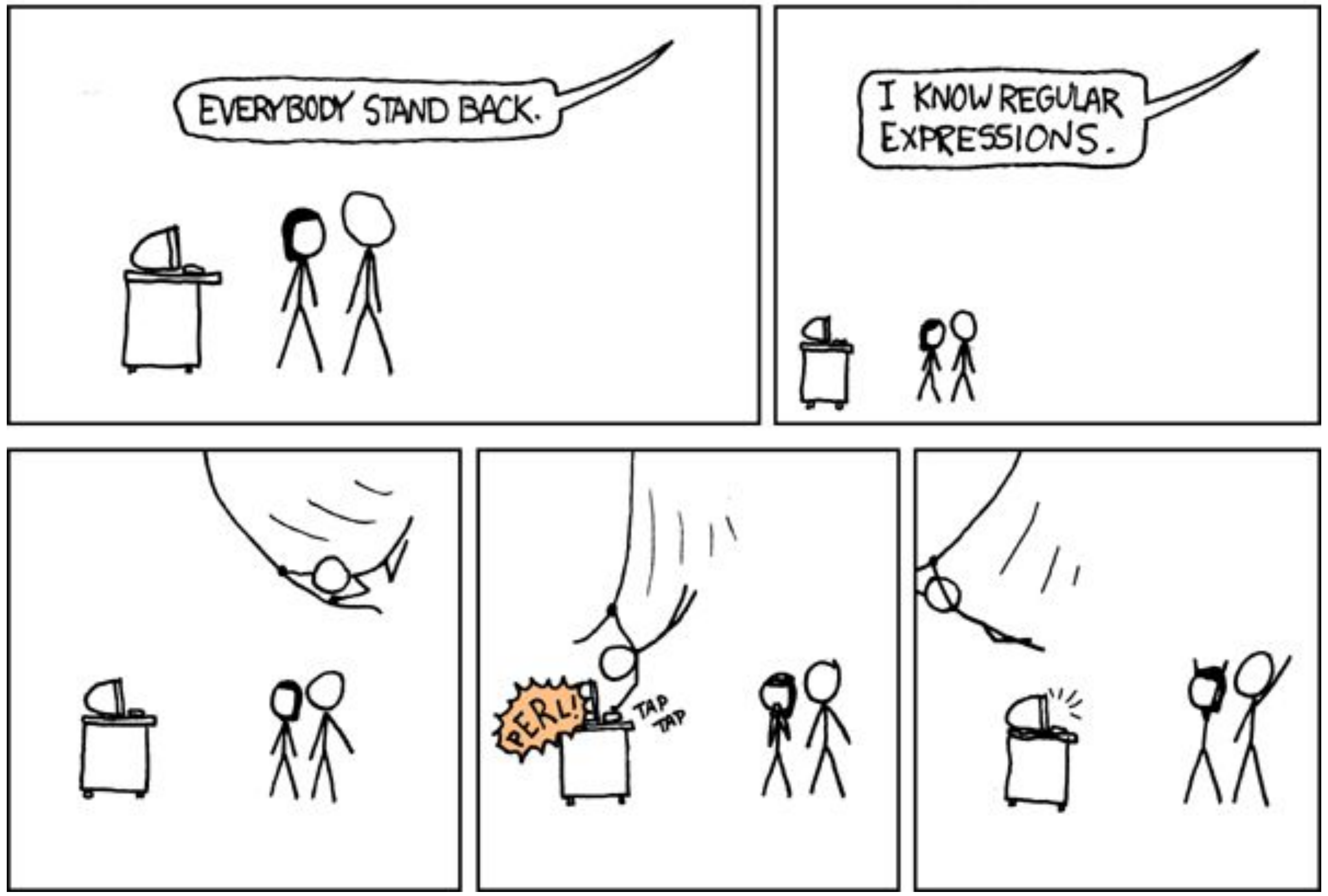
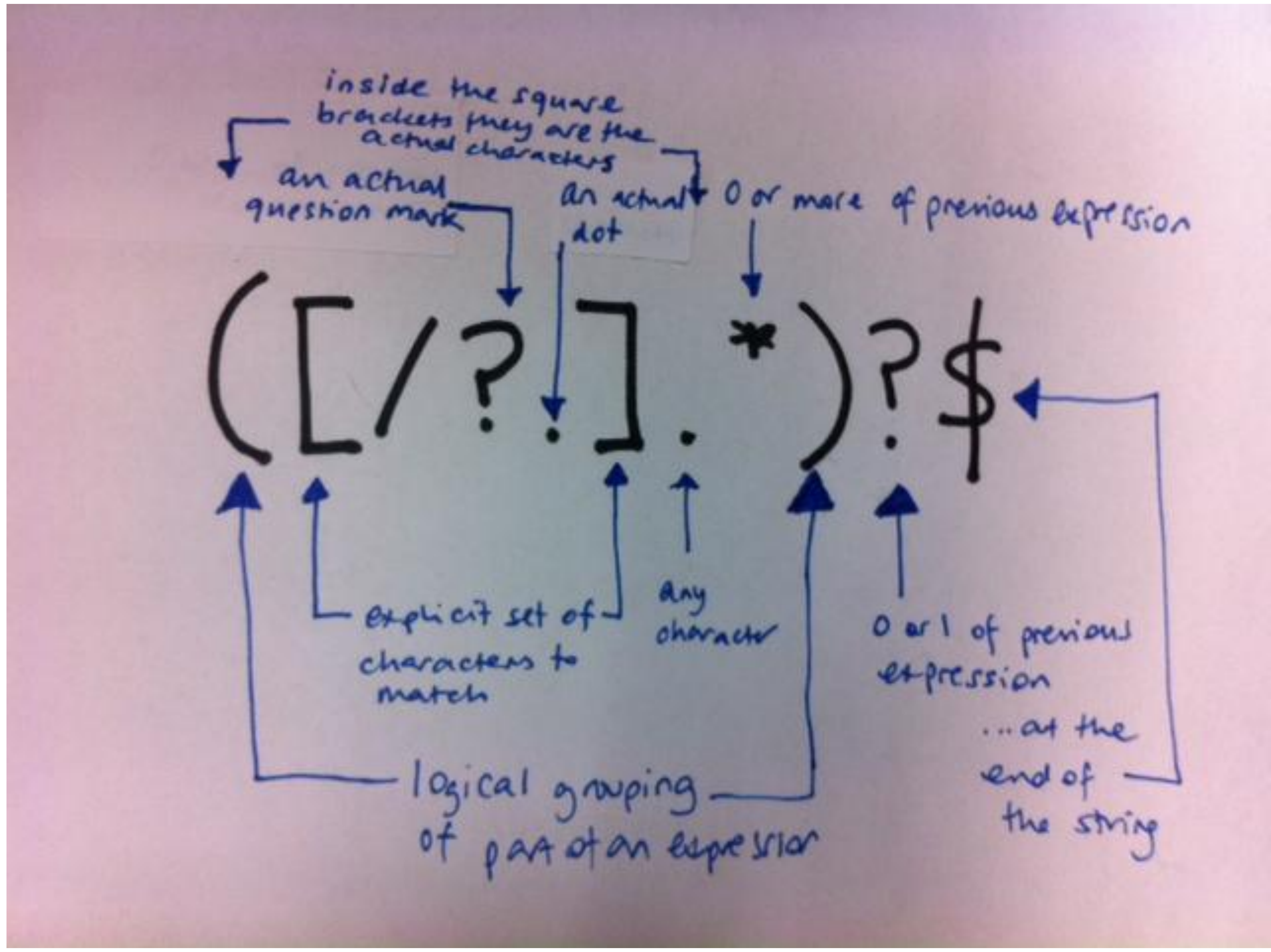
Intro & Regular Expressions

