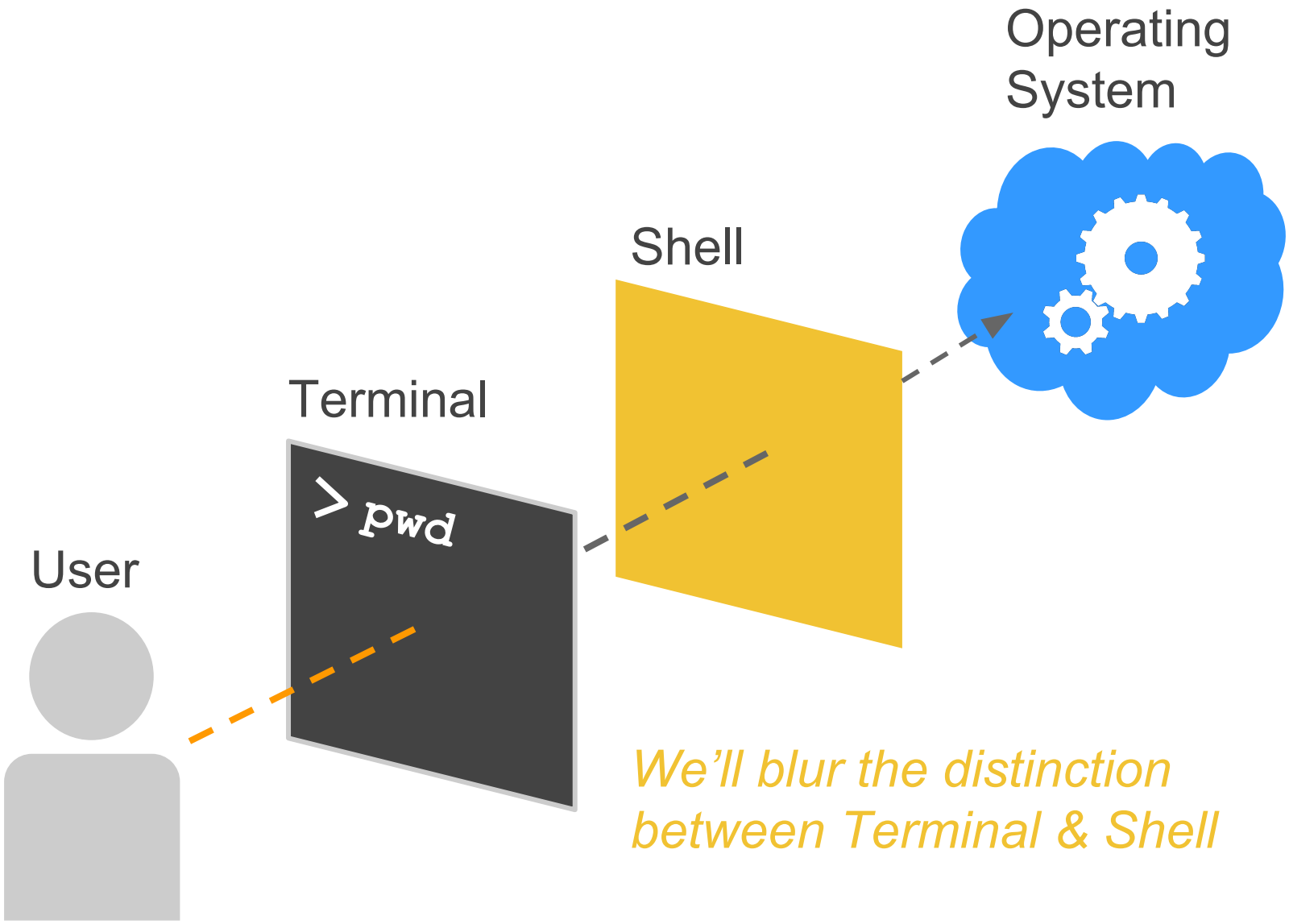
Command Line (Terminal)

