

Using text Editors

Stephen A. Sefick

September 5, 2017

Outline

1 Introduction

2 Text Editors

3 Syntax Highlighting

1 Introduction

2 Text Editors

3 Syntax Highlighting

Introduction

1 Motivation

Introduction

- 1 Motivation
- 2 Let's talk about text

Introduction

- 1 Motivation
- 2 Let's talk about text
 - data

Introduction

- ① Motivation
- ② Let's talk about text
 - data
 - scripts

Introduction

- 1 Motivation
- 2 Let's talk about text
 - data
 - scripts
- 3 Syntax Highlighting

Motivation

- 1 What can we store as text?

Motivation

- 1 What can we store as text?
 - TRICK QUESTION!!!

Motivation

- 1 What can we store as text?
 - TRICK QUESTION!!!
 - EVERYTHING!!!

Motivation

- 1 What can we store as text?
 - TRICK QUESTION!!!
 - EVERYTHING!!!
- 2 Relevance to biology

Motivation

- ① What can we store as text?
 - TRICK QUESTION!!!
 - EVERYTHING!!!
- ② Relevance to biology
 - Raw Data files

Motivation

- ❶ What can we store as text?
 - TRICK QUESTION!!!
 - EVERYTHING!!!
- ❷ Relevance to biology
 - Raw Data files
 - vcf files

Motivation

- ❶ What can we store as text?
 - TRICK QUESTION!!!
 - EVERYTHING!!!
- ❷ Relevance to biology
 - Raw Data files
 - vcf files
 - bed files

Motivation

① What can we store as text?

- TRICK QUESTION!!!
- EVERYTHING!!!

② Relevance to biology

- Raw Data files
 - vcf files
 - bed files
 - fasta files

Motivation

① What can we store as text?

- TRICK QUESTION!!!
- EVERYTHING!!!

② Relevance to biology

- Raw Data files
 - vcf files
 - bed files
 - fasta files
 - data logger files

Motivation

① What can we store as text?

- TRICK QUESTION!!!
- EVERYTHING!!!

② Relevance to biology

- Raw Data files
 - vcf files
 - bed files
 - fasta files
 - data logger files
 - etc., etc., etc.

Motivation

① What can we store as text?

- TRICK QUESTION!!!
- EVERYTHING!!!

② Relevance to biology

- Raw Data files
 - vcf files
 - bed files
 - fasta files
 - data logger files
 - etc., etc., etc.
- Script files

Motivation

① What can we store as text?

- TRICK QUESTION!!!
- EVERYTHING!!!

② Relevance to biology

- Raw Data files
 - vcf files
 - bed files
 - fasta files
 - data logger files
 - etc., etc., etc.
- Script files
 - R

Motivation

① What can we store as text?

- TRICK QUESTION!!!
- EVERYTHING!!!

② Relevance to biology

- Raw Data files
 - vcf files
 - bed files
 - fasta files
 - data logger files
 - etc., etc., etc.
- Script files
 - R
 - Python

Motivation

① What can we store as text?

- TRICK QUESTION!!!
- EVERYTHING!!!

② Relevance to biology

- Raw Data files
 - vcf files
 - bed files
 - fasta files
 - data logger files
 - etc., etc., etc.
- Script files
 - R
 - Python
 - Perl

Motivation

① What can we store as text?

- TRICK QUESTION!!!
- EVERYTHING!!!

② Relevance to biology

- Raw Data files
 - vcf files
 - bed files
 - fasta files
 - data logger files
 - etc., etc., etc.
- Script files
 - R
 - Python
 - Perl
 - Bash

Motivation

① What can we store as text?

- TRICK QUESTION!!!
- EVERYTHING!!!

② Relevance to biology

- Raw Data files
 - vcf files
 - bed files
 - fasta files
 - data logger files
 - etc., etc., etc.
- Script files
 - R
 - Python
 - Perl
 - Bash
 - C, GO, etc.

Why is this so important?

- 1 plain text is persistent

Why is this so important?

- 1 plain text is persistent
 - Won't produce weird and obsolete file formats

Why is this so important?

- ❶ plain text is persistent
 - Won't produce weird and obsolete file formats
 - Every heard of Quattro Pro?

Why is this so important?

- ❶ plain text is persistent
 - Won't produce weird and obsolete file formats
 - Every heard of Quattro Pro?
 - Data lost to the world!!!

Why is this so important?

