

Python Function Signature Survival Guide

This guide summarizes how Python functions can use positional arguments, keyword-only arguments, and argument unpacking. It includes syntax explanations and usage tips.

1. Basic Function Parameters

```
def foo(a, b, c):  
    # a, b, and c are positional or keyword  
    pass
```

2. Default Arguments

```
def foo(a, b=2):  
    # b is optional with a default value  
    pass
```

3. *args - Variable Positional Arguments

```
def foo(*args):  
    # args is a tuple of all extra positional arguments  
    pass
```

4. **kwargs - Variable Keyword Arguments

```
def foo(**kwargs):  
    # kwargs is a dict of all extra keyword arguments  
    pass
```

5. *- Keyword-Only Arguments

```
def foo(a, *, b, c=3):  
    # b and c must be passed as keywords: foo(1, b=2)  
    pass
```

6. Combined Usage

```
def foo(a, *args, b, **kwargs):  
    # a is required positional  
    # args collects extra positionals  
    # b is required keyword-only  
    # kwargs collects all extra keywords  
    pass
```