Stat 133 with Gaston Sanchez



We'll blur the distinction between Terminal & Shell

Terminal


>_
THE TERMINAL

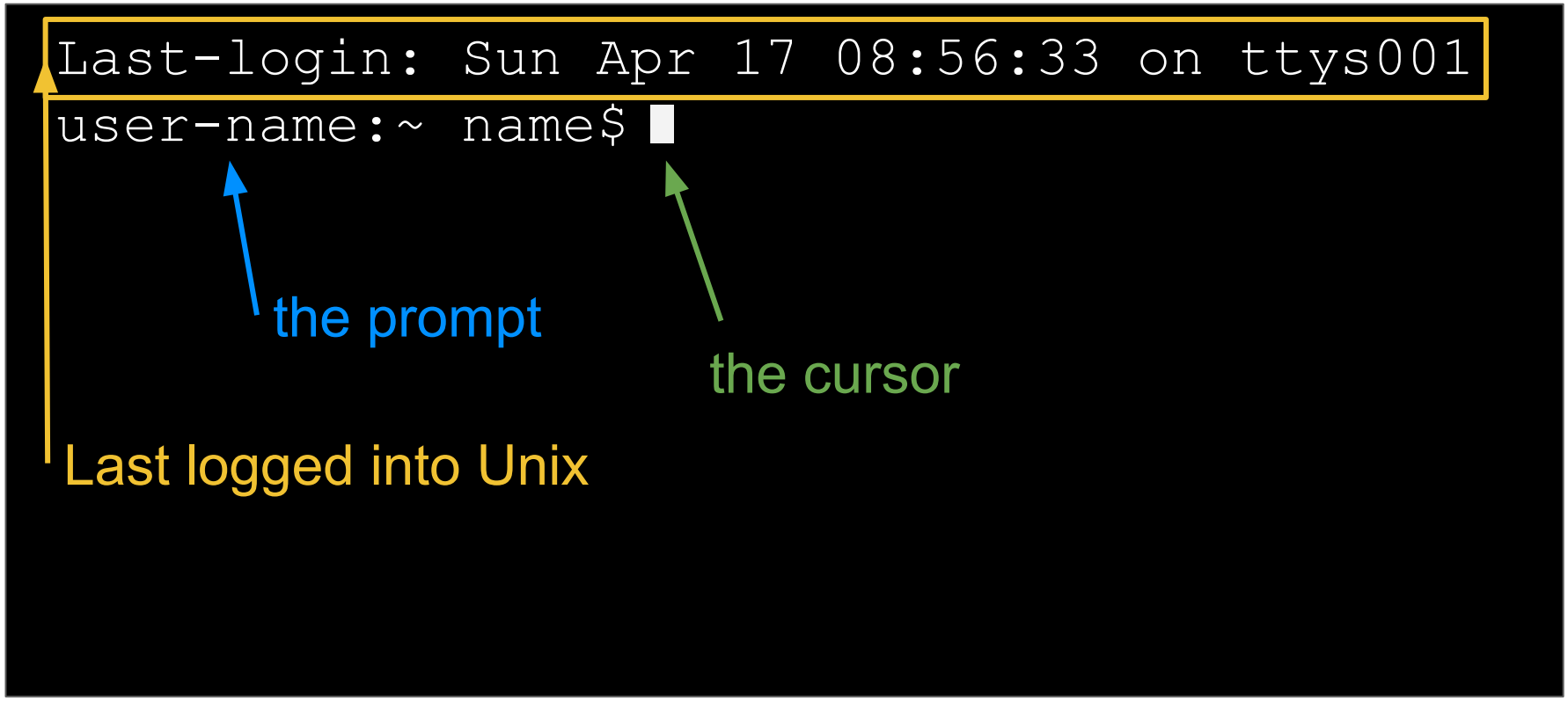
The Terminal

The way to interact with UNIX is from the command line

If you use Mac, access to the command line is with the **Terminal** Application

That's why many people use the terms "command line" and "terminal" as synonyms (although they're not the same thing)

The Terminal



Some shortcuts

Up / Down arrows: review previous commands

Ctrl + A: move cursor to start of line

Ctrl + E: move cursor to end of line

Option + click line: move cursor to click point

Ctrl + L: clear screen

Tab: try to complete the command or file

Terminal

Where is it?

How to open it?

Mac:

Applications > Utilities > Terminal

Windows:

Use Git-bash

About the shell

The shell does 4 simple things:

- displays a prompt in the terminal window (waiting for commands)
- reads your command
- runs the command
- prints the output, if any, to the Terminal window

BASH

The most common type of Shell is **Bash**

BASH: Bourne Again Shell

Default shell for Linux

Bash is usually the default shell on Mac

It is also used in Git-bash

About the terminal

The terminal's job is basically to:

- open windows and tabs
- manage shells
- resize windows
- change colors in windows
- handle copy-paste operations

