For Loop :-

1. Write a Python program to print numbers from 1 to 10 using a for loop.
2. Explain the difference between a for loop and a while loop in Python.
3. Write a Python program to calculate the sum of all numbers from 1 to 100 using a for loop.
4. How do you iterate through a list using a for loop in Python?
5. Write a Python program to find the product of all elements in a list using a for loop.
6. Create a Python program that prints all even numbers from 1 to 20 using a for loop.
7. Write a Python program that calculates the factorial of a number using a for loop.
8. How can you iterate through the characters of a string using a for loop in Python?
9. Write a Python program to find the largest number in a list using a for loop.
10. Create a Python program that prints the Fibonacci sequence up to a specified limit using a for loop.
11. Write a Python program to count the number of vowels in a given string using a for loop.
12. Create a Python program that generates a multiplication table for a given number using a for loop.
13. Write a Python program to reverse a list using a for loop.
14. Write a Python program to find the common elements between two lists using a for loop.
15. Explain how to use a for loop to iterate through the keys and values of a dictionary in Python.
16. Write a Python program to find the GCD (Greatest Common Divisor) of two numbers using a for loop.
17. Create a Python program that checks if a string is a palindrome using a for loop.
18. Write a Python program to remove duplicates from a list using a for loop.
19. Create a Python program that counts the number of words in a sentence using a for loop.
20. Write a Python program to find the sum of all odd numbers from 1 to 50 using a for loop.
21. Write a Python program that checks if a given year is a leap year using a for loop.
22. Create a Python program that calculates the square root of a number using a for loop.
23. Write a Python program to find the LCM (Least Common Multiple) of two numbers using a for loop.

If else :

1. Write a Python program to check if a number is positive, negative, or zero using an if-else statement.
2. Create a Python program that checks if a given number is even or odd using an if-else statement.
3. How can you use nested if-else statements in Python, and provide an example?
4. Write a Python program to determine the largest of three numbers using if-else.
5. Write a Python program that calculates the absolute value of a number using if-else.

6. Create a Python program that checks if a given character is a vowel or consonant using if-else.
7. Write a Python program to determine if a user is eligible to vote based on their age using if-else.
8. Create a Python program that calculates the discount amount based on the purchase amount using if-else.
9. Write a Python program to check if a number is within a specified range using if-else.
10. Create a Python program that determines the grade of a student based on their score using if-else.
11. Write a Python program to check if a string is empty or not using if-else.
12. Create a Python program that identifies the type of a triangle (e.g., equilateral, isosceles, or scalene) based on input values using if-else.
13. Write a Python program to determine the day of the week based on a user-provided number using if-else.
14. Create a Python program that checks if a given year is a leap year using both if-else and a function.
15. How do you use the "assert" statement in Python to add debugging checks within if-else blocks?
16. Create a Python program that determines the eligibility of a person for a senior citizen discount based on age using if-else.
17. Write a Python program to categorize a given character as uppercase, lowercase, or neither using if-else.
18. Write a Python program to determine the roots of a quadratic equation using if-else.
19. Create a Python program that checks if a given year is a century year or not using if-else.
20. Write a Python program to determine if a given number is a perfect square using if-else.
21. Explain the purpose of the "continue" and "break" statements within if-else loops.
22. Create a Python program that calculates the BMI (Body Mass Index) of a person based on their weight and height using if-else.
23. How can you use the "filter()" function with if-else statements to filter elements from a list?
24. Write a Python program to determine if a given number is prime or not using if-else.

Map :-

1. Explain the purpose of the `map()` function in Python and provide an example of how it can be used to apply a function to each element of an iterable.
2. Write a Python program that uses the `map()` function to square each element of a list of numbers.
3. How does the `map()` function differ from a list comprehension in Python, and when would you choose one over the other?
4. Create a Python program that uses the `map()` function to convert a list of names to uppercase.
5. Write a Python program that uses the `map()` function to calculate the length of each word in a list of strings.

