = 5**

**a = 0**

**b = 0**

**c = 0**

**for i in range(1,n+1):**

**a += i**

**c += i\*\*i**

**fact = i**

**for j in range(1,i):**

**fact \*= j**

**b += fact**

**print("3.1. Sum of digits upto",n,"=",a)**

**print("3.2. Sum of all factorials upto",n,"=",b)**

**print("3.3. Sum of power series upto",n,"=",c)**

**# 4. Sum of all odd numbers between 1 to n**

**def odd\_sum():**

**n = 5**

**sum\_of\_odds = 0**

**for i in range(1,n+1):**

**if i%2 != 0:**

**sum\_of\_odds += i**

**print("4. Sum of all odd numbers upto",n,"=",sum\_of\_odds)**

**# 5. Sum of all prime numbers between 1 to n.**

**def prime\_sum():**

**n = 10**

**x = 1**

**sum\_of\_primes = 0**

**for num in range(1,n+1):**

**count\_divisible = 0**

**for j in range(1, x+1):**

**if x % j == 0:**

**count\_divisible += 1**

**if count\_divisible == 2:**

**sum\_of\_primes += x**

**else:**

**num -= 1**

**x += 1**

**print("5. Sum of all primes upto",n,"=",sum\_of\_primes)**

**# 6. Write a program to find print the following Fibonacci series using functions: 1 1 2 3 5 8...n terms.**

**def fibonacci():**

**upto = 10**

**f0 = 0**

**f1 = 1**

**fn = f1**

**count = 1**

**print("6. Fibonacci series upto",upto,"=",f0,f1,end=" ")**

**while(count != upto):**

**print(fn, end=" ")**

**count += 1**

**f0, f1 = f1, fn**

**fn = f0 + f1**

**print()**

**# 7. Write a program to find sum of digits of a number.**

**def digits\_sum():**

**n = 103429**

**sum\_of\_digits = 0**

**quo = n**

**while(quo != 0):**

**rem = quo % 10**

**quo = quo // 10**

**sum\_of\_digits += rem**

**print("7. Sum of digits of",n,"=",sum\_of\_digits)**

**# 8. Write a program find reverse of a number.**

**def num\_reverse():**

**n = 8765**

**quo = n**

**print("8. Reverse of",n,"=",end=" ")**

**while(quo != 0):**

**rem = quo % 10**

**quo = quo // 10**

**print(rem,end="")**

**print()**

**# 9. Write a program to check if entered number is a palindrome or not.**

**def palindrome\_or\_not():**

**n = 12321**

**quo = n**

**rev = 0**

**while quo > 0:**

**rem = quo % 10**

**rev = (rev \* 10) + rem**

**quo = quo // 10**

**if n == rev:**

**print("9.",n,"is a Palindrome number.")**

**else:**

**print("9.",n,"is not a Palindrome number.")**

**# 10. Write a program to check if entered year is a leap year or not.**

**def leap\_or\_not():**

**year = 1900**

**if year % 4 == 0:**

**if year % 100 == 0:**

**if year % 400 == 0:**

**flag = True**

**else:**

**flag = False**

**else:**

**flag = True**

**else:**

**flag = False**

**if flag:**

**print("10.",year,"is a leap year.")**

**else:**

**print("10.",year,"is a not leap year.")**

**# 11. WAP to check if a given number is Armstrong number or not. For each task create separate functions.**

**def armstrong\_or\_not():**

**num = 153**

**quo = num**

**len\_of\_digits = len(str(num))**

**sum\_of\_power = 0**

**while(quo!=0):**

**rem = quo % 10**

**quo = quo // 10**

**sum\_of\_power += rem \*\* len\_of\_digits**

**if sum\_of\_power == num:**

**print("11.",num,"is an Armtsrong number.")**

**else:**

**print("11.",num,"is not an Armstrong number.")**

**rectangle\_area()**

**circle\_area()**

**various\_sum()**

**odd\_sum()**

**prime\_sum()**

**fibonacci()**

**digits\_sum()**

**num\_reverse()**

**palindrome\_or\_not()**

**leap\_or\_not()**

**armstrong\_or\_not()**

**Type 2 :- Without Parameter, With Return type.**

**# 1. Write a program to calculate area of rectangle.**

**def rectangle\_area():**

**length = 4**

**breadth = 3**

**area = length \* breadth**

**return area**

**# 2. Write a program to calculate area of circle.**

**def circle\_area():**

**radius = 7**

**area = 3.14 \* radius\*\*2**

**return area**

**# 3. Write a program to find sum of following series using functions:**

**# a. 1 + 2 + 3 + 4 +..... + n**

**# b. 1! + 2! + 3! + 4! +..... + n!