Command Line Interface

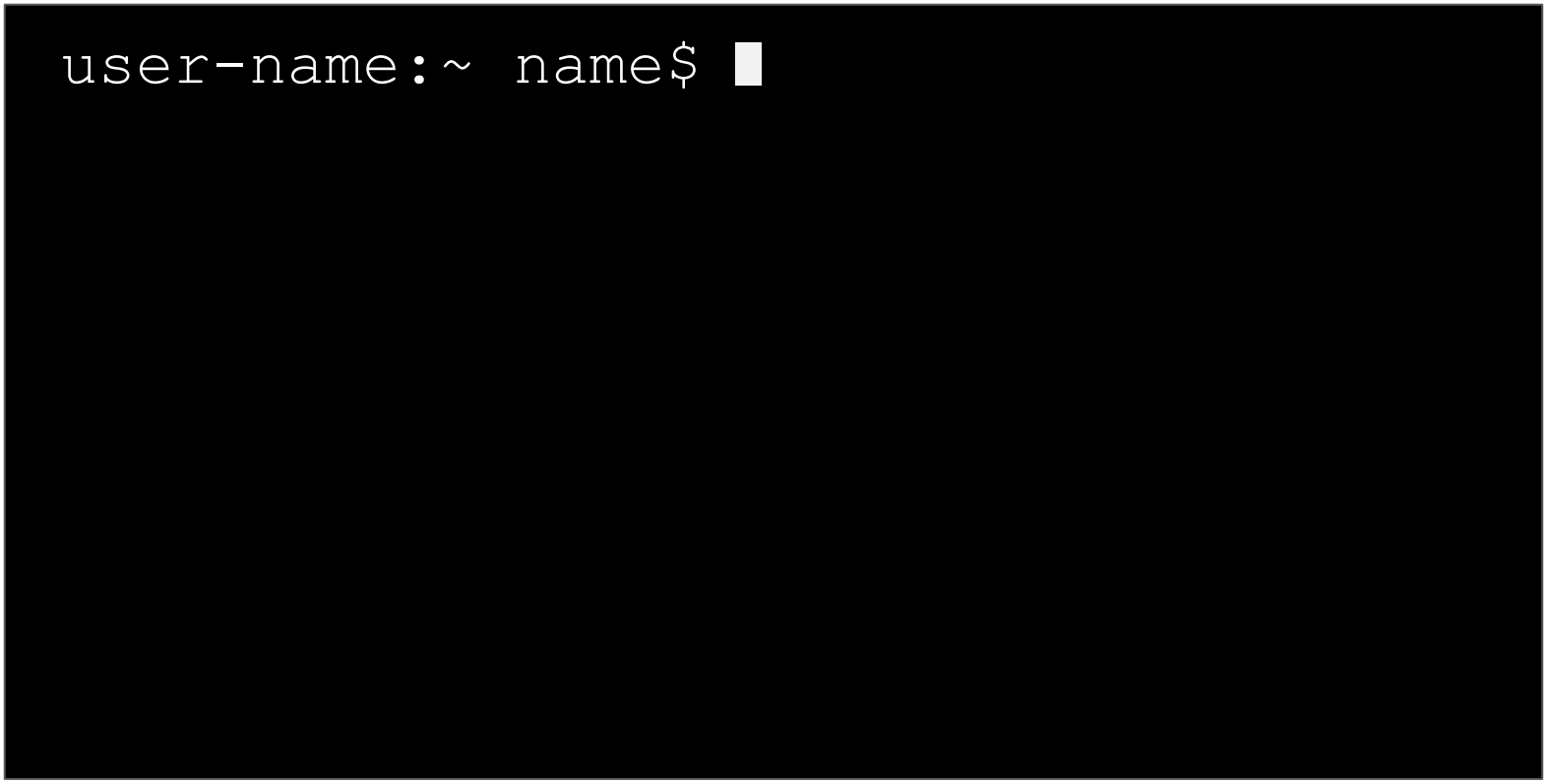
It is text-based

You type commands

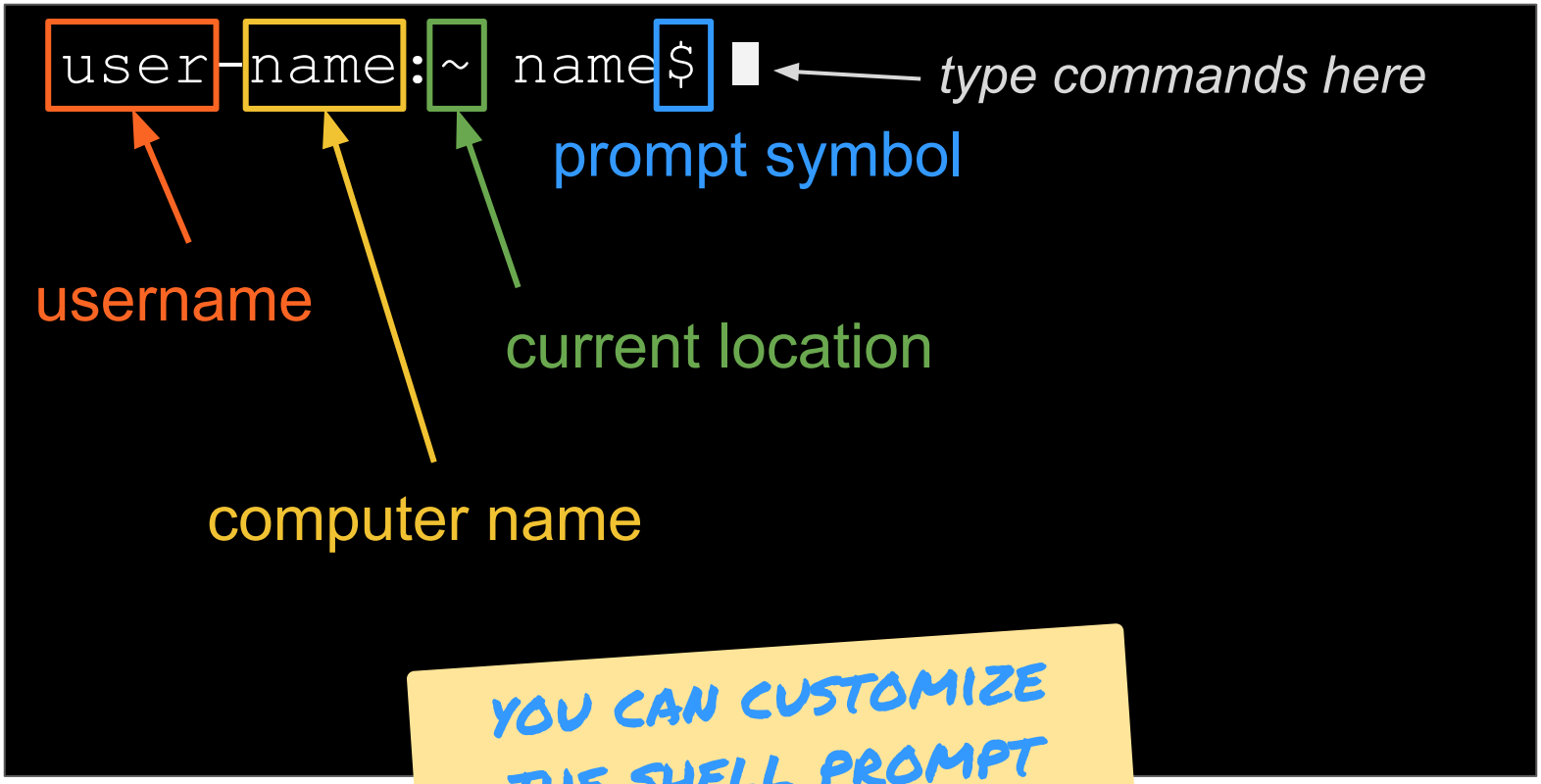
Commands are executed

… keep typing commands

The Prompt



The Prompt


user-name:~ name$
type commands here
prompt symbol
username
current location
computer name
YOU CAN CUSTOMIZE THE SHELL PROMPT

Commands

Interacting with the shell requires using commands

General Command Syntax

```
ls -lh /usr/bin

sort -u users.txt

grep -i "needle" haystack
```

```
command option(s) argument(s)
```

General Command Syntax

```
ls -lh
```

```
command option(s) argument(s)
```

General Command Syntax

```
ls /usr/bin
```

```
sort users.txt
```

```
command option(s) argument(s)
```

General Command Syntax

```
ls

sort

grep
```

The command is the program you're running

```
command option(s) argument(s)
```

General Command Syntax

```
-lh

-u

-i "needle"
```

Options tell the program how to operate

```
command option(s) argument(s)
```

Command Options

Options tell a command how to operate

Start with a dash (some with double dash)

Usually one letter

More than one offered by most commands

Options

```
ls -l -a -h /var/log
```

```
command option(s) argument(s)
```

Options

```
ls -lah /var/log
```

You can combine various options

```
command option(s) argument(s)
```

General Command Syntax

```
/usr/bin

users.txt

haystack
```

Arguments tell the command what to operate on

```
command option(s) argument(s)
```

Command Arguments

Arguments tell the command what to operate on

Usually:

- File or directory
- Set of files or folders
- Path

Command Structure

```
command␣options␣arguments
```

Single space *Single space*

ALWAYS IN THIS ORDER!