Spring 2025, Week 1
January 17, 2025




inside the square brackets they are the actual characters
an actual question mark
an actual dot
0 or more of previous expression
([/?.].*)?$
explicit set of characters to match
any character
0 or 1 of previous expression
...at the end of the string
logical grouping of part of an expression

Check-in, intros

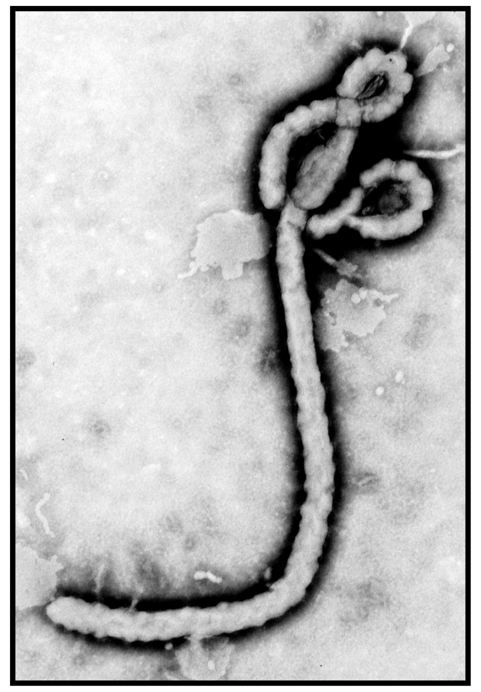
PhD - Evolutionary genetics

HOPKINS
MARINE STATION
OF STANFORD UNIVERSITY

PostDoc - Pathogen genomics

USAMRIID
United States Army
Medical Research Institute
of Infectious Diseases

Biodefense solutions to protect our nation



PATHOGEN & MICROBIOME INSTITUTE
NAU & TGen

Associate Professor
Dept of Biological Sciences
Pathogen and Microbiome Institute

