1. Why are functions advantageous to have in your programs?

Ans:- A function is a block of reusable code that is used to perform a specific action. The advantages of using functions are:

* Reducing duplication of code
* Decomposing complex problems into simpler pieces
* Improving clarity of the code
* Reuse of code
* Information hiding

2. When does the code in a function run: when it's specified or when it's called?

Ans:-The code run when function is called. When a function is called the program control jumps to that function definition and executes the statements inside the function body. After executing the body of the function, the program control jumps back to the part of the program which called the function, and resumes execution at that point.

3. What statement creates a function?

Ans:- A function is defined by using the def keyword, followed by a name of your choosing, followed by a set of parentheses which hold any parameters the function will take (they can be empty), and ending with a colon.

def hello():

4. What is the difference between a function and a function call?.

Ans:- Using a function to do a particular task any point in program is called as function call. So the difference between the function and function call is, A function is procedure to achieve a particular result while function call is using this function to achive that task.

5. How many global scopes are there in a Python program? How many local scopes?

Ans:- There's only one global Python scope per program execution. This scope remains in existence until the program terminates and all its names are forgotten. Otherwise, the next time you were to run the program, the names would remember their values from the previous run.

When you use an unqualified name inside a function, Python searches three scopes—the local (L), then the global (G), and then the built-in (B)—and stops at the first place the name is found.

6. What happens to variables in a local scope when the function call returns?

Ans:- A local variable retains its value until the next time the function is called a local variable becomes undefined after the function call completes the local variable can be used outside the function any time after the function call completes.

7. What is the concept of a return value? Is it possible to have a return value in an expression?

Ans:- A return statement is used to end the execution of the function call and “returns” the result (value of the expression following the return keyword) to the caller. The statements after the return statements are not executed. If the return statement is without any expression, then the special value None is returned

Yes If you build a return statement without specifying a return value, then you'll be implicitly returning None . If function returns a numeric value, you can use that value in a math expression or any other kind of expression in which the value has a logical or coherent meaning

8. If a function does not have a return statement, what is the return value of a call to that function?

Ans:- If no return statement appears in a function definition, control automatically returns to the calling function after the last statement of the called function is executed. In this case, the return value of the called function is undefined

9. How do you make a function variable refer to the global variable?

Ans:- If you want to refer to a global variable in a function, you can use the global keyword to declare which variables are global.

10. What is the data type of None?

Ans:- The None keyword is used to define a null value, or no value at all. None is not the same as 0, False, or an empty string. None is a data type of its own (NoneType) and only None can be None.

11. What does the sentence import areallyourpetsnamederic do?

Ans:- import statement imports a module named areallyourpetsnamederic .

12. If you had a bacon() feature in a spam module, what would you call it after importing spam?

Ans:-

Module spam in spam.py:

def bacon():

print('hello')

import spam

spam.bacon()

#this will print hello

13. What can you do to save a programme from crashing if it encounters an error?

Ans:- If an error occurs in a program, we don’t want the program to unexpectedly crash on the user. Instead, error handling can be used to notify the user of why the error occurred and gracefully exit the process that caused the error.

14. What is the purpose of the try clause? What is the purpose of the except clause?

Ans Try and Except statement is used to handle these errors within our code in Python. The try block is used to check some code for errors i.e the code inside the try block will execute when there is no error in the program. Whereas the code inside the except block will execute whenever the program encounters some error in the preceding try block.

try:

# Some Code

except:

# Executed if error in the

# try block