



CSCI 141

Scott Wehrwein

Keyword Arguments

Goals

- Know the purpose and syntax for `keyword arguments`
- Know how to use the `print` function's `sep`, and `end` keyword arguments.

Reminder: Function Call Syntax

```
print("Bellingham", "WA", "USA")
```

The diagram illustrates the syntax of a function call. It shows the word 'print' in green, followed by an open parenthesis '(', three arguments in quotes separated by commas, and a closing parenthesis ')'. Arrows point from the labels to their corresponding parts: 'Function name' points to 'print', 'Open paren' points to the first parenthesis, 'Comma-separated list of arguments' points to the three quoted strings, and 'Close paren' points to the final parenthesis.

Function name Comma-separated list of arguments

Open paren Close paren

Keyword Arguments

A mechanism for **optionally** passing information to a function.

```
print("Bellingham", "WA", "USA", sep=" ")
```

The `print` function can take a `sep` **keyword argument** to change what is printed *between* arguments

If you leave it out, it's equivalent to passing a single space:

```
print("Bellingham", "WA", "USA") # same as:  
print("Bellingham", "WA", "USA", sep=" ")
```

Keyword Arguments

A mechanism for **optionally** passing information to a function.

```
print("Bellingham", "WA", "USA", end="!")
```

The `print` function can take an `end` keyword argument to specifies what to print after the last argument.

The newline character

In a string, the special character sequence `\n` indicates a **newline**, or line break.

Example:

```
>>> print("line one\nline two")
line one
line two
>>>
```

Demo: Print's Keyword Args

Demo: Print's Keyword Args

- Print with sep
- Print with end=""
 - End defaults to newline
- Print with end="!", end="!\n"
- Print with sep and end