

## Scripting with Bash



### Workshop module IV

## Coping with Bash



### Workshop module IV

# What are we looking at

1. Warmup – She-Bang and parameter expansion
2. Scraping Researcher – branches and loops
3. Managing Dataset – more loops and ffmpeg

# Some words on origin

- FSF's answer to the Unix shell
- Started in 1987, version 3 published in 2004
- Brian Fox paid as main developer



# Some words on origin

- FSF's answer to the Unix shell
- Started in 1987, version 3 published in 2004
- Brian Fox paid as main developer
- From 1990 Chet Ramey
- Current version 5.1 released in April



# Why learn Bash?

- Learning Bash helps you understand Linux/Unix
- Shell scripts, especially in Bash, are frequently encountered on Unix systems

# Why learn Bash?

- Learning Bash helps you understand Linux/Unix
- Shell scripts, especially in Bash, are frequently encountered on Unix systems
- Shell scripts evolve naturally from your workflow



# Why learn Bash?

- Learning Bash helps you understand Linux/Unix
- Shell scripts, especially in Bash, are frequently encountered on Unix systems
- Shell scripts evolve naturally from your workflow
- Shell scripts are often used to glue more “advanced” programming languages





# Why is Bash such a weird land?

- Feels like string substitutions mixed with functional programming



# Why is Bash so weird?

- Feels like string substitutions mixed with functional programming
- Evolved over many years → some cruft (three ways to branch, two ways for functions)

# Why is Bash so weird?

- Feels like string substitutions mixed with functional programming
- Evolved over many years → some cruft
- Loves special symbols



# What to expect

- Provide a fast-forward path for your knowledge of terminals
- Provide additional practice to those commands learned in the first session
- Understand basic concept of Bash
- Bash is very weird, expect to use a subset

# Warmup session

- The She-Bang
- Read command
- The structure of a program
- Parameter expansion
- Protecting variables

# Warmup session

- The She-Bang
  - Read command
  - The structure of a program
  - Parameter expansion
  - Protecting variables
- 
- More usecases for variable expansion
  - ``set -noe pipeofail``
  - How to compose channels from files
  - How to do logging

# Researcher scraper

- Conditionals
- Single, many statements
- Working with arrays
- Conditional structures
- Automation with crontab

# Researcher scraper

- Conditionals
- Single, many statements
- Working with arrays
- Conditional structures
- Automation with crontab
  
- add a routine to send you an email of the diff, e.g. ``git diff –stat``
- Create second scraper for Arxiv or search pattern for Google Scholar



# Managing a Dataset

- More loops
- Functions
- Inline branches
- Comparing strings with ``comm``

# Managing a Dataset

- More loops
- Functions
- Inline branches
- Comparing strings with ``comm``
  
- Actually check file, not just ending (use ``soxi``)
- Slice audio into chunks (for block based processing)
- Etc.

# Back again

- Use your browser in modal mode (avoid scrolling)



# Back again

- Use your browser in modal mode (avoid scrolling)
- Tiling window manager, e.g. i3



**i3**

improved tiling wm

# Back again

- Use your browser in modal mode (avoid scrolling)
- Tiling window manager, e.g. i3
- Markup which is simple, portable, accessible

```
1 # Markdown Preview
2
3 Type on the __left__, see it
4 * _rendered_ on the __right__.
5
6 This is a
7 * [link](https://github.com).
8
9 - and this is
10 - a list
11
12 :tada: :fireworks:
```



**Markdown Preview**

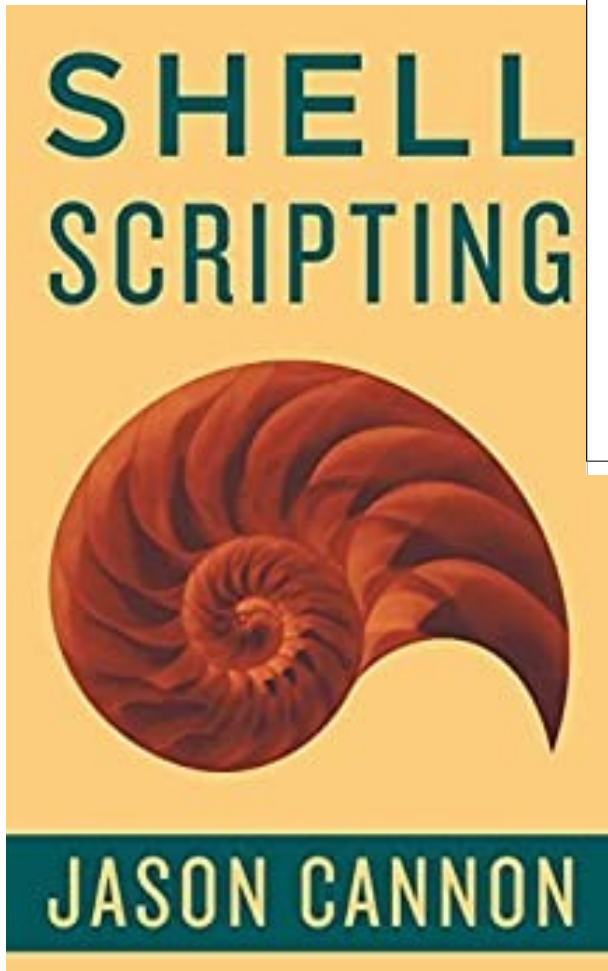
Type on the **left**, see it *rendered* on the **right**.

This is a [link](https://github.com).

- and this is
- a list

🎆 🎇

# Open Stage



## Bash Reference Manual

### Table of Contents

- 1 Introduction**
  - 1.1 What is Bash?
  - 1.2 What is a shell?
- 2 Definitions**
- 3 Basic Shell Features**
  - 3.1 Shell Syntax
    - 3.1.1 Shell Operation
    - 3.1.2 Quoting
      - 3.1.2.1 Escape Character
      - 3.1.2.2 Single Quotes
      - 3.1.2.3 Double Quotes
      - 3.1.2.4 ANSI-C Quoting
      - 3.1.2.5 Locale-Specific Translation
    - 3.1.3 Comments

