

Functions – 20 Questions

1. What is a function in Python?

A function is a block of code that performs a specific task when called and it is reusable.

2. How do you define a function in Python?

```
def welcome():  
    print("Hello")
```

3. How do you call a function in Python?

```
welcome()
```

4. What is the purpose of the `return` statement?

It sends a result back to the caller function and ends the function.

5. Write a function to add two numbers and return the result.

```
def add(a, b):  
    return a + b
```

6. What is the difference between parameters and arguments?

- Parameters are variables in the function definition.
- Arguments are the actual values passed to the function.

7. What happens if you define a function but never call it?

Nothing happens; the code inside the function won't run.

8. Write a function to check if a number is even or odd.

```
def check_even_odd(n):  
    if n % 2 == 0:  
        print("Even")  
    else:  
        print("Odd")
```

9. What is a default argument in a function? Give an example.

An argument that has a default value.

Example :

```
def welcome(name="Guest"):  
    print("Hello", name)
```

10. What is a keyword argument?

An argument passed by explicitly specifying the parameter name.

Example : `welcome(name="Alice")`

11. Write a function with a default argument.

```
def welcome(name="Guest"):
    print("Hello,", name)
welcome("Alice")
welcome()
```

12. Write a function that returns the square of a number.

```
def square(n):
    return n * n
```

13. Can a function return multiple values? Give an example.

Yes, we can return many values:

Example :

```
def calc(a, b):
    return a + b, a - b
```

14. What is the difference between `print()` and `return` inside a function?

print() displays the output while return sends the value back to the caller.

15. What is a global variable?

Global variable is defined outside all functions and can be used anywhere.

16. What is a local variable inside a function?

Local variable exists only inside a function.

17. Write a function that accepts name and age and prints them.

```
def display(name, age):
    print("Name:", name)
    print("Age:", age)
```

18. What is the use of the `pass` statement in a function?

It is used as a placeholder when a function has no code yet.

19. Write a lambda function to multiply two numbers.

```
multiply = lambda x, y: x * y
```

20. Write a function that calculates the factorial of a number using recursion.

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

While Loop – 20 Questions

21. What is a while loop?

A loop that continues to execute as long as a condition is True.

22. Write a while loop that prints numbers from 1 to 5.

```
i = 1
while i <= 5:
```

```
    print(i)
```

```
    i += 1
```

23. What is an infinite while loop?

A loop that never ends because the condition is always True.

24. How do you stop an infinite while loop?

Using a break statement or by changing the condition inside the loop.

25. Write a while loop to print even numbers between 1 and 10.

```
i = 2
while i <= 10:
```

```
    print(i)
```

```
    i += 2
```

26. What is the difference between `for` and `while` loops?

- for is used when the number of iterations is known.
- while is used when the condition is based on logic, not a fixed count.

27. How do you use a break statement in a while loop?

```
i = 1
while i <= 10:
```

```
    if i == 5:
```

```
        break
```

```
    print(i)
```

```
    i += 1
```

28. How do you use a continue statement in a while loop?

```
i = 0
while i < 5:
```

```
    i += 1
```

```
    if i == 3:
```

```
        continue
```

```
    print(i)
```

29. Write a program to sum numbers from 1 to 100 using a while loop.

```
total = 0
i = 1
while i <= 100:
    total += i
    i += 1
print(total)
```

30. Write a while loop that asks the user for input until they type “exit”.

```
while True:
    user_input = input("Enter something (type 'exit' to quit): ")
    if user_input == "exit":
        break
```

31. What is the purpose of initializing a variable before a while loop?

To set a starting value for the loop condition.

32. Write a while loop to print the multiplication table of 5.

```
i = 1
while i <= 10:
    print("5 x", i, "=", 5 * i)
    i += 1
```

33. What happens if the condition of a while loop is always true?

The loop will run forever (infinite loop) unless stopped with break.

34. How can you create a countdown from 10 to 1 using a while loop?

```
i = 10
while i >= 1:
    print(i)
    i -= 1
```

35. Write a while loop that prints only odd numbers from 1 to 20.

```
i = 1
while i <= 20:
    print(i)
    i += 2
```

36. Can you use else with a while loop? Give an example.

Yes we can use else with a while loop.

```
i = 1
while i <= 5:
    if i % 2 == 0:
        print(i, "is Even")
    else:
        print(i, "is Odd")
    i += 1
else:
    print("Finished checking all numbers")
```

37. Write a while loop to reverse a number (e.g., input 123, output 321).

```
n = int(input("Enter number: "))
rev = 0
while n > 0:
    rev = rev * 10 + n % 10
    n = n // 10
print("Reversed:", rev)
```

38. What is the importance of updating the condition inside a while loop?

We need to update, or else it will go in infinite loop.

39. Write a while loop that calculates the factorial of a number.

```
n = int(input("Enter number: "))
f = 1
i = 1
while i <= n:
    f *= i
    i += 1
print("Factorial:", f)
```

40. Write a program using while loop to check if a number is a palindrome.

```
n = int(input("Enter number: "))
original = n
rev = 0
while n > 0:
    rev = rev * 10 + n % 10
    n //= 10
if rev == original:
    print("Palindrome")
else:
    print("Not palindrome")
```