- ❶ plain text is persistent
 - Won't produce weird and obsolete file formats
 - Every heard of Quattro Pro?
 - Data lost to the world!!!
- ❷ Plain text is OS/editor independent

Why is this so important?

- ❶ plain text is persistent
 - Won't produce weird and obsolete file formats
 - Every heard of Quattro Pro?
 - Data lost to the world!!!
- ❷ Plain text is OS/editor independent
 - e.g., emacs org mode files opened on windows

Why is this so important?

- ❶ plain text is persistent
 - Won't produce weird and obsolete file formats
 - Every heard of Quattro Pro?
 - Data lost to the world!!!
- ❷ Plain text is OS/editor independent
 - e.g., emacs org mode files opened on windows
- ❸ Easily manipulated!!!

Why is this so important?

- ❶ plain text is persistent
 - Won't produce weird and obsolete file formats
 - Every heard of Quattro Pro?
 - Data lost to the world!!!
- ❷ Plain text is OS/editor independent
 - e.g., emacs org mode files opened on windows
- ❸ Easily manipulated!!!
 - Python or Perl

Why is this so important?

- ❶ plain text is persistent
 - Won't produce weird and obsolete file formats
 - Every heard of Quattro Pro?
 - Data lost to the world!!!
- ❷ Plain text is OS/editor independent
 - e.g., emacs org mode files opened on windows
- ❸ Easily manipulated!!!
 - Python or Perl
- ❹ Version control

Why is this so important?

- ❶ plain text is persistent
 - Won't produce weird and obsolete file formats
 - Every heard of Quattro Pro?
 - Data lost to the world!!!
- ❷ Plain text is OS/editor independent
 - e.g., emacs org mode files opened on windows
- ❸ Easily manipulated!!!
 - Python or Perl
- ❹ Version control
 - git!!!

1 Introduction

2 Text Editors

3 Syntax Highlighting

Types of editors

1 Linux

Types of editors

- 1 Linux
 - Vi / Vim editor

Types of editors

1 Linux

- Vi / Vim editor
- gedit editor

Types of editors

1 Linux

- Vi / Vim editor
- gedit editor
- Nano editor

Types of editors

1 Linux

- Vi / Vim editor
- gedit editor
- Nano editor
- Emacs Editor

Types of editors

1 Linux

- Vi / Vim editor
- gedit editor
- Nano editor
- Emacs Editor

2 Mac

Types of editors

1 Linux

- Vi / Vim editor
- gedit editor
- Nano editor
- Emacs Editor

2 Mac

- text wrangler

Types of editors

① Linux

- Vi / Vim editor
- gedit editor
- Nano editor
- Emacs Editor

② Mac

- text wrangler

③ Windows

Types of editors

① Linux

- Vi / Vim editor
- gedit editor
- Nano editor
- Emacs Editor

② Mac

- text wrangler

③ Windows

- emacs

Types of editors

1 Linux

- Vi / Vim editor
- gedit editor
- Nano editor
- Emacs Editor

2 Mac

- text wrangler

3 Windows

- emacs
- probably other things...

Why use/what is an editor?

- 1 Text editors are similar to word processors

Why use/what is an editor?

- 1 Text editors are similar to word processors
 - less "features" (or annoyances...)

Why use/what is an editor?

- 1 Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text

Why use/what is an editor?

- 1 Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- 2 Basic text editor generally OS supplied

Why use/what is an editor?

- 1 Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- 2 Basic text editor generally OS supplied
- 3 Main uses

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text
- ❹ Good text editors have

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text
- ❹ Good text editors have
 - Syntax highlighting

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text
- ❹ Good text editors have
 - Syntax highlighting
 - color coded text based on language

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text
- ❹ Good text editors have
 - Syntax highlighting
 - color coded text based on language
 - nice shortcuts that can speed your txt editing

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text
- ❹ Good text editors have
 - Syntax highlighting
 - color coded text based on language
 - nice shortcuts that can speed your txt editing
- ❺ A popular text editor for Linux is called 'Vi'.

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text
- ❹ Good text editors have
 - Syntax highlighting
 - color coded text based on language
 - nice shortcuts that can speed your txt editing
- ❺ A popular text editor for Linux is called 'Vi'.
 - UNIX-y.