Command Structure

```
command    options    arguments
```

Name of the command

Always a single word

The thing you want to do

Command Structure

```
command    options    arguments
```

Options are optional

Not all commands require options

Controls the behavior of the command

Specified with single or double dashes

Command Structure

```
command    options    arguments
```

What to operate on:

- File(s) or directory(ies)
- Path

Manual (help) documentation

For bash (Mac or Linux users):

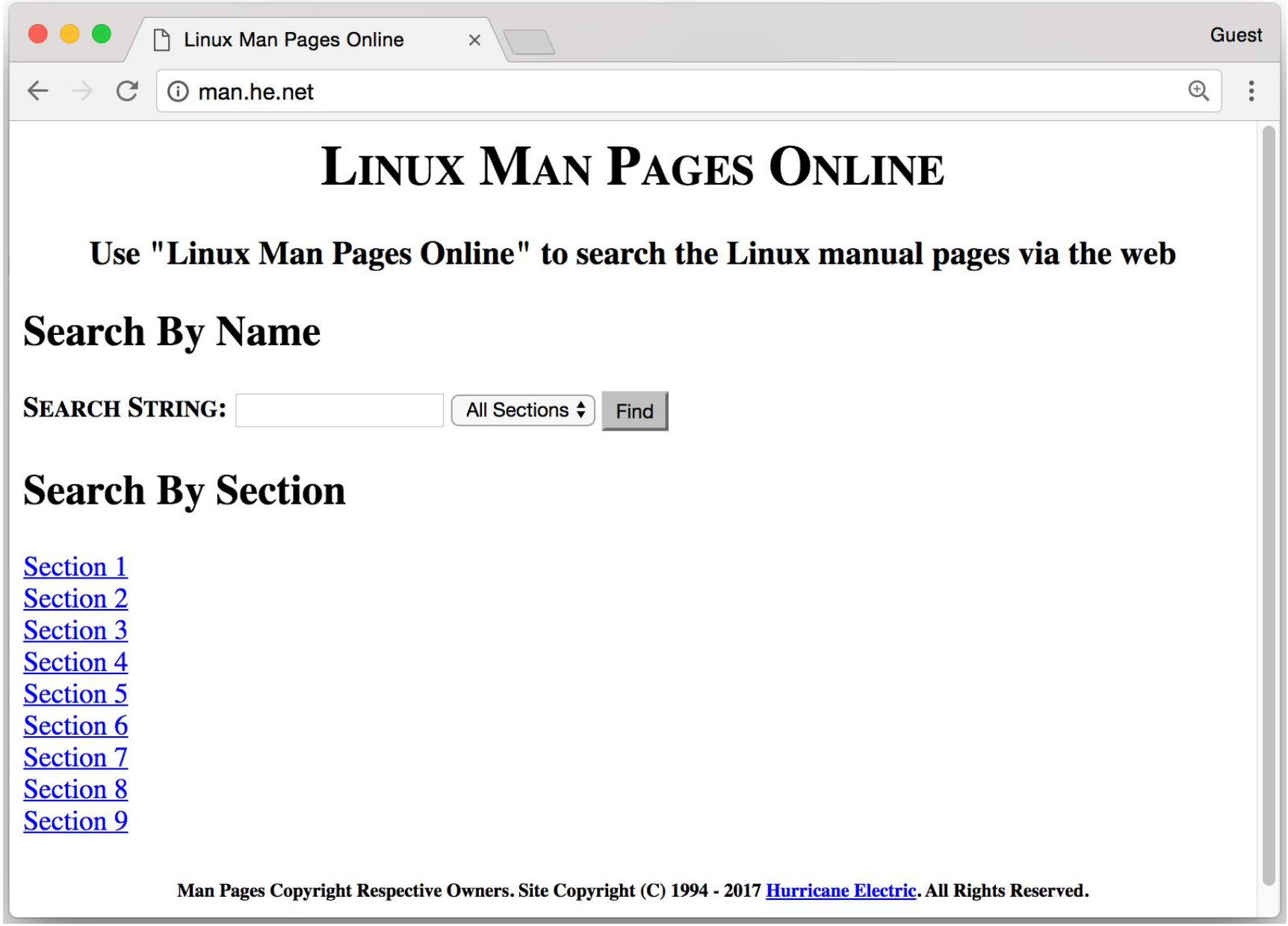
```
man <command>
```

```
e.g. man ls
```

For Git-bash (users): there is no "**man**"

```
<command> --help
```

```
e.g. ls --help
```



http://man.he.net/