

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

ANSWER: Functions reduce the need for duplication of the code. This makes program shorter, easier to read and update. Also the memory and the time used for running the code will be less when compared to the programs without functions.

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

ANSWER: The code in the function executes when the function is called.

3. What statement creates a function?

ANSWER: The def statement defines or creates the function.

```
In [4]: def myfunc(a,b):  
        return a+b  
  
        myfunc(2,3)  
  
Out[4]: 5
```

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

ANSWER: A function consist the def statement and the code in its def clause. A function call means invoking or calling that function which evaluates the code in def clause of the function and provides the desired output or result.

```
In [4]: def myfunc(a,b):# function  
        return a+b #code in the def clause of the function  
  
        myfunc(2,3) #calling a function  
  
Out[4]: 5
```

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

ANSWER: There's only one global Python scope per program execution. This scope remains in existence until the program terminates, and a local scope is created whenever a function is called.

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

ANSWER: Whenever the function is called the local scope is destroyed.

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

ANSWER: A return value is the value that a function call evaluates to. Like any value, a return value can be used as part of an expression.

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

ANSWER: 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?

ANSWER: To make the function variable refer to the global variable global keywords can be used to declare which variables are global.

10. What is the data type of None?

ANSWER: None is a data type of its own i.e None type is used to define a null value.

```
In [5]: print(type(None))  
        <class 'NoneType'>
```

11. What does the sentence import areallyourpetsnamederic do?

ANSWER: Imports the module named areallyourpetsnamederic, however this module name does not exist in Python.

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

ANSWER: This function can be called with spam. bacon().

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

ANSWER: We can use try except clause to use the function from crashing. If the code performs well its gets executed or else it prints the exception.

```
try:  
    a=10  
    b=0  
    c=a/b  
    print(c)  
except ZeroDivisionError:  
    print("Number cannot be divided by zero")
```

```
Number cannot be divided by zero
```

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

ANSWER: try and except clause is used to encounter the errors in the program. try clause is used to write the entire code. If the code is containing zero errors the program executes the code in the try clause. Except clause is used to avoid the program written in try clause from crashing and helps you to understand which block of code contains what type of error i.e if code in try clause contains some error it will execute the code written in except clause.

```
try:
    a=10
    b=0
    c=a/b
    print(c)
except ZeroDivisionError:
    print("Number cannot be divided by zero")
```

Number cannot be divided by zero