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text
- ❹ Good text editors have
 - Syntax highlighting
 - color coded text based on language
 - nice shortcuts that can speed your txt editing
- ❺ A popular text editor for Linux is called 'Vi'.
 - UNIX-y.
 - red hat super user default editor

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text
- ❹ Good text editors have
 - Syntax highlighting
 - color coded text based on language
 - nice shortcuts that can speed your txt editing
- ❺ A popular text editor for Linux is called 'Vi'.
 - UNIX-y.
 - red hat super user default editor
- ❻ Improved to 'VIM' which means 'Vi IMproved'.

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text
- ❹ Good text editors have
 - Syntax highlighting
 - color coded text based on language
 - nice shortcuts that can speed your txt editing
- ❺ A popular text editor for Linux is called 'Vi'.
 - UNIX-y.
 - red hat super user default editor
- ❻ Improved to 'VIM' which means 'Vi IMproved'.
- ❼ VIM is hard to use... (I am an emacs man myself!)

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text
- ❹ Good text editors have
 - Syntax highlighting
 - color coded text based on language
 - nice shortcuts that can speed your txt editing
- ❺ A popular text editor for Linux is called 'Vi'.
 - UNIX-y.
 - red hat super user default editor
- ❻ Improved to 'VIM' which means 'Vi IMproved'.
- ❼ VIM is hard to use... (I am an emacs man myself!)
- ❽ Emacs is hard to use

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text
- ❹ Good text editors have
 - Syntax highlighting
 - color coded text based on language
 - nice shortcuts that can speed your txt editing
- ❺ A popular text editor for Linux is called 'Vi'.
 - UNIX-y.
 - red hat super user default editor
- ❻ Improved to 'VIM' which means 'Vi IMproved'.
- ❼ VIM is hard to use... (I am an emacs man myself!)
- ❽ Emacs is hard to use
 - definite learning curve

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text
- ❹ Good text editors have
 - Syntax highlighting
 - color coded text based on language
 - nice shortcuts that can speed your txt editing
- ❺ A popular text editor for Linux is called 'Vi'.
 - UNIX-y.
 - red hat super user default editor
- ❻ Improved to 'VIM' which means 'Vi IMproved'.
- ❼ VIM is hard to use... (I am an emacs man myself!)
- ❽ Emacs is hard to use
 - definite learning curve
 - but ...

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text
- ❹ Good text editors have
 - Syntax highlighting
 - color coded text based on language
 - nice shortcuts that can speed your txt editing
- ❺ A popular text editor for Linux is called 'Vi'.
 - UNIX-y.
 - red hat super user default editor
- ❻ Improved to 'VIM' which means 'Vi IMproved'.
- ❼ VIM is hard to use... (I am an emacs man myself!)
- ❽ Emacs is hard to use
 - definite learning curve
 - but ...
 - well worth it

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text
- ❹ Good text editors have
 - Syntax highlighting
 - color coded text based on language
 - nice shortcuts that can speed your txt editing
- ❺ A popular text editor for Linux is called 'Vi'.
 - UNIX-y.
 - red hat super user default editor
- ❻ Improved to 'VIM' which means 'Vi IMproved'.
- ❼ VIM is hard to use... (I am an emacs man myself!)
- ❽ Emacs is hard to use
 - definite learning curve
 - but ...
 - well worth it
 - can be used to replace everything!!!

Why use/what is an editor?

- ❶ Text editors are similar to word processors
 - less "features" (or annoyances...)
 - generally better at editing text
- ❷ Basic text editor generally OS supplied
- ❸ Main uses
 - write in plain text
- ❹ Good text editors have
 - Syntax highlighting
 - color coded text based on language
 - nice shortcuts that can speed your txt editing
- ❺ A popular text editor for Linux is called 'Vi'.
 - UNIX-y.
 - red hat super user default editor
- ❻ Improved to 'VIM' which means 'Vi IMproved'.
- ❼ VIM is hard to use... (I am an emacs man myself!)
- ❽ Emacs is hard to use
 - definite learning curve
 - but ...
 - well worth it
 - can be used to replace everything!!!
- ❾ We will focus on other editors

Types of editors

- 1 Two main types of editors.

