```
In [ ]: 1)  Why are functions advantageous to have in your programs?

        Functions reduce the need for duplicate code.
        This makes programs shorter, easier to read, and easier to update.
```

```
In [ ]: 2)  When does the code in a function run:
            when its specified or when its called ?

            The code in a function executes when the function is called,
            not when the function is defined.
```

```
In [ ]: 3)  What statement creates a function?

        The def statement defines (that is, creates) a function.
```

```
In [ ]: 4)  What is the difference between a function and a function call?

        A function consists of the def statement and the code in its def clause.

        A function call is what moves the program execution into the function,
        and the function call evaluates to the function's return value.
```

```
In [ ]: 5)How many global scopes are there in a Python program?
        How many local scopes?

        There is one global scope, and a local scope is created
        whenever a function is called.
```

In [ ]:
```
6)What happens to variables in a local scope when the
function call returns?

When a function returns, the local scope is destroyed,
and all the variables in it are forgotten.
```

In [ ]:
```
7) What is the concept of a  return value? is it possible to have a return value in an expression?

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.
```

In [ ]:
```
8) If a function does not have a return statement,
what is the return value of a call to that function?

If there is no return statement for a function,
its return value is None.
```

In [ ]:
```
9) How do you make a function variable refer to the
 global variable?

A global statement will force a variable in a function
to refer to the global variable.
```

In [ ]:
```
10) What is the data type of None?

The data type of None is NoneType.
```

```
In [ ]: 11) What does the import areallyourpetsnamederic do?

        That import statement imports a module named areallyourpetsnamederic.
        (This isn't a real Python module, by the way.)
```

```
In [ ]: 12) If you had a bacon() feature in a spam module ,
        what would you call it after importing spam?

        This function can be called with spam.bacon().
```

```
In [ ]: 13) How can you do to save a programme from crashing if it encounters an error?

        Place the line of code that might cause an error in a try clause.
```

```
In [ ]: 14) What is the purpose of the try clause?
        What is the purpose of the except clause?


        The code that could potentially cause an error goes in the try clause.

        The code that executes if an error happens goes in the except clause.
```