**

**# c. 1^1 + 2^2 + 3^3 + ...... + n^n**

**def various\_sum():**

**n = 5**

**a = 0**

**b = 0**

**c = 0**

**for i in range(1,n+1):**

**a += i**

**c += i\*\*i**

**fact = i**

**for j in range(1,i):**

**fact \*= j**

**b += fact**

**return n, a, b, c**

**# 4. Sum of all odd numbers between 1 to n**

**def odd\_sum():**

**n = 5**

**sum\_of\_odds = 0**

**for i in range(1,n+1):**

**if i%2 != 0:**

**sum\_of\_odds += i**

**return n, sum\_of\_odds**

**# 5. Sum of all prime numbers between 1 to n.**

**def prime\_sum():**

**n = 10**

**x = 1**

**sum\_of\_primes = 0**

**for num in range(1,n+1):**

**count\_divisible = 0**

**for j in range(1, x+1):**

**if x % j == 0:**

**count\_divisible += 1**

**if count\_divisible == 2:**

**sum\_of\_primes += x**

**else:**

**num -= 1**

**x += 1**

**return n, sum\_of\_primes**

**# 6. Write a program to find print the following Fibonacci series using functions: 1 1 2 3 5 8...n terms.**

**# Cannot happen in Type 2**

**# 7. Write a program to find sum of digits of a number.**

**def digits\_sum():**

**n = 103429**

**sum\_of\_digits = 0**

**quo = n**

**while(quo != 0):**

**rem = quo % 10**

**quo = quo // 10**

**sum\_of\_digits += rem**

**return n, sum\_of\_digits**

**# 8. Write a program find reverse of a number.**

**def num\_reverse():**

**n = 8765**

**quo = n**

**rev = 0**

**while(quo != 0):**

**rem = quo % 10**

**rev = rev \* 10 + rem**

**quo = quo // 10**

**return n, rev**

**# 9. Write a program to check if entered number is a palindrome or not.**

**def palindrome\_or\_not():**

**n = 12321**

**quo = n**

**rev = 0**

**while quo > 0:**

**rem = quo % 10**

**rev = (rev \* 10) + rem**

**quo = quo // 10**

**res = bool(n == rev)**

**return n, res**

**# 10. Write a program to check if entered year is a leap year or not.**

**def leap\_or\_not():**

**year = 1900**

**if year % 4 == 0:**

**if year % 100 == 0:**

**if year % 400 == 0:**

**flag = True**

**else:**

**flag = False**

**else:**

**flag = True**

**else:**

**flag = False**

**return year, flag**

**# 11. WAP to check if a given number is Armstrong number or not. For each task create separate functions.**

**def armstrong\_or\_not():**

**num = 153**

**quo = num**

**len\_of\_digits = len(str(num))**

**sum\_of\_power = 0**

**while(quo!=0):**

**rem = quo % 10**

**quo = quo // 10**

**sum\_of\_power += rem \*\* len\_of\_digits**

**res = sum\_of\_power == num**

**return num, res**

**rect\_area = rectangle\_area()**

**print("1. Area of the rectangle =",rect\_area)**

**cir\_area = circle\_area()**

**print("2. Area of the circle =",cir\_area)**

**n, a, b, c = various\_sum()**

**print("3.1. Sum of digits upto",n,"=",a)**

**print("3.2. Sum of all factorials upto",n,"=",b)**

**print("3.3. Sum of power series upto",n,"=",c)**

**n, osum = odd\_sum()**

**print("4. Sum of all odd numbers upto",n,"=",osum)**

**n, psum = prime\_sum()**

**print("5. Sum of all primes upto",n,"=",psum)**

**print("6. Fibonacci series cannot work in Type 2!")**

**n, dsum = digits\_sum()**

**print("7. Sum of digits of",n,"=",dsum)**

**n, rev = num\_reverse()**

**print("8. Reverse of",n,"=",rev)**

**n, pres = palindrome\_or\_not()**

**if pres:**

**print("9.",n,"is a Palindrome number.")**

**else:**

**print("9.",n,"is not a Palindrome number.")**

**year, lres = leap\_or\_not()**

**if lres:**

**print("10.",year,"is a leap year.")**

**else:**

**print("10.",year,"is not a leap year.")**

**num, ares = armstrong\_or\_not()**

**if ares:**

**print("11.",num,"is an Armtsrong number.")**

**else:**

**print("11.",num,"is not an Armstrong number.")**

**Type 3 :- With Parameter, Without Return type.**

**# 1. Write a program to calculate area of rectangle.**

**def rectangle\_area(length, breadth):**

**area = length \* breadth**

**print("1. Area of the rectangle =",area)**

**# 2. Write a program to calculate area of circle.