Types of editors

- 1 Two main types of editors.
 - Console based editors

Types of editors

- 1 Two main types of editors.
 - Console based editors
 - GUI based editors

Types of editors

- 1 Two main types of editors.
 - Console based editors
 - GUI based editors
- 2 Console based editors

Types of editors

- ❶ Two main types of editors.
 - Console based editors
 - GUI based editors
- ❷ Console based editors
 - emacs

Types of editors

- ❶ Two main types of editors.
 - Console based editors
 - GUI based editors
- ❷ Console based editors
 - emacs
 - nano

Types of editors

- ❶ Two main types of editors.
 - Console based editors
 - GUI based editors
- ❷ Console based editors
 - emacs
 - nano
 - vim

Types of editors

- ❶ Two main types of editors.
 - Console based editors
 - GUI based editors
- ❷ Console based editors
 - emacs
 - nano
 - vim
- ❸ GUI based editors

Types of editors

- ❶ Two main types of editors.
 - Console based editors
 - GUI based editors
- ❷ Console based editors
 - emacs
 - nano
 - vim
- ❸ GUI based editors
 - PC

Types of editors

- ❶ Two main types of editors.
 - Console based editors
 - GUI based editors
- ❷ Console based editors
 - emacs
 - nano
 - vim
- ❸ GUI based editors
 - PC
 - gedit,

Types of editors

- ❶ Two main types of editors.
 - Console based editors
 - GUI based editors
- ❷ Console based editors
 - emacs
 - nano
 - vim
- ❸ GUI based editors
 - PC
 - gedit,
 - TextWrangler

Types of editors

- ❶ Two main types of editors.
 - Console based editors
 - GUI based editors
- ❷ Console based editors
 - emacs
 - nano
 - vim
- ❸ GUI based editors
 - PC
 - gedit,
 - TextWrangler
 - Linux

Types of editors

1 Two main types of editors.

- Console based editors
- GUI based editors

2 Console based editors

- emacs
- nano
- vim

3 GUI based editors

- PC
 - gedit,
 - TextWrangler
- Linux
 - gedit

Types of editors

1 Two main types of editors.

- Console based editors
- GUI based editors

2 Console based editors

- emacs
- nano
- vim

3 GUI based editors

- PC
 - gedit,
 - TextWrangler
- Linux
 - gedit
 - emacs

Types of editors

1 Two main types of editors.

- Console based editors
- GUI based editors

2 Console based editors

- emacs
- nano
- vim

3 GUI based editors

- PC
 - gedit,
 - TextWrangler
- Linux
 - gedit
 - emacs
- Mac

Types of editors

1 Two main types of editors.

- Console based editors
- GUI based editors

2 Console based editors

- emacs
- nano
- vim

3 GUI based editors

- PC
 - gedit,
 - TextWrangler
- Linux
 - gedit
 - emacs
- Mac
 - TextWrangler

Console based editors

❶ \$ nano somefile.txt

Console based editors

- 1 `$ nano somefile.txt`
- 2 `$ emacs -nw somefile.txt`

Console based editors

- 1 \$ nano somefile.txt
- 2 \$ emacs -nw somefile.txt
- 3 \$ vi somefile.txt

Console based editors

- ❶ `$ nano somefile.txt`
- ❷ `$ emacs -nw somefile.txt`
- ❸ `$ vi somefile.txt`
- ❹ `somefile.txt`

Console based editors

- ❶ `$ nano somefile.txt`
- ❷ `$ emacs -nw somefile.txt`
- ❸ `$ vi somefile.txt`
- ❹ `somefile.txt`
 - exist or not in present working directory (`pwd`)

Console based editors

- ❶ `$ nano somefile.txt`
- ❷ `$ emacs -nw somefile.txt`
- ❸ `$ vi somefile.txt`
- ❹ `somefile.txt`
 - exist or not in present working directory (`pwd`)
 - if exists will open

Console based editors

- ❶ `$ nano somefile.txt`
- ❷ `$ emacs -nw somefile.txt`
- ❸ `$ vi somefile.txt`
- ❹ `somefile.txt`
 - exist or not in present working directory (`pwd`)
 - if exists will open
 - else will create

Console based editors

- ❶ `$ nano somefile.txt`
- ❷ `$ emacs -nw somefile.txt`
- ❸ `$ vi somefile.txt`
- ❹ `somefile.txt`
 - exist or not in present working directory (`pwd`)
 - if exists will open
 - else will create
 - `PATH` is important

Console based editors

- ❶ `$ nano somefile.txt`
- ❷ `$ emacs -nw somefile.txt`
- ❸ `$ vi somefile.txt`
- ❹ `somefile.txt`
 - exist or not in present working directory (`pwd`)
 - if exists will open
 - else will create
 - `PATH` is important
 - `emacs -nw somefile.txt` (in `~/Desktop/`)

