

LING 300 - Topics in Linguistics:
Introduction to Programming and Text Processing for Linguists

Week 1

—

Intro,
Unix, Shell,
Environment, Files

Who are we?

Rob Voigt

`robvoigt@northwestern.edu`

Assistant Professor
of Linguistics

Thomas Sostarics

`tsostarics@northwestern.edu`

PhD Student
in Linguistics

Who is this class for?

- Linguists, social scientists, humanists
- Little-to-no programming experience
- Applications to research

Goals

- Lots of hands-on practice
- Teach you how to teach yourself



Who is this class *not* for?

- Folks with lots of programming experience
- CS Majors (probably - email me if this is you)
- COMP_SCI 110 is similar in focus (and uses one of the same textbooks) - what's different?
 - CS110 - broad, more CS-y (e.g. debugging and testing)
 - LING300 - narrow focus on applications to text, we will purposefully skip less-relevant stuff



What will we learn?

- Unix Command Line

basic usage, remote access, and tools for text

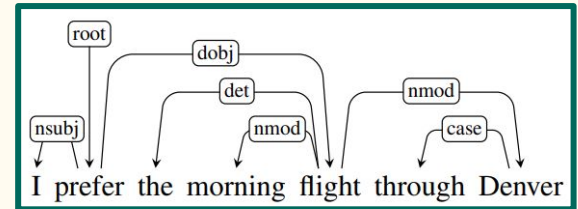
- Basic Python

programming concepts, syntax, useful libraries for text

- Applications (as much as we have time)

web scraping, APIs,
data munging, text analysis

```
[rfj5679@quser21 COHA]$ ls
db_lexicon_coha.zip sources.zip text urls.txt wc
[rfj5679@quser21 COHA]$ cd db
[rfj5679@quser21 db]$ ls
db_1810s_kwp.zip db_1860s_msl.zip db_1910s_aow.zip
db_1820s_lse.zip db_1870s_fhs.zip db_1920s_bsj.zip
db_1830s_sje.zip db_1880s_xjs.zip db_1930s_bkk.zip
db_1840s_ieo.zip db_1890s_lsp.zip db_1940s_jsk.zip
db_1850s_qoe.zip db_1900s_ahs.zip db_1950s_shy.zip
[rfj5679@quser21 db]$ for i in *zip; do unzip $i; done
Archive: db_1810s_kwp.zip
  inflating: 1810.txt
```



When and where will we see each other?

Zoom at normal class times (optional but highly recommended)
short lectures, breakout room workshopping
recorded if you can't make it

Office hours	<i>Rob</i>	Wednesdays noon-1pm and by appt
	<i>Thomas</i>	Mondays 11am-noon, Fridays 1-2pm

Piazza discussion board for questions
help each other out!

Why are we doing this?

1. Get computationally “free” -
GUIs only let you do things someone else decided on
2. Processing text data is useful for anyone’s research
3. This is the start of computational linguistics!
web search, speech-to-text, conversational AI,
“big data” language analysis, etc etc

How will we do it?

Syllabus on course website:

https://faculty.wcas.northwestern.edu/robvoigt/courses/2021_winter/ling300/

Assignments, peer review, final project

Assignments generally out on Monday,
due the following Sunday night

Videos/readings before class; working on assignments during
in collaborative breakout rooms

How will we do grading?

Heavy emphasis on qualitative feedback

Thomas primary grader,

I'll read your comments and be spot-checking

Numerical grades based on effortful completion,

Midterm and final self-evaluations

The point of this whole thing is for you to learn, period!

The Struggle!

Learning programming is like learning a new language

You have to soak in it and use it daily

It will feel unnatural at first, push through

Don't be afraid to play around and break stuff

The Struggle Illustrated

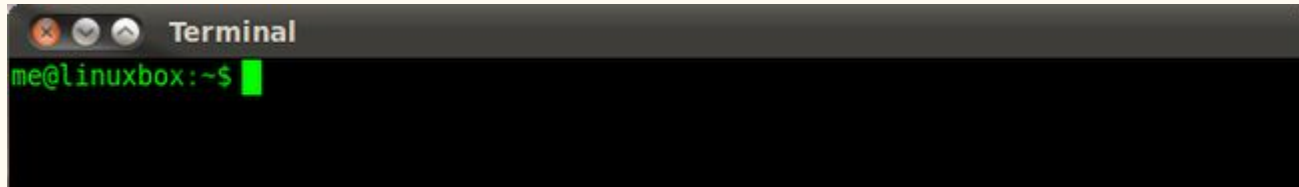


YOU
CAN
DO
IT

No such thing as a
dumb question here.

ERRORS
ARE
YOUR
NEW
FRIENDS

Our new home: the command line



Precision - the challenge of exactitude

One wrong letter, space, or punctuation mark
can easily derail you

These mistakes are at first *very hard to see*

Double-check, triple-check your code
and relevant documentation

(a beloved acronym by programmers is RTFM - read the flippin' manual!)

Take a break and come back to it

Benefits of command line interfaces

Automatable

easy to do
something 1000x

Fast

GUI interfaces are
computationally ‘heavy’

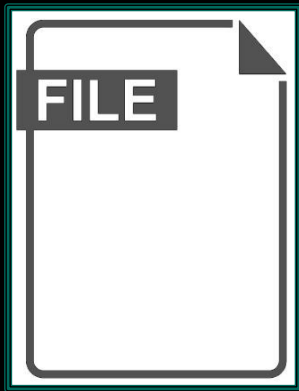
Consistent

same command always
does the same thing

Transparent

you’ll learn what your files
actually are

What is a file?



An abstraction!

... but ultimately,
an array of bytes

e.g., for ASCII text:

<i>Character</i>	L	I	N	G
<i>Bits</i>	100 1100	100 1001	100 1110	100 0111

Types of Files

Text

bytes representing characters
txt, code (like .py), html, logs

Executable

compiled code in binary format
to run as a program

Data

everything else: images, zip files,
doc/ppt/pdf, and so on

**file
extensions
are just a
helpful
suggestion!**

Quest!

Remote computing environment,
cluster of computers running Linux

Common for “big data” and
high-performance tasks

Can schedule complex stuff,
not waste your own machine

Ideal to use Quest
exclusively if you can

If it is slow because of
where you are, you can do
everything locally, then
upload assignments