1. What exactly is []?

Functions allow code reusability, improve modularity, reduce redundancy, and make programs easier to debug and maintain.

Ex:-

```
def greet():
    print('Hello!')


# Call the function
greet()
```

2. In a list of values stored in a variable called spam, how would you assign the value 'hello' as the third value? (Assume [2, 4, 6, 8, 10] are in spam.)

The code in a function runs only when it is explicitly called.

Ex:-

```
def add(a, b):
    return a + b


# Call the function
result = add(3, 5)  # Function runs here.
```

3. What is the value of spam[int(int('3' * 2) / 11)]? (Assume spam = ['a', 'b', 'c', 'd'])

The 'def' statement creates a function.

Ex:-

```
def my_function():
    print('This is a function.')
```

4. What is the value of spam[-1]?

A function is a block of code defined using 'def', while a function call executes it.

Ex:-

```
def square(n):
    return n * n


result = square(4)  # Function is called here.
```

5. What is the value of spam[:2]?

There is only one global scope in a Python program, but local scopes are created whenever a function is called.

Ex:-

```
x = 10  # Global scope
def local_example():
    y = 5  # Local scope
```

6. What is the value of bacon.index('cat')? (Assume bacon = [3.14, 'cat', 11, 'cat', True])

Variables in a local scope are destroyed when the function call completes.

Ex:-

```
def temporary_scope():
    temp = 42  # Exists only during this function call
```

7. How does bacon.append(99) change the look of the list value in bacon?

A return value is the output a function sends back after execution. Yes, return values can be used in expressions.

Ex:-

```
def multiply(a, b):
    return a * b

result = multiply(3, 4) + 2
```

8. How does bacon.remove('cat') change the look of the list in bacon?

If a function does not have a return statement, it implicitly returns None.

Ex:-

```
def no_return():
    print('No return value')

result = no_return()  # result is None
```

9. What are the list concatenation and list replication operators?

Use the 'global' keyword to refer to a global variable inside a function.

Ex:-

```
x = 5
def modify_global():
    global x
    x = 10
```

10. What is the difference between the list methods append() and insert()?

The data type of None is NoneType.

Ex:-

```
result = None
print(type(result))  # Outputs: <class 'NoneType'>
```

11. What are the two methods for removing items from a list?

The statement imports a module named 'areallyourpetsnamederic', if it exists.

Ex:-

```
import areallyourpetsnamederic
```

12. Describe how list values and string values are identical.

Call the bacon() function from the spam module as follows:

Ex:-

```
import spam
spam.bacon()
```

13. What's the difference between tuples and lists?

Use try and except blocks to handle errors and prevent crashes.

Ex:-

```
try:
    result = 10 / 0
except ZeroDivisionError:
    print('Cannot divide by zero.')
```

14. How do you type a tuple value that only contains the integer 42?

The try clause contains the code that might raise an exception, and the except clause handles it.

Ex:-

```
try:
    print(10 / 0)
except ZeroDivisionError:
    print('Division by zero error.')
```