Console based editors

- ❶ `$ nano somefile.txt`
- ❷ `$ emacs -nw somefile.txt`
- ❸ `$ vi somefile.txt`
- ❹ `somefile.txt`
 - exist or not in present working directory (`pwd`)
 - if exists will open
 - else will create
 - `PATH` is important
 - `emacs -nw somefile.txt` (in `~/Desktop/`)
 - `emacs -nw ~/somefile.txt`

1 OS: Linux and Mac

Nano

- 1 OS: Linux and Mac
- 2 GNU Nano written in 1999 by Chris Allegretta.

Nano

- 1 OS: Linux and Mac
- 2 GNU Nano written in 1999 by Chris Allegretta.
- 3 GNU clone of Pico editor.

Nano

- 1 OS: Linux and Mac
- 2 GNU Nano written in 1999 by Chris Allegretta.
- 3 GNU clone of Pico editor.
- 4 GNU Nano is a small and friendly text editor

Nano

- 1 OS: Linux and Mac
- 2 GNU Nano written in 1999 by Chris Allegretta.
- 3 GNU clone of Pico editor.
- 4 GNU Nano is a small and friendly text editor
- 5 Besides basic text editing

Nano

- 1 OS: Linux and Mac
- 2 GNU Nano written in 1999 by Chris Allegretta.
- 3 GNU clone of Pico editor.
- 4 GNU Nano is a small and friendly text editor
- 5 Besides basic text editing
 - interactive search and replace

Nano

- ❶ OS: Linux and Mac
- ❷ GNU Nano written in 1999 by Chris Allegretta.
- ❸ GNU clone of Pico editor.
- ❹ GNU Nano is a small and friendly text editor
- ❺ Besides basic text editing
 - interactive search and replace
 - go to line and column number

Nano

- ❶ OS: Linux and Mac
- ❷ GNU Nano written in 1999 by Chris Allegretta.
- ❸ GNU clone of Pico editor.
- ❹ GNU Nano is a small and friendly text editor
- ❺ Besides basic text editing
 - interactive search and replace
 - go to line and column number
 - auto-indentation

Nano

- ❶ OS: Linux and Mac
- ❷ GNU Nano written in 1999 by Chris Allegretta.
- ❸ GNU clone of Pico editor.
- ❹ GNU Nano is a small and friendly text editor
- ❺ Besides basic text editing
 - interactive search and replace
 - go to line and column number
 - auto-indentation
 - feature toggles

Nano

- ❶ OS: Linux and Mac
- ❷ GNU Nano written in 1999 by Chris Allegretta.
- ❸ GNU clone of Pico editor.
- ❹ GNU Nano is a small and friendly text editor
- ❺ Besides basic text editing
 - interactive search and replace
 - go to line and column number
 - auto-indentation
 - feature toggles
 - internationalization support

Nano

- ❶ OS: Linux and Mac
- ❷ GNU Nano written in 1999 by Chris Allegretta.
- ❸ GNU clone of Pico editor.
- ❹ GNU Nano is a small and friendly text editor
- ❺ Besides basic text editing
 - interactive search and replace
 - go to line and column number
 - auto-indentation
 - feature toggles
 - internationalization support
 - filename tab completion

- 1 OS: Unix, Linux and Windows

VI/VIM editor

- ❶ OS: Unix, Linux and Windows
- ❷ First Vim editor built in 1991

VI/VIM editor

- 1 OS: Unix, Linux and Windows
- 2 First Vim editor built in 1991
- 3 Vim

VI/VIM editor

- 1 OS: Unix, Linux and Windows
- 2 First Vim editor built in 1991
- 3 Vim
 - command-line interface (CLI)

VI/VIM editor

- 1 OS: Unix, Linux and Windows
- 2 First Vim editor built in 1991
- 3 Vim
 - command-line interface (CLI)
 - graphical user interface (GUI)

VI/VIM editor

- ❶ OS: Unix, Linux and Windows
- ❷ First Vim editor built in 1991
- ❸ Vim
 - command-line interface (CLI)
 - graphical user interface (GUI)
 - gVim is the graphical version

EMACS!!!!!!!!!!!!!!

- 1 OS: Unix, Linux and Windows

EMACS!!!!!!!!!!!!!!