6. How can you use the `map()` function to apply a custom function to elements of multiple lists simultaneously in Python?
7. Create a Python program that uses `map()` to convert a list of temperatures from Celsius to Fahrenheit.
8. Write a Python program that uses the `map()` function to round each element of a list of floating-point numbers to the nearest integer.

Reduce :-

1. What is the `reduce()` function in Python, and what module should you import to use it? Provide an example of its basic usage.
2. Write a Python program that uses the `reduce()` function to find the product of all elements in a list.
3. Create a Python program that uses `reduce()` to find the maximum element in a list of numbers.
4. How can you use the `reduce()` function to concatenate a list of strings into a single string?
5. Write a Python program that calculates the factorial of a number using the `reduce()` function.
6. Create a Python program that uses `reduce()` to find the GCD (Greatest Common Divisor) of a list of numbers.
7. Write a Python program that uses the `reduce()` function to find the sum of the digits of a given number.

Filter :-

1. Explain the purpose of the `filter()` function in Python and provide an example of how it can be used to filter elements from an iterable.
2. Write a Python program that uses the `filter()` function to select even numbers from a list of integers.
3. Create a Python program that uses the `filter()` function to select names that start with a specific letter from a list of strings.
4. Write a Python program that uses the `filter()` function to select prime numbers from a list of integers.
5. How can you use the `filter()` function to remove None values from a list in Python?
6. Create a Python program that uses `filter()` to select words longer than a certain length from a list of strings.

7. Write a Python program that uses the `filter()` function to select elements greater than a specified threshold from a list of values.

Recursion:-

1. Explain the concept of recursion in Python. How does it differ from iteration?
2. Write a Python program to calculate the factorial of a number using recursion.
3. Create a recursive Python function to find the nth Fibonacci number.
4. Write a recursive Python function to calculate the sum of all elements in a list.
5. How can you prevent a recursive function from running indefinitely, causing a stack overflow error?
6. Create a recursive Python function to find the greatest common divisor (GCD) of two numbers using the Euclidean algorithm.
7. Write a recursive Python function to reverse a string.
8. Create a recursive Python function to calculate the power of a number (x^n).
9. Write a recursive Python function to find all permutations of a given string.
10. Write a recursive Python function to check if a string is a palindrome.
11. Create a recursive Python function to generate all possible combinations of a list of elements.

Basics of Functions:

1. What is a function in Python, and why is it used?
2. How do you define a function in Python? Provide an example.
3. Explain the difference between a function definition and a function call.
4. Write a Python program that defines a function to calculate the sum of two numbers and then calls the function.
5. What is a function signature, and what information does it typically include?
6. Create a Python function that takes two arguments and returns their product.

Function Parameters and Arguments:

1. Explain the concepts of formal parameters and actual arguments in Python functions.
2. Write a Python program that defines a function with default argument values.

3. How do you use keyword arguments in Python function calls? Provide an example.
4. Create a Python function that accepts a variable number of arguments and calculates their sum.
5. What is the purpose of the `*args` and `**kwargs` syntax in function parameter lists?

Return Values and Scoping:

1. Describe the role of the `return` statement in Python functions and provide examples.
2. Explain the concept of variable scope in Python, including local and global variables.
3. Write a Python program that demonstrates the use of global variables within functions.
4. Create a Python function that calculates the factorial of a number and returns it.
5. How can you access variables defined outside a function from within the function?

Lambda Functions and Higher-Order Functions:

1. What are lambda functions in Python, and when are they typically used?
2. Write a Python program that uses lambda functions to sort a list of tuples based on the second element.
3. Explain the concept of higher-order functions in Python, and provide an example.
4. Create a Python function that takes a list of numbers and a function as arguments, applying the function to each element in the list.

Built-in Functions:

1. Describe the role of built-in functions like `len()`, `max()`, and `min()` in Python.
2. Write a Python program that uses the `map()` function to apply a function to each element of a list.
3. How does the `filter()` function work in Python, and when would you use it?
4. Create a Python program that uses the `reduce()` function to find the product of all elements in a list.

Function Documentation and Best Practices:

1. Explain the purpose of docstrings in Python functions and how to write them.

2. Describe some best practices for naming functions and variables in Python, including naming conventions and guidelines.