

Test Program

Jason R. Fruit

February 13, 2010

Contents

1	Introduction	1
2	Implementation	1
2.1	Output files	1
2.2	Retrieving the OS description	2
2.3	Building the message	2
2.4	Printing the message	3

1 Introduction

This test program prints the word “hello”, followed by the name of the operating system as understood by Python. It is implemented in Python and uses the `os` module. It builds the message string in two different ways, and writes separate versions of the program to two different files.

2 Implementation

2.1 Output files

This document contains the makings of two files; the first, `test.py`, uses simple string concatenation to build its output message:

Code example test.py (1)

```
▷ Code Example Import the os module (3)
▷ Code Example Get the OS description (4)
▷ Code Example Construct the message with Concatenation (5)
▷ Code Example Print the message (7)
```

The second uses string substitution:

Code example test2.py (2)

```
▷ Code Example Import the os module (3)
▷ Code Example Get the OS description (4)
▷ Code Example Construct the message with Substitution (6)
▷ Code Example Print the message (7)
```

2.2 Retrieving the OS description

First we must import the os module so we can learn about the OS:

Code example Import the os module (3)

```
import os
```

Referred to in:

Code example test.py (1) (Sect. 2.1, p. 1) ;

Code example test2.py (2) (Sect. 2.1, p. 1)

That having been done, we can retrieve Python's name for the OS type:

Code example Get the OS description (4)

```
os_name = os.name
```

Referred to in:

Code example test.py (1) (Sect. 2.1, p. 1) ;

Code example test2.py (2) (Sect. 2.1, p. 1)

2.3 Building the message

Now, we're ready for the meat of the application: concatenating two strings:

Code example Construct the message with Concatenation (5)

```
msg = "Hello, " + os_name + "!"
```

Referred to in:

Code example test.py (1) (Sect. 2.1, p. 1)

But wait! Is there a better way? Using string substitution might be better:

Code example Construct the message with Substitution (6)

```
msg = "Hello, %s!" % os_name
```

Referred to in:

Code example test2.py (2) (Sect. 2.1, p. 1)

We'll use the first of these methods in `test.py`, and the other in `test2.py`.

2.4 Printing the message

Finally, we print the message out for the user to see. Hopefully, a cheery greeting will make them happy to know what operating system they have:

Code example Print the message (7)

```
print msg
```

Referred to in:

Code example test.py (1) (Sect. 2.1, p. 1) ;

Code example test2.py (2) (Sect. 2.1, p. 1)