- ① OS: Unix, Linux and Windows
- ② first written in 1972 at MIT

EMACS!!!!!!!!!!!!!!

- ① OS: Unix, Linux and Windows
- ② first written in 1972 at MIT
- ③ GNU Emacs written 1984 by Richard Stallman

EMACS!!!!!!!!!!!!!!

- ① OS: Unix, Linux and Windows
- ② first written in 1972 at MIT
- ③ GNU Emacs written 1984 by Richard Stallman
- ④ Emacs

EMACS!!!!!!!!!!!!!!

- 1 OS: Unix, Linux and Windows
- 2 first written in 1972 at MIT
- 3 GNU Emacs written 1984 by Richard Stallman
- 4 Emacs
 - emacs -nw file; command-line interface (CLI)

EMACS!!!!!!!!!!!!!!

- ① OS: Unix, Linux and Windows
- ② first written in 1972 at MIT
- ③ GNU Emacs written 1984 by Richard Stallman
- ④ Emacs
 - emacs -nw file; command-line interface (CLI)
 - emacs file; graphical user interface (GUI)

EMACS!!!!!!!!!!!!!!

- ① OS: Unix, Linux and Windows
- ② first written in 1972 at MIT
- ③ GNU Emacs written 1984 by Richard Stallman
- ④ Emacs
 - emacs -nw file; command-line interface (CLI)
 - emacs file; graphical user interface (GUI)
 - supports everything!!!

EMACS!!!!!!!!!!!!!!

- ❶ OS: Unix, Linux and Windows
- ❷ first written in 1972 at MIT
- ❸ GNU Emacs written 1984 by Richard Stallman
- ❹ Emacs
 - emacs -nw file; command-line interface (CLI)
 - emacs file; graphical user interface (GUI)
 - supports everything!!!
- ❺ Must install

EMACS!!!!!!!!!!!!!!

- ❶ OS: Unix, Linux and Windows
- ❷ first written in 1972 at MIT
- ❸ GNU Emacs written 1984 by Richard Stallman
- ❹ Emacs
 - emacs -nw file; command-line interface (CLI)
 - emacs file; graphical user interface (GUI)
 - supports everything!!!
- ❺ Must install
 - What linux distro thinlink?

- 1 OS: Mac

TextWrangler

- 1 OS: Mac
- 2 I know people love it.

TextWrangler

- 1 OS: Mac
- 2 I know people love it.
- 3 Looks like TextWrangler is being "sunsetting"

TextWrangler

- 1 OS: Mac
- 2 I know people love it.
- 3 Looks like TextWrangler is being "sunsetting"
 - use BBedit

TextWrangler

- 1 OS: Mac
- 2 I know people love it.
- 3 Looks like TextWrangler is being "sunsetting"
 - use BBedit
- 4 <https://www.barebones.com/products/textwrangler/>

- 1 When I used Mac. . .

- 1 When I used Mac. . .
- 2 this editor ruled!

- 1 When I used Mac...
- 2 this editor ruled!
- 3 <https://www.peterborgapps.com/smultron/>

1 Introduction

2 Text Editors

3 Syntax Highlighting

What is Syntax Highlighting?

- ❶ colorful text!
- ❷ importantly it helps
 - Readability
 - error finding
 - speed
- ❸ Used?
- ❹ In text editors
- ❺ Scripts

Let's set Nano up

- 1 open `hello_python.py`

Let's set Nano up

- 1 open `hello_python.py`
 - `nano hello_python.py`

Let's set Nano up

- 1 open `hello_python.py`
 - `nano hello_python.py`
 - notice not colorful

Let's set Nano up

- 1 open `hello_python.py`
 - `nano hello_python.py`
 - notice not colorful
- 2 `chmod +x hello_python.py`

Let's set Nano up

- ❶ open `hello_python.py`
 - `nano hello_python.py`
 - notice not colorful
- ❷ `chmod +x hello_python.py`
- ❸ `./hello_python`

Let's set Nano up

- 1 open `hello_python.py`
 - `nano hello_python.py`
 - notice not colorful
- 2 `chmod +x hello_python.py`
- 3 `./hello_python`
- 4 Do you have `.nanorc`?

