System: Command Line Arguments

CPSC 217: Introduction to Computer Science for Multidisciplinary Studies I
Fall 2020

Jonathan Hudson, Ph.D
Instructor
Department of Computer Science
University of Calgary

Tuesday, September 8, 2020

UNIVERSITY OF CALGARY

Command Line Arguments

Command-line Arguments

- You can pass 0, 1 or more command-line arguments to a Python program

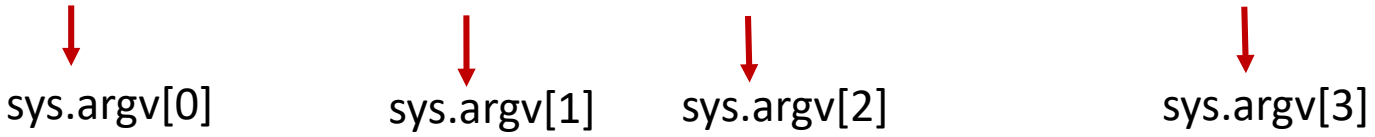
```
>python programName.py

>python programName.py arg1

>python programName.py arg1 arg2

>python programName.py arg1 arg2 ...
```

Command-line Arguments

- The arguments must be separated by space.
- If one of the arguments contain a space (a string containing spaces), then you must surround it with double quotes.
- The program accesses these arguments through ***argv*** from the ***sys*** package
- **argv** (**arg**ument **v**alues) is a list of the passed arguments.

Command-line Arguments

```
>python myProgram.py 1 string "string string"
```



Command-line Arguments

```
import sys
msg = 'Passed %d arguments. These are: %s' % (len(sys.argv), sys.argv)
print(msg)
```

Command-line Arguments

```
import sys
msg = 'Passed %d arguments. These are: %s' % (len(sys.argv), sys.argv)
print(msg)
```

```
>python myProgram.py 1 string "string string"
```

First argument is always the module name

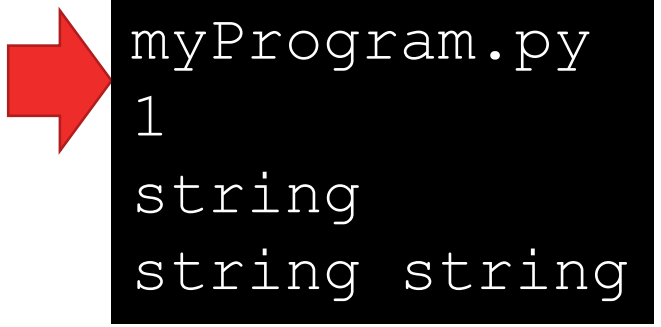
```
Passed 4 arguments.
    These are: ['myProgram.py', '1', 'string', 'string string']
```

Command-line Arguments

- You can access individual arguments using indices or by iterating through *argv*.

```
>python myProgram.py 1 string "string string“
```

```
import sys
print(sys.argv[0])#first argument

for arg in sys.argv:#print all arguments
    print(arg)
```



Errors

Command-line Arguments

- When accessing arguments using index, you must be careful not to use an index that is outside the range of the list.
  - Otherwise, an error will occur.

```
>python myProgram.py 1 string "string string"
```

```
from sys import argv
i = 0
while i <= len(argv):
    print(argv[i])
    i+=1
```

```
myProgram.py
1
string
string string
Traceback (most recent call last):
  File "myProgram.py", line 4, in <module>
    print(argv[counter])
IndexError: list index out of range
```

Command-line Arguments

- To ensure that an argument is within the range of the *argv* list, use the *len()* function:

```
>python myProgram.py "New Argument"
```

```
import sys
if len(sys.argv) != 2:
      print('Invalid number of arguments')
else:
      value1 = sys.argv[1]
      print("Received", value1)
```

One argument is expected (in addition to the module name)

Design

Good Program Design

- Check if argument count is correct
  - If not then end program with error message
- Check if argument data is correct
  - If not then end program with error message
- Check if files can be opened?
  - Next topic

Onward to …
files.

Jonathan Hudson
jwhudson@ucalgary.ca
https://pages.cpsc.ucalgary.ca/~hudsonj/

UNIVERSITY OF CALGARY