Positional Arguments:

These are the most common type of arguments.
The values are assigned to parameters based on their order.

```
def example_function(a, b, c):
    # a, b, and c are positional arguments
    print(a, b, c)

example_function(1, 2, 3)  # Output: 1 2 3
```

Keyword Arguments:

Values are assigned to parameters by explicitly specifying the parameter names.

```
def example_function(a, b, c):
    print(a, b, c)

example_function(a=1, b=2, c=3)  # Output: 1 2 3
```

Default Arguments:

Parameters can have default values, which are used if the caller doesn't provide a value.

```
def example_function(a, b=2, c=3):
    print(a, b, c)

example_function(1)         # Output: 1 2 3
example_function(1, 4)      # Output: 1 4 3
example_function(1, 4, 5)   # Output: 1 4 5
```

Variable-Length Positional Arguments (Arbitrary Arguments):

Allows a function to accept any number of positional arguments.

```
def example_function(*args):
    for arg in args:
        print(arg)

example_function(1, 2, 3)  # Output: 1 2 3
```

Variable-Length Keyword Arguments (Arbitrary Keyword Arguments):

Allows a function to accept any number of keyword arguments.

```
def example_function(**kwargs):
    for key, value in kwargs.items():
        print(key, value)
```

```
example_function(a=1, b=2, c=3)  # Output: a 1, b 2, c 3
```

Combining Argument Types:

Functions can use a combination of positional, keyword, default, and variable-length arguments.

```
def example_function(a, b=2, *args, c=3, **kwargs):
    print(a, b, args, c, kwargs)

example_function(1, 4, 5, 6, c=7, d=8, e=9)
# Output: 1 4 (5, 6) 7 {'d': 8, 'e': 9}
```