Let's set Nano up

- ❶ open `hello_python.py`
 - `nano hello_python.py`
 - notice not colorful
- ❷ `chmod +x hello_python.py`
- ❸ `./hello_python`
- ❹ Do you have `.nanorc`?
- ❺ if no proceed to 6; if yes proceed to 8

Let's set Nano up

- ❶ open `hello_python.py`
 - `nano hello_python.py`
 - notice not colorful
- ❷ `chmod +x hello_python.py`
- ❸ `./hello_python`
- ❹ Do you have `.nanorc`?
- ❺ if no proceed to 6; if yes proceed to 8
- ❻ `cp etc/nanorc ~/.nanorc`

Let's set Nano up

- ❶ open `hello_python.py`
 - `nano hello_python.py`
 - notice not colorful
- ❷ `chmod +x hello_python.py`
- ❸ `./hello_python`
- ❹ Do you have `.nanorc`?
- ❺ if no proceed to 6; if yes proceed to 8
- ❻ `cp etc/nanorc ~/.nanorc`
- ❼ `nano ~/.nanorc`

Let's set Nano up

- ❶ open `hello_python.py`
 - `nano hello_python.py`
 - notice not colorful
- ❷ `chmod +x hello_python.py`
- ❸ `./hello_python`
- ❹ Do you have `.nanorc`?
- ❺ if no proceed to 6; if yes proceed to 8
- ❻ `cp etc/nanorc ~/.nanorc`
- ❼ `nano ~/.nanorc`
- ❽ uncomment whatever languages you use

Let's set Nano up

- ❶ open `hello_python.py`
 - `nano hello_python.py`
 - notice not colorful
- ❷ `chmod +x hello_python.py`
- ❸ `./hello_python`
- ❹ Do you have `.nanorc`?
- ❺ if no proceed to 6; if yes proceed to 8
- ❻ `cp etc/nanorc ~/.nanorc`
- ❼ `nano ~/.nanorc`
- ❽ uncomment whatever languages you use
 - `python`

Let's set Nano up

- ❶ open `hello_python.py`
 - `nano hello_python.py`
 - notice not colorful
- ❷ `chmod +x hello_python.py`
- ❸ `./hello_python`
- ❹ Do you have `.nanorc`?
- ❺ if no proceed to 6; if yes proceed to 8
- ❻ `cp etc/nanorc ~/.nanorc`
- ❼ `nano ~/.nanorc`
- ❽ uncomment whatever languages you use
 - python
 - R

Let's set Nano up

- ❶ open `hello_python.py`
 - `nano hello_python.py`
 - notice not colorful
- ❷ `chmod +x hello_python.py`
- ❸ `./hello_python`
- ❹ Do you have `.nanorc`?
- ❺ if no proceed to 6; if yes proceed to 8
- ❻ `cp etc/nanorc ~/.nanorc`
- ❼ `nano ~/.nanorc`
- ❽ uncomment whatever languages you use
 - python
 - R
 - bash

Let's set Nano up

- ❶ open `hello_python.py`
 - `nano hello_python.py`
 - notice not colorful
- ❷ `chmod +x hello_python.py`
- ❸ `./hello_python`
- ❹ Do you have `.nanorc`?
- ❺ if no proceed to 6; if yes proceed to 8
- ❻ `cp etc/nanorc ~/.nanorc`
- ❼ `nano ~/.nanorc`
- ❽ uncomment whatever languages you use
 - python
 - R
 - bash
- ❾ open `hello_python.py`

Let's set Nano up

- ❶ open `hello_python.py`
 - `nano hello_python.py`
 - notice not colorful
- ❷ `chmod +x hello_python.py`
- ❸ `./hello_python`
- ❹ Do you have `.nanorc`?
- ❺ if no proceed to 6; if yes proceed to 8
- ❻ `cp etc/nanorc ~/.nanorc`
- ❼ `nano ~/.nanorc`
- ❽ uncomment whatever languages you use
 - python
 - R
 - bash
- ❾ open `hello_python.py`
 - notice the wonderful colors!

Let's set Nano up

- ❶ open `hello_python.py`
 - `nano hello_python.py`
 - notice not colorful
- ❷ `chmod +x hello_python.py`
- ❸ `./hello_python`
- ❹ Do you have `.nanorc`?
- ❺ if no proceed to 6; if yes proceed to 8
- ❻ `cp etc/nanorc ~/.nanorc`
- ❼ `nano ~/.nanorc`
- ❽ uncomment whatever languages you use
 - python
 - R
 - bash
- ❾ open `hello_python.py`
 - notice the wonderful colors!
- ❿ That's syntax highlighting!