**Python Keywords**

1. **False**: represent the Boolean value false.
2. **True**: represent the Boolean value true.
3. **None**: represent Null of No value.
4. **and**: represent the logical operator AND.
5. **or**: represent the logical operator OR.
6. **not**: represent the logical operator NOT.
7. **as**: used to create alias.
8. **assert**: used in debugging , test if the condition is true. If not is raises an AssertionError.
9. **async**: used to declare Asynchronous function.
10. **await**: pauses the execution of async function until the awaited task is completed.
11. **break**: exists the nearest loop
12. **class**: used to define a class.
13. **continue**: used to skip the current iteration of the loop.
14. **def**: to define a new function.
15. **return**: exits a function and optionally returns a value.
16. **del**: to deletes an object, variable or item from the list.
17. **if**: begins a conditional statement.
18. **elif**: used in conditional statement as ‘else if’ to test another condition if the previous is false.
19. **else**: used in conditional statement to define what happens if none of the previous conditions are true.
20. **try**: starts a block of code to test for exception.
21. **except**: used in exception handling to catch and handle an error.
22. **finally**: used in exception handling , a block that execute whether an exception occur.
23. **raise**: raise an exception manually.
24. **for**: a loop that iterates over a collection or a range.
25. **while**: starts a loop that continues as long as a specified condition is true.
26. **import**: imports a module of specific attributes/ function from a module.
27. **from**: used with import to import specific attribute or function.
28. **global**: to declare a variable as global.
29. **in**: checks if a value exists in a sequence.
30. **is**: checks if two variables points on the same object in memory.
31. **lambda**: creates anonymous functions.
32. **nonlocal**: declares an non local variable.
33. **pass**: a null statement used as placeholder in code where nothing is supposed to happen.
34. **yield**: pauses a function and returns a value temporarily.