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

a.) Reusability

b.) Decomposing bigger problems into smaller and simple pieces.

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

When it’s called

3. What statement creates a function?

def function(a, b)

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

Def sum1(a,b): is a function : - It is a definition of a function which will return some value to a calling function.

Sum1(4,5) is a function call which will call that particular function

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

Each module is a global scope—a namespace where variables created (assigned) at the top level of a module file live.

Each call to a function is a new local scope Every time you call a function, you create a new local scope—a namespace where names created inside the function usually live.

Assigned names are local, unless declared global

By default, all the names assigned inside a function definition are put in the local scope (the namespace associated with the function call). If you need to assign a name that lives at the top-level of the module enclosing the function, you can do so by declaring it in a global statement inside the function.

All other names are global or built-in

Names not assigned a value in the function definition are assumed to be globals (in the enclosing module’s namespace) or built-in (in the predefined names module Python provides).

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

Python will check if there is any variable already exist in Global scope and if not then python will return NameError variable is not defined.

That means local variables scope is ended and can’t use anymore and it is destroyed.

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

Return value will return a value to called function variable for ex. def Add(a,b) return a+b Result = Add(1+2)

Yes, possible to return value in an expression also. The **return** statement is followed by an expression which is evaluated. Its result is returned to the caller as the “fruit” of calling this function. Because the return statement can contain any Python expression

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

Return value of a call to that function will give you answer as None

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

The global Keyword

10. What is the data type of None?

NoneType

11. What does the sentence import areallyourpetsnamederic do?

Import areallyourpetsnamederic will check python library and if there is any library with this name then Module is found else system will throw ModuleNameNotFoundError .

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

From spam import bacon bacon() or we can call it via spam.bacon()

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

Ctrl + c

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

We need to put suspicious code inside try block and except comes when try block is failed.

Here we can catch many define exceptions , so that program does not give runtime error.