```
# Q: What are the primitive data types in Python?
A: The primitive data types in Python are integers, floats, booleans, strings, and
None.

Q: What is the difference between "==" and "is" operators in Python?
A: The "==" operator checks for value equality, while the "is" operator checks for
object identity.

Q: Explain the difference between 'and' and 'or' operators in Python.
A: The 'and' operator returns True if both operands are true, while the 'or' operator
returns True if either operand is true.

Q: How do you check if a variable is of a specific data type in Python?
A: You can use the isinstance() function to check if a variable is of a specific data
type. For example: isinstance(var, int).

Q: What is the ternary operator in Python?
A: The ternary operator, also known as the conditional expression, is a shorter way to
write if-else statements in a single line.

Q: What is the purpose of the "pass" statement in Python?
A: The "pass" statement is a placeholder that does nothing. It is used as a syntactic
placeholder where a statement is required.

Q: Explain the difference between "break" and "continue" statements in Python.
A: The "break" statement is used to exit the loop entirely, while the "continue"
statement skips the current iteration and moves to the next.

Q: How can you iterate over a list and its indices simultaneously in Python?
A: You can use the enumerate() function to iterate over a list along with its indices.
For example: for index, value in enumerate(my_list).

Q: What is the purpose of the range() function in Python?
A: The range() function is used to generate a sequence of numbers. It is commonly used
in for loops to iterate a specific number of times.

Q: How do you define a function in Python?
A: You can define a function using the "def" keyword, followed by the function name,
parentheses, and a colon. For example: def my_function().

Q: What is the difference between arguments and parameters in Python?
A: Parameters are variables defined in the function signature, while arguments are the
actual values passed to the function when it is called.

Q: Can a function return multiple values in Python?
A: Yes, a function can return multiple values in Python by separating them with commas.
They are returned as a tuple and can be unpacked.

Q: What is a lambda function in Python?
A: A lambda function, also known as an anonymous function, is a small function that
doesn't have a name and is defined using the lambda keyword.

Q: How can you pass a function as an argument to another function in Python?
A: In Python, you can pass a function as an argument by using the function name without
parentheses. For example: my_function(another_function).

Q: What is the difference between local and global variables in Python?
A: Local variables are defined within a function and have a local scope, while global
variables are defined outside functions and have a global scope.
```

Q: How can you modify a global variable inside a function in Python?
A: To modify a global variable inside a function, you need to use the "global" keyword before the variable name. For example: global x.

Q: What is the purpose of a docstring in Python?
A: A docstring is a string literal that appears as the first line in a function, module, or class. It is used to document and describe its purpose.

Q: What is recursion in Python? Explain with an example.
A: Recursion is a programming technique where a function calls itself to solve a problem. An example is the factorial function: def factorial(n): return n * factorial(n-1).

Q: How can you handle exceptions in Python?
A: Exceptions in Python can be handled using the try-except block. Code that might raise an exception is placed within the try block, and the except block catches and handles the exception.

Q: What is the purpose of the "return" statement in a function?
A: The "return" statement is used to exit a function and optionally return a value. It can also be used to terminate the execution of a loop or a program

In [ ]: