|  |  |
| --- | --- |
|  | Chapter 3 Practice Questions – Functions |
| **Q1** | **Why are functions advantageous to have in your programs?** |
| A | They help organise code into ‘black boxes’ that perform certain tasks by taking in parameters and giving out outputs which can be used in other parts of a program |
| **Q2** | **When does the code in a function execute: when the function is defined or when the function is called?** |
| A | When the function is called |
| **Q3** | **What statement creates a function?** |
| A | Def |
| **Q4** | **What is the difference between a function and a function call?** |
| A | A function call involves passing in parameters for the function to process and give an output whereas a function is the code of the function i.e. when it is defined |
| **Q5** | **How many global scopes are there in a Python program? How many local scopes?** |
| A | 1 global scope is created when the program begins |
| **Q6** | **What happens to variables in a local scope when the function call returns?** |
| A | Variables assigned within a function exist in the local scope and when the function returns, the local scope is destroyed along with any variables |
| **Q7** | **What is a return value? Can a return value be part of an expression?** |
| A | A return value is the value that a function call evaluates to. An expression consists of values and operators and can always evaluate down to a single value therefore return values are normally not part of an expression |
| **Q8** | **If a function does not have a return statement, what is the return value of a call to that function?** |
| A | Python adds return value = None to the end of any function definition with no return statement |
| **Q9** | **How can you force a variable in a function to refer to the global variable?** |
| A | If you need to modify a global variable from within a function, use the global statement |
| **Q10** | **What is the data type of None?** |
| A | None is the only value of the NonType data type |
| **Q11** | **What does the** *import areallyourpetsnamederic* **statement do?** |
| A | The import statement itself imports a module which contains many functions |
| **Q12** | **If you had a function named bacon() in a module named spam, how would you call it after importing spam?** |
| A | You call it by typing the module name followed by a period and then the function name as below: spam.bacon(*arguments*) |
| **Q13** | **How can you prevent a program from crashing when it gets an error?** |
| A | The code that could potentially have an error is put in a try clause. When code in a try clause causes an error, the program execution immediately moves to the code in the except clause |
| **Q14** | **What goes in the try clause? What goes in the except clause?** |
| A | The code that could potentially cause an error goes in the try clause while the code in the except clause is what is executed right after an error is encountered in the try clause |