

Python Function Arguments Study Note

Python Function Arguments

This note summarizes the key concepts about function arguments in Python from the video.

Types of Arguments

- **Required Arguments:** Must be passed when the function is called. Failing to do so results in an error.
- **Default Arguments:** Have a pre-assigned default value that is used if a new value is not provided.
- **Keyword Arguments:** Passed using the parameter name, so the order doesn't matter.
- **Variable-Length Arguments:** Allow a function to accept a variable number of arguments.
 - `*args`: For a variable number of non-keyword arguments, which are passed as a tuple.
 - `**kwargs`: For a variable number of keyword arguments, which are passed as a dictionary.

Code Examples

Simple Average Function

```
...  
def average(a, b):  
    print("The average is", (a + b) / 2)  
  
average(3, 7)  
...
```

Average Function with Variable-Length Arguments

```
...  
def average(*numbers):  
    sum = 0  
    for i in numbers:  
        sum = sum + i  
    print("The average is", sum / len(numbers))
```

```
average(5, 6, 7, 1)
'''
```

Return Statement

The ``return`` statement is used to send a value back to the calling function. If a function does not have a ``return`` statement, it returns ``None`` by default.