**

**def circle\_area(radius):**

**area = 3.14 \* radius\*\*2**

**print("2. Area of circle = ",area)**

**# 3. Write a program to find sum of following series using functions:**

**# a. 1 + 2 + 3 + 4 +..... + n**

**# b. 1! + 2! + 3! + 4! +..... + n!**

**# c. 1^1 + 2^2 + 3^3 + ...... + n^n**

**def various\_sum(n):**

**a = 0**

**b = 0**

**c = 0**

**for i in range(1,n+1):**

**a += i**

**c += i\*\*i**

**fact = i**

**for j in range(1,i):**

**fact \*= j**

**b += fact**

**print("3.1. Sum of digits upto",n,"=",a)**

**print("3.2. Sum of all factorials upto",n,"=",b)**

**print("3.3. Sum of power series upto",n,"=",c)**

**# 4. Sum of all odd numbers between 1 to n**

**def odd\_sum(n):**

**sum\_of\_odds = 0**

**for i in range(1,n+1):**

**if i%2 != 0:**

**sum\_of\_odds += i**

**print("4. Sum of all odd numbers upto",n,"=",sum\_of\_odds)**

**# 5. Sum of all prime numbers between 1 to n.**

**def prime\_sum(n):**

**x = 1**

**sum\_of\_primes = 0**

**for num in range(1,n+1):**

**count\_divisible = 0**

**for j in range(1, x+1):**

**if x % j == 0:**

**count\_divisible += 1**

**if count\_divisible == 2:**

**sum\_of\_primes += x**

**else:**

**num -= 1**

**x += 1**

**print("5. Sum of all primes upto",n,"=",sum\_of\_primes)**

**# 6. Write a program to find print the following Fibonacci series using functions: 1 1 2 3 5 8...n terms.**

**def fibonacci(upto):**

**f0 = 0**

**f1 = 1**

**fn = f1**

**count = 1**

**print("6. Fibonacci series upto",upto,"=",f0,f1,end=" ")**

**while(count != upto):**

**print(fn, end=" ")**

**count += 1**

**f0, f1 = f1, fn**

**fn = f0 + f1**

**print()**

**# 7. Write a program to find sum of digits of a number.**

**def digits\_sum(n):**

**sum\_of\_digits = 0**

**quo = n**

**while(quo != 0):**

**rem = quo % 10**

**quo = quo // 10**

**sum\_of\_digits += rem**

**print("7. Sum of digits of",n,"=",sum\_of\_digits)**

**# 8. Write a program find reverse of a number.**

**def num\_reverse(n):**

**quo = n**

**print("8. Reverse of",n,"=",end=" ")**

**while(quo != 0):**

**rem = quo % 10**

**quo = quo // 10**

**print(rem,end="")**

**print()**

**# 9. Write a program to check if entered number is a palindrome or not.**

**def palindrome\_or\_not(n):**

**quo = n**

**rev = 0**

**while quo > 0:**

**rem = quo % 10**

**rev = (rev \* 10) + rem**

**quo = quo // 10**

**if n == rev:**

**print("9.",n,"is a Palindrome number.")**

**else:**

**print("9.",n,"is not a Palindrome number.")**

**# 10. Write a program to check if entered year is a leap year or not.**

**def leap\_or\_not(year):**

**if year % 4 == 0:**

**if year % 100 == 0:**

**if year % 400 == 0:**

**flag = True**

**else:**

**flag = False**

**else:**

**flag = True**

**else:**

**flag = False**

**if flag:**

**print("10.",year,"is a leap year.")**

**else:**

**print("10.",year,"is a not leap year.")**

**# 11. WAP to check if a given number is Armstrong number or not. For each task create separate functions.**

**def armstrong\_or\_not(num):**

**quo = num**

**len\_of\_digits = len(str(num))**

**sum\_of\_power = 0**

**while(quo!=0):**

**rem = quo % 10**

**quo = quo // 10**

**sum\_of\_power += rem \*\* len\_of\_digits**

**if sum\_of\_power == num:**

**print("11.",num,"is an Armtsrong number.")**

**else:**

**print("11.",num,"is not an Armstrong number.")**

**rectangle\_area(length = 4, breadth = 3)**

**circle\_area(radius = 7)**

**various\_sum(n = 5)**

**odd\_sum(n = 5)**

**prime\_sum(n = 10)**

**fibonacci(upto = 10)**

**digits\_sum(n = 103429)**

**num\_reverse(n = 8765)**

**palindrome\_or\_not(n = 12321)**

**leap\_or\_not(year = 1900)**

**armstrong\_or\_not(num = 153)**

**Type 4 :- With Parameter, With Return type.**

**# 1. Write a program to calculate area of rectangle.**

**def rectangle\_area(length, breadth):**

**area = length \* breadth**

**return area**

**# 2. Write a program to calculate area of circle.**

**def circle\_area(radius):**

**area = 3.14 \* radius\*\*2**

**return area**

**# 3. Write a program to find sum of following series using functions:**

**# a. 1 + 2 + 3 + 4 +..... + n**

**# b. 1! + 2! + 3! + 4! +..... + n!**

**# c. 1^1 + 2^2 + 3^3 + ...... + n^n**

**def various\_sum(n):**

**a = 0**

**b = 0**

**c = 0**

**for i in range(1,n+1):**

**a += i**

**c += i\*\*i**

**fact = i**

**for j in range(1,i):**

**fact \*= j**

**b += fact**

**return n, a, b, c**

**# 4. Sum of all odd numbers between 1 to n**

**def odd\_sum(n):**

**sum\_of\_odds = 0**

**for i in range(1,n+1):**

**if i%2 != 0:**

**sum\_of\_odds += i**

**return n, sum\_of\_odds**

**# 5. Sum of all prime numbers between 1 to n.**

**def prime\_sum(n):**

**x = 1**

**sum\_of\_primes = 0**

**for num in range(1,n+1):**

**count\_divisible = 0**

**for j in range(1, x+1):**

**if x % j == 0:**

**count\_divisible += 1**

**if count\_divisible == 2:**

**sum\_of\_primes += x**

**else:**

**num -= 1**

**x += 1**

**return n, sum\_of\_primes**

**# 6. Write a program to find print the following Fibonacci series using functions: 1 1 2 3 5 8...n terms.**

**# Cannot happen in Type 4**

**# 7. Write a program to find sum of digits of a number.**

**def digits\_sum(n):**

**sum\_of\_digits = 0**

**quo = n**

**while(quo != 0):**

**rem = quo % 10**

**quo = quo // 10**

**sum\_of\_digits += rem**

**return n, sum\_of\_digits**

**# 8. Write a program find reverse of a number.**

**def num\_reverse(n):**

**quo = n**

**rev = 0**

**while(quo != 0):**

**rem = quo % 10**

**rev = rev \* 10 + rem**

**quo = quo // 10**

**return n, rev**

**# 9. Write a program to check if entered number is a palindrome or not.**

**def palindrome\_or\_not(n):**

**quo = n**

**rev = 0**

**while quo > 0:**

**rem = quo % 10**

**rev = (rev \* 10) + rem**

**quo = quo // 10**

**res = bool(n == rev)**

**return n, res**

**# 10. Write a program to check if entered year is a leap year or not.**

**def leap\_or\_not(year):**

**if year % 4 == 0:**

**if year % 100 == 0:**

**if year % 400 == 0:**

**flag = True**

**else:**

**flag = False**

**else:**

**flag = True**

**else:**

**flag = False**

**return year, flag**

**# 11. WAP to check if a given number is Armstrong number or not. For each task create separate functions.**

**def armstrong\_or\_not(num):**

**quo = num**

**len\_of\_digits = len(str(num))**

**sum\_of\_power = 0**

**while(quo!=0):**

**rem = quo % 10**

**quo = quo // 10**

**sum\_of\_power += rem \*\* len\_of\_digits**

**res = sum\_of\_power == num**

**return num, res**

**rect\_area = rectangle\_area(4, 3)**

**print("1. Area of the rectangle =",rect\_area)**

**cir\_area = circle\_area(7)**

**print("2. Area of the circle =",cir\_area)**

**n, a, b, c = various\_sum(5)**

**print("3.1. Sum of digits upto",n,"=",a)**

**print("3.2. Sum of all factorials upto",n,"=",b)**

**print("3.3. Sum of power series upto",n,"=",c)**

**n, osum = odd\_sum(5)**

**print("4. Sum of all odd numbers upto",n,"=",osum)**

**n, psum = prime\_sum(10)**

**print("5. Sum of all primes upto",n,"=",psum)**

**print("6. Fibonacci series cannot work in Type 4!")**

**n, dsum = digits\_sum(103429)**

**print("7. Sum of digits of",n,"=",dsum)**

**n, rev = num\_reverse(8765)**

**print("8. Reverse of",n,"=",rev)**

**n, pres = palindrome\_or\_not(n = 12321)**

**if pres:**

**print("9.",n,"is a Palindrome number.")**

**else:**

**print("9.",n,"is not a Palindrome number.")**

**year, lres = leap\_or\_not(1900)**

**if lres:**

**print("10.",year,"is a leap year.")**

**else:**

**print("10.",year,"is not a leap year.")**

**num, ares = armstrong\_or\_not(153)**

**if ares:**

**print("11.",num,"is an Armtsrong number.")**

**else:**

**print("11.",num,"is not an Armstrong number.")**