Intros

1. Your name (Optional: pronouns)
2. Check-in
3. Your research focus
4. A data processing/analysis challenge

Outline

- Course organization
- Plain text files
- Regular expressions

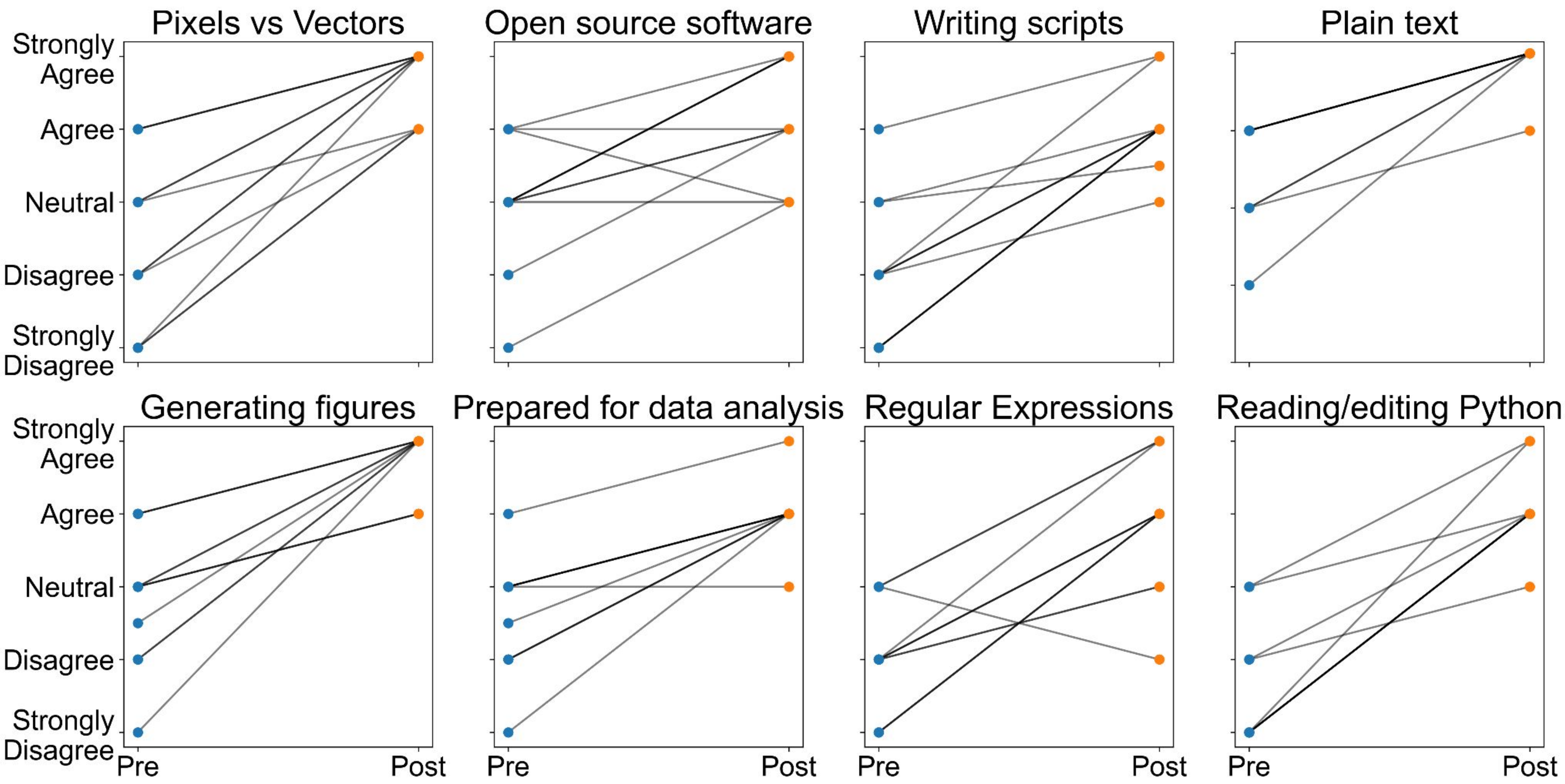
Course organization

What this course is: Intro to general computing techniques broadly applicable to many research-related tasks

What it isn't: A bioinformatics class

Goal: Introduce a variety of potentially useful tools

2024 Assessment Results



2024 Assessment Results

Pixels vs Vectors
Open source software
Writing scripts
Plain text
Generating figures
Prepared for data analysis
Regular Expressions
Reading/editing Python
−1
0
1
2
3
4
Change in Score

Syllabus on Canvas

NAU
Account
Dashboard
Courses
Calendar
Inbox
History
My Media
Studio
Help
BIO-682 - SEC001 - QUANTITATIVE BIOLOGY › Syllabus
Immersive Reader
Spring 2025
Home
Modules
Announcements
Syllabus
Grades
Assignments
CoursEvals
Jump to Today
Page 1 of 4
ZOOM

Course Syllabus

Syllabus_BIO682_Spring2025.pdf

Minimize File Preview

College of Environment, Forestry and Natural Sciences
BIO 682, Quantitative Biology
Spring 2025, 3 credit hours

Instructor: Jason Ladner

Contact Info: ARD Building (#56) room 245
Phone: 928-523-0647
Email: Jason.Ladner@nau.edu

Office Hours: TBD

Course Purpose:

As datasets continue to increase in complexity, computational analysis is becoming a routine part of scientific research. This is a hands-on training course that teaches you how to use general computing tools to work more efficiently and

January 2025

29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	1
2	3	4	5	6	7	8

Assignments are weighted by group:

Group	Weight
Assignments	33%
Attendance_Participation	33%
Final Project	34%
Reading	0%
Total	100%

Required text

- Haddock, S. H. D. and Dunn, C. W. (2010). Practical Computing for Biologists. Sinauer Associates

- http://practicalcomputing.org/

- Reading must be complete **PRIOR** to class

Class organization

New Content

(First 11 weeks)

- Lectures
- Demos
- In class work time
  (homework assignments)

Individual projects

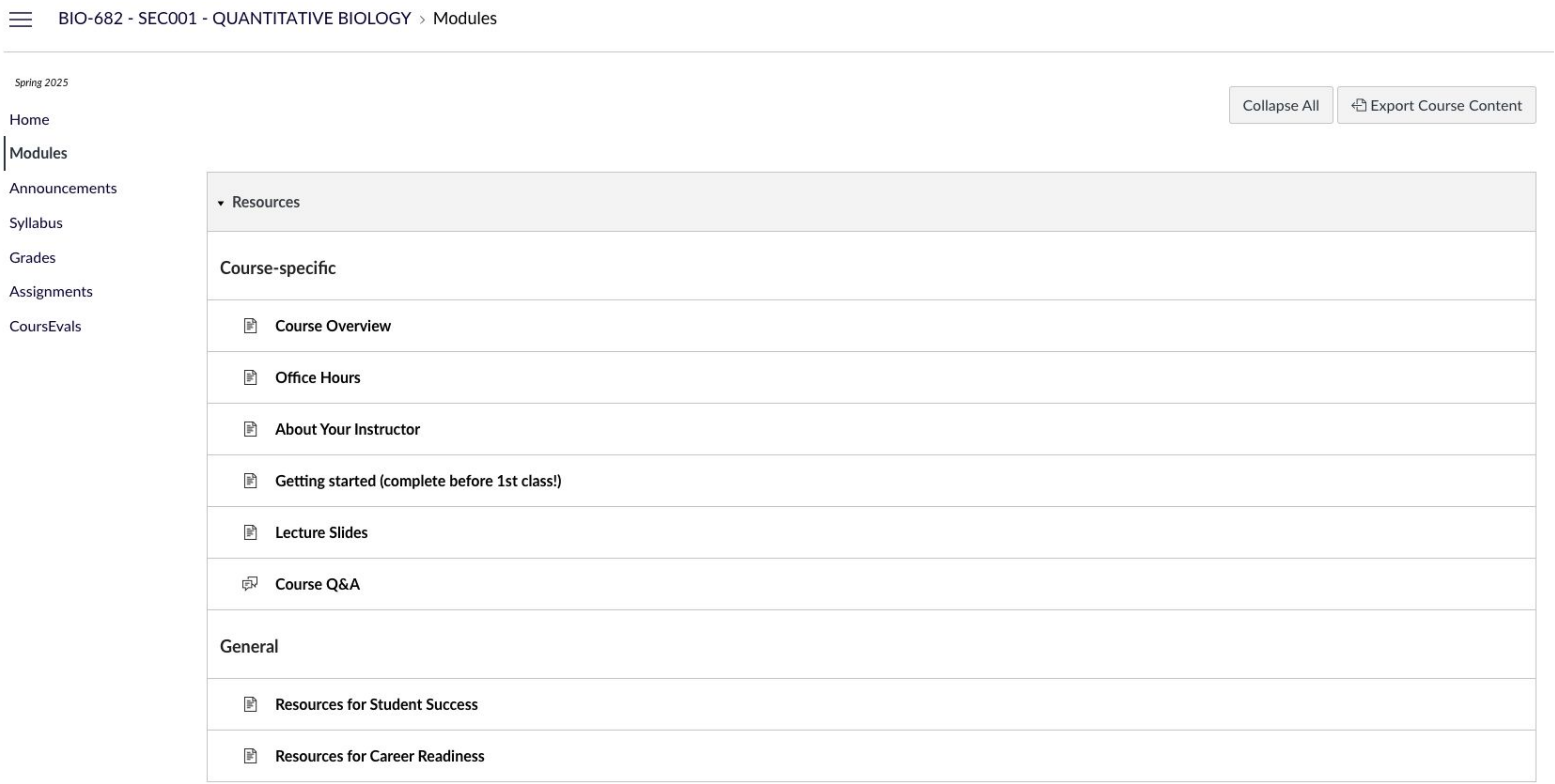
(Last 4 weeks)

