Introduction to `Bash`

Mauricio Sevilla

email= j.sevillam@uniandes.edu.co

01.21.19

Introduction

`Shell` is one of the most powerful tools on scientific programming,

- Natural for programming.
- Way to interact with a server.
- Very useful when managing data.

Interfaces

There are two different kinds of user interfaces

- GUI (Graphical User Interface)
    - Aqua (Mac)
    - Gnome (Ubuntu/Linux)
    - Unity (Ubuntu/Linux)
- CLI (Command Line Interface)
    - Bash
    - DOS
    - Powershell

We are very used to the GUI.

At some cases CLI is way better.

Graphical User Interface

It is done for *using*,

- Using already *existing* software.
- Just by clicking on *existing* controls you get to use *existing* funtionalities
- Pretty useful for some computational operations.

Command Line Interface

It is done for creating.

Bash

`Bash` is the name of the `shell` used on Ubuntu and Mac.

It will allow us to acces to the computer and its services and programs.

Special Characters

There are some characters `saved` for `Bash`

Character	Description	Character	Description
\	Escape Character	/	Directory Separator
.	Current Directory	..	Parent Directory
~	Home Directory	*	Bonus Character
?	Bonus Character (1)	[ ]	Range of Values
|	Pipeline	>	Redirect Output
>>	Redirect Output	<	Redirect input
;	Run in Background	&&	Command Separator
&	Command Separator	#	Comment

Files and Navigation

There are some commands we need to learn in order to move along the bash

Command	Description
pwd	Print Working Directory
ls	List
cd	Change Directory
mkdir	Make Directory
rmdir	Remove Directory
rm	Remove
more	To read files
cat	To read files

To open a terminal,

- *On Ubuntu*,

  `ctr+alt+t`

- *On Mac*,

  Others/terminal

Examples

`echo` prints the following string

In [1]:
```
echo test
```

```
test
```

`cd` for Change-Directory, this command allows you to move among folders

In [2]:
```
cd
```

`ls` for list, this command shows the files on the working folder

In [3]:
```
ls
```

```
Applications    Downloads       Movies          Public
Desktop         Dropbox         Music           untitled folder
Documents       Library         Pictures
```

`pwd` for print-working-directory

In [4]:
```
pwd
```

```
/Users/jmsevillam
```

In [5]:
```
man pwd
```

```
PWD(1)                    BSD General Commands Manual                   PWD(1)

NAME
     pwd -- return working directory name

SYNOPSIS
     pwd [-L | -P]

DESCRIPTION
     The pwd utility writes the absolute pathname of the current working
     directory to the standard output.

     Some shells may provide a builtin pwd command which is similar or identi-
     cal to this utility.  Consult the builtin(1) manual page.

     The options are as follows:

     -L      Display the logical current working directory.

     -P      Display the physical current working directory (all symbolic
             links resolved).

     If no options are specified, the -L option is assumed.

ENVIRONMENT
     Environment variables used by pwd:

     PWD  Logical current working directory.

EXIT STATUS
     The pwd utility exits 0 on success, and >0 if an error occurs.

SEE ALSO
     builtin(1), cd(1), csh(1), sh(1), getcwd(3)
```

STANDARDS

The pwd utility conforms to IEEE Std 1003.1-2001 (``POSIX.1'').

BUGS

In csh(1) the command dirs is always faster because it is built into that shell. However, it can give a different answer in the rare case that the current directory or a containing directory was moved after the shell descended into it.

The -L option does not work unless the PWD environment variable is exported by the shell.

In [6]:
```
ls
```

```
Applications    Downloads       Movies          Public
Desktop         Dropbox         Music           untitled folder
Documents       Library         Pictures
```

In [7]:
```
mkdir test_new_folder
```

In [8]:
```
ls
```

```
Applications    Downloads       Movies          Public
Desktop         Dropbox         Music           test_new_folder
Documents       Library         Pictures        untitled folder
```

In [9]:
```
rmdir test_new_folder
```

In [10]:
```
ls
```

```
Applications    Downloads       Movies          Public
Desktop         Dropbox         Music           untitled folder
Documents       Library         Pictures
```

Working with files

We can now start using some of the previous concepts and instructions

In [11]:
```
echo text > file.txt
```

In [12]:
```
cat file.txt
```

```
text
```

Exercise

Try with `more` and `less`

Can you see a difference?