- Individual coding projects
- Topic of your choice
- 2 work weeks
- 2 weeks for presentations

Assignments

- One assignment per week (weeks 1-11)
- Focus on hands-on time in class (may need to complete outside of class)
- Always due by 11:59 pm on Thursday
- Can submit 1 revision per assignment

Modules page in Canvas



BIO-682 - SEC001 - QUANTITATIVE BIOLOGY › Modules

Spring 2025

- Home
- Modules
- Announcements
- Syllabus
- Grades
- Assignments
- CoursEvals

Collapse All | Export Course Content

▾ Resources

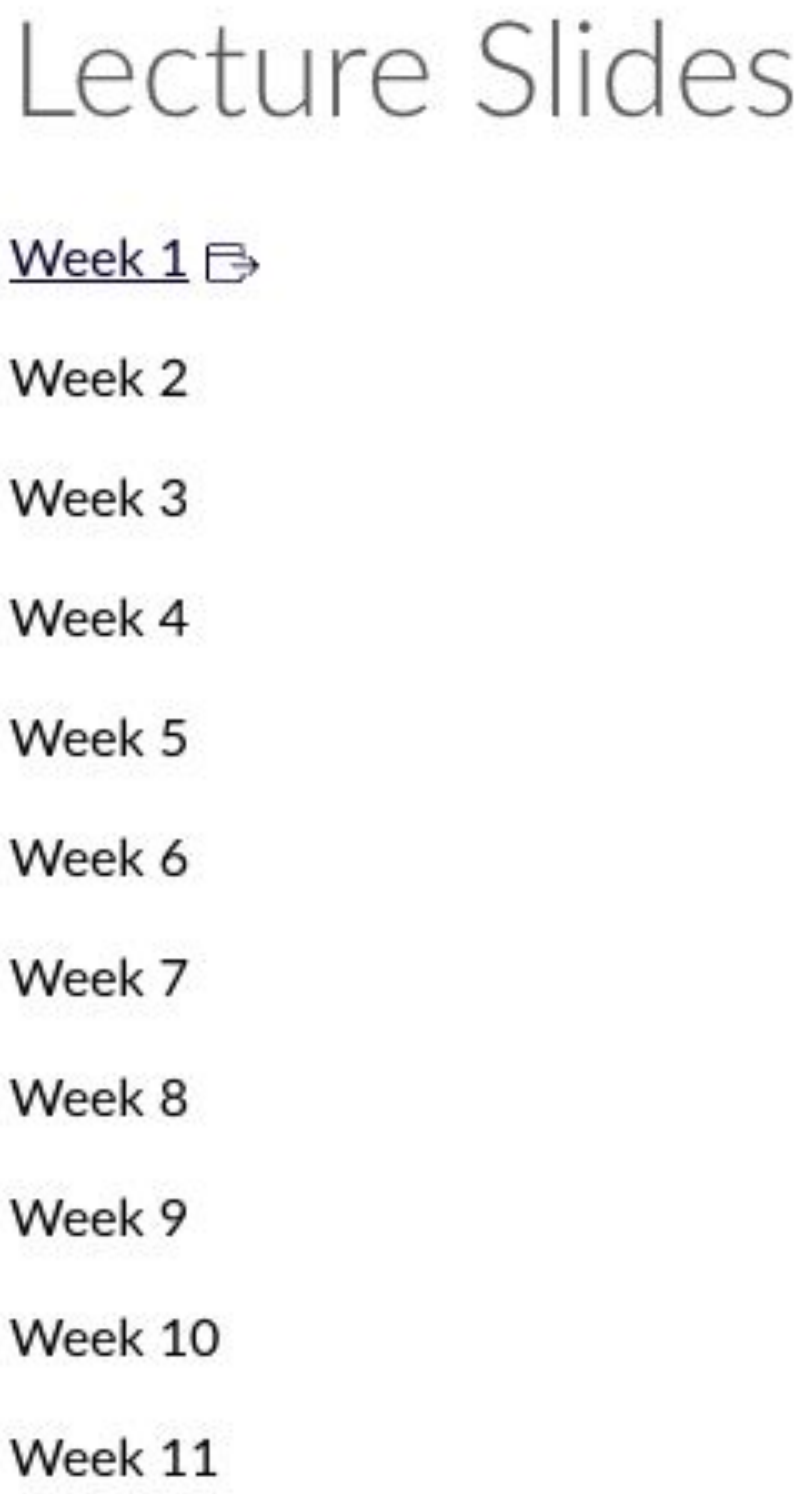
Course-specific

- Course Overview
- Office Hours
- About Your Instructor
- Getting started (complete before 1st class!)
- Lecture Slides
- Course Q&A

General

- Resources for Student Success
- Resources for Career Readiness

Links to Lecture Slides



Lecture Slides

Week 1

Week 2

Week 3

Week 4

Week 5

Week 6

Week 7

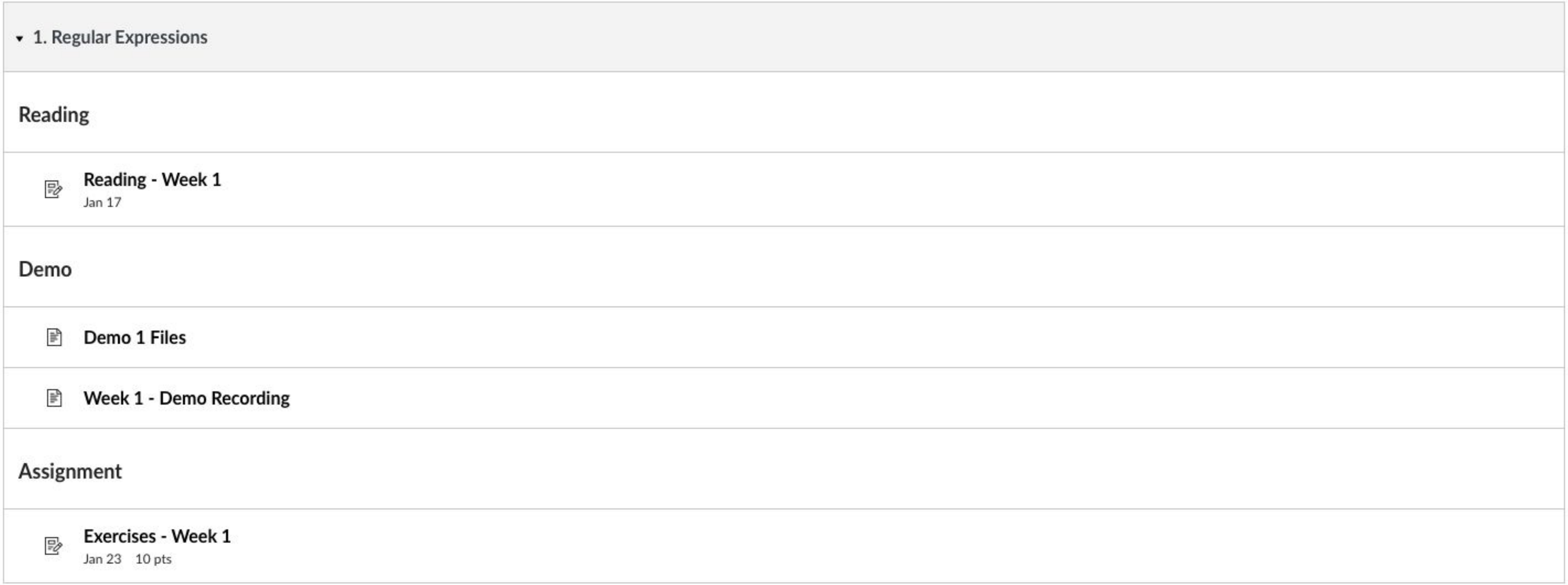
Week 8

Week 9

Week 10

Week 11

Access assignments via Modules



▾ 1. Regular Expressions

Reading

Reading - Week 1
Jan 17

Demo

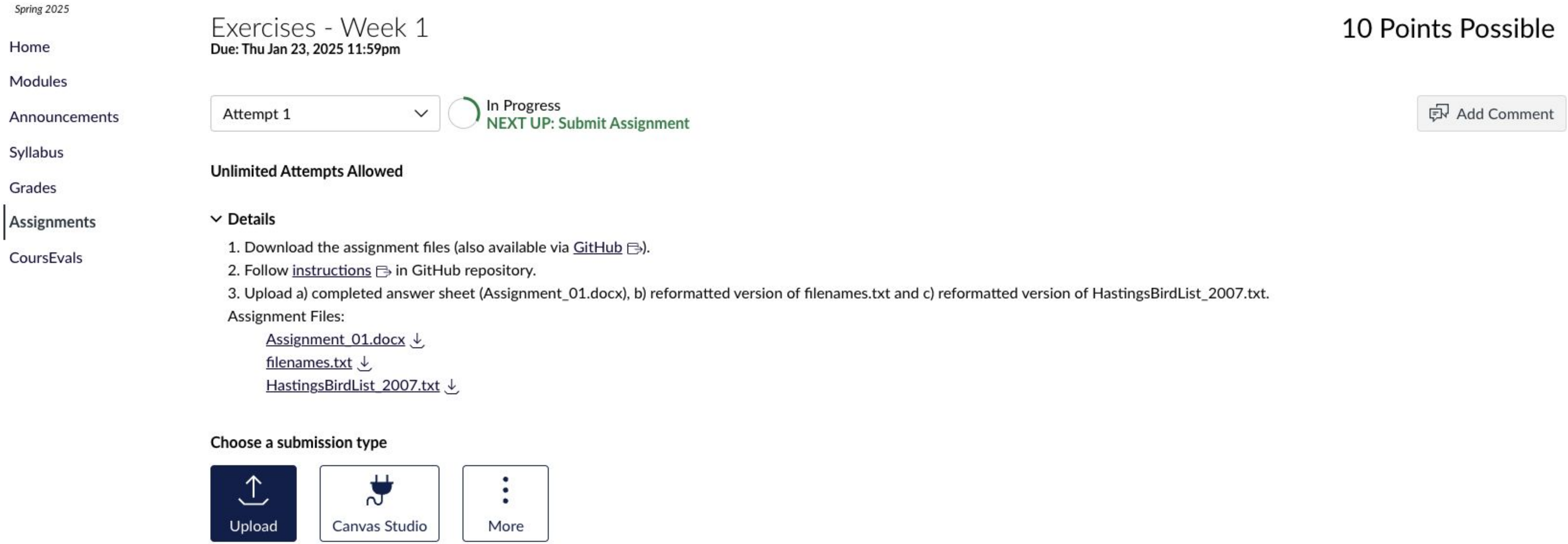
Demo 1 Files

Week 1 - Demo Recording

Assignment

Exercises - Week 1
Jan 23 10 pts

Assignments submitted via Canvas



Exercises - Week 1

Due: Thu Jan 23, 2025 11:59pm

10 Points Possible

In Progress

NEXT UP: Submit Assignment

Unlimited Attempts Allowed

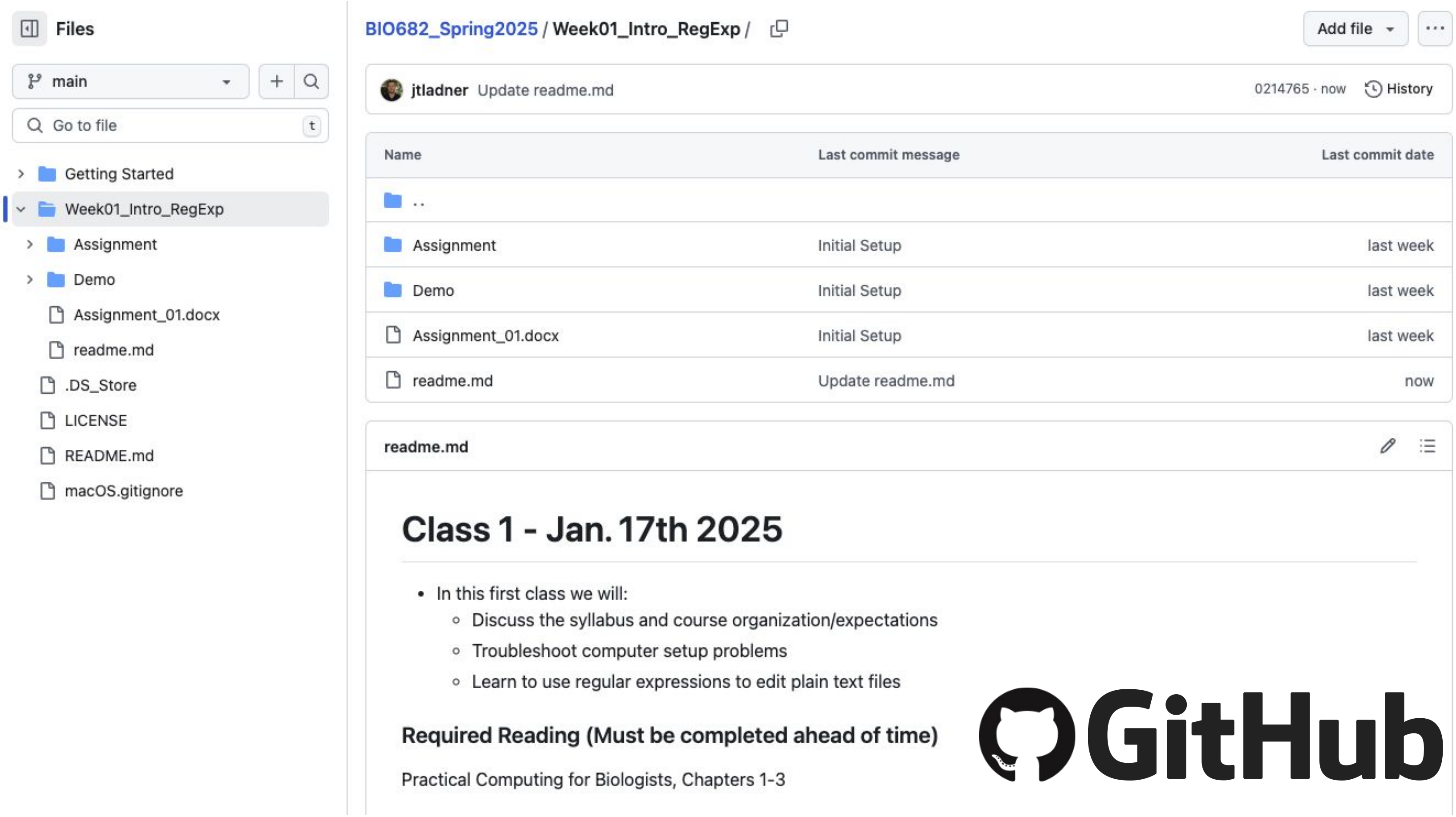
Details

1. Download the assignment files (also available via GitHub).
2. Follow instructions in GitHub repository.
3. Upload a) completed answer sheet (Assignment_01.docx), b) reformatted version of filenames.txt and c) reformatted version of HastingsBirdList_2007.txt.

Assignment Files:

- Assignment_01.docx
- filenames.txt
- HastingsBirdList_2007.txt

Choose a submission type



jtladner Update readme.md 0214765 · now

Name	Last commit message	Last commit date
..		
Assignment	Initial Setup	last week
Demo	Initial Setup	last week
Assignment_01.docx	Initial Setup	last week
readme.md	Update readme.md	now

readme.md

Class 1 - Jan. 17th 2025

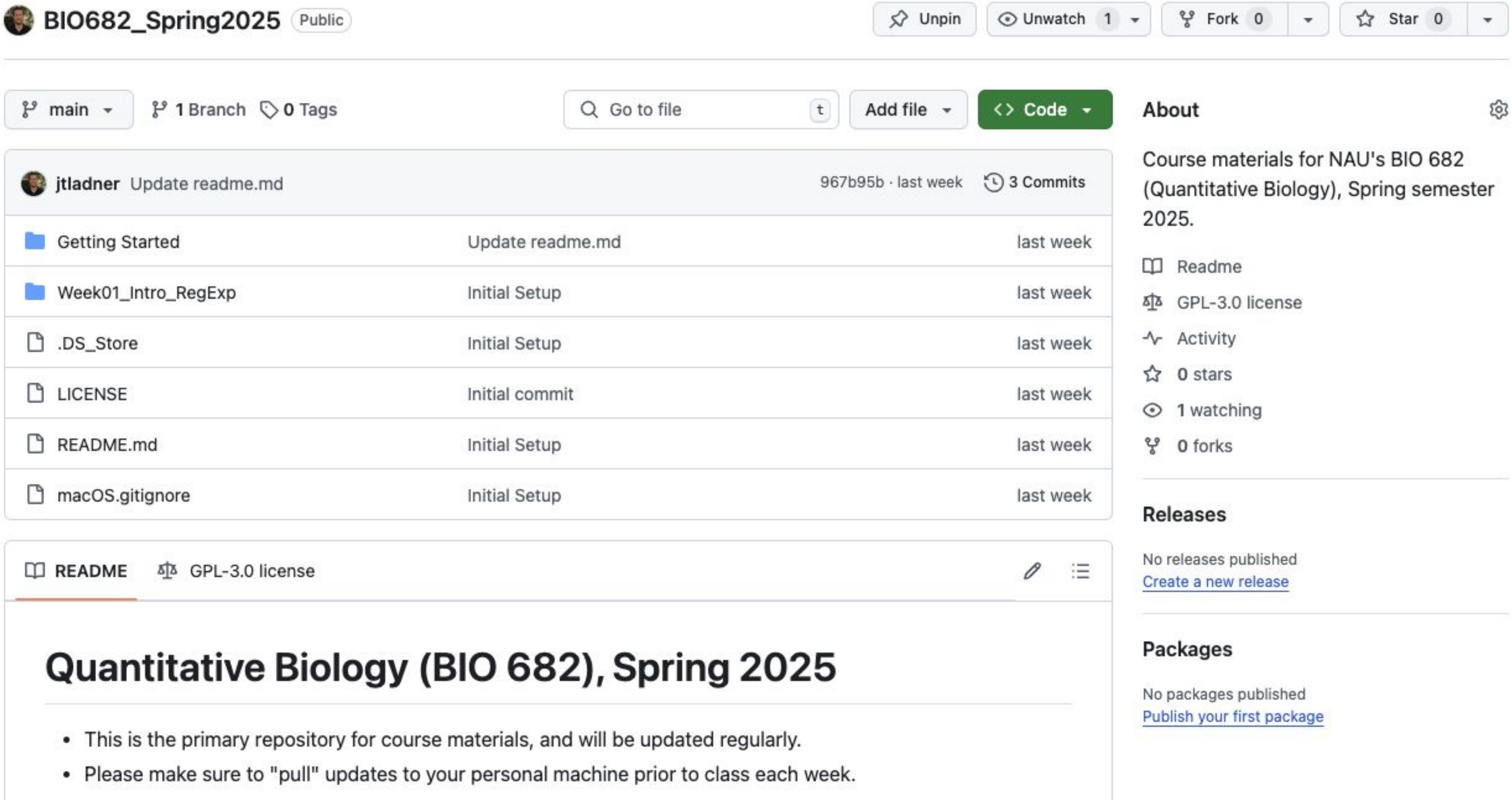
- In this first class we will:
  - Discuss the syllabus and course organization/expectations
  - Troubleshoot computer setup problems
  - Learn to use regular expressions to edit plain text files

Required Reading (Must be completed ahead of time)

Practical Computing for Biologists, Chapters 1-3

GitHub

Materials will be continually added to repository



jtladner Update readme.md 967b95b · last week 3 Commits

Getting Started	Update readme.md	last week
Week01_Intro_RegExp	Initial Setup	last week
.DS_Store	Initial Setup	last week
LICENSE	Initial commit	last week
README.md	Initial Setup	last week
macOS.gitignore	Initial Setup	last week

Quantitative Biology (BIO 682), Spring 2025

- This is the primary repository for course materials, and will be updated regularly.
- Please make sure to "pull" updates to your personal machine prior to class each week.

About

Course materials for NAU's BIO 682 (Quantitative Biology), Spring semester 2025.

Releases

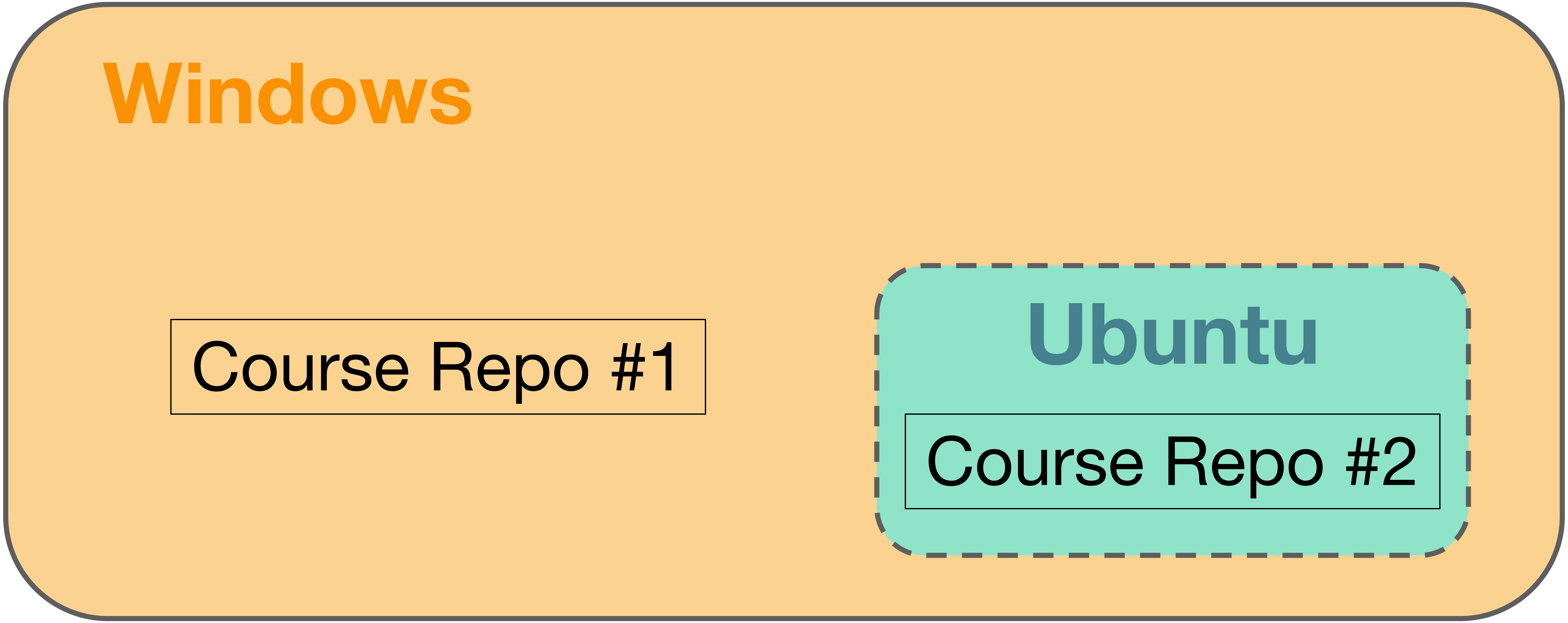
No releases published
Create a new release

Packages

No packages published
Publish your first package

“Pulling” GitHub updates

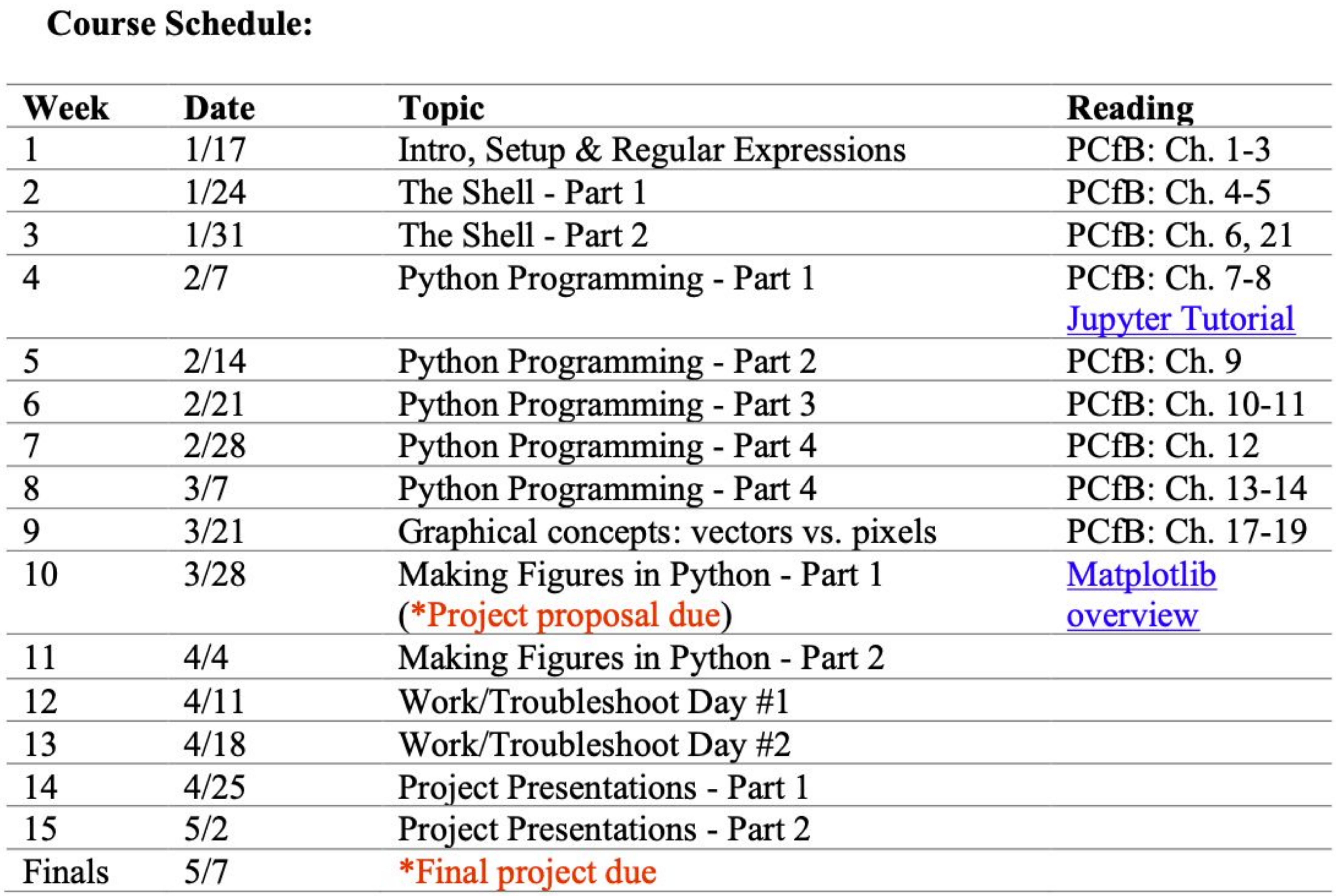
Windows users: maintain 2 copies of the repository



Grading

- Assignments (33%)
- Attendance/Participation (33%)
- Final Project/Presentation (34%)

Final project - deadlines



Course Schedule:

Week	Date	Topic	Reading
1	1/17	Intro, Setup & Regular Expressions	PCfB: Ch. 1-3
2	1/24	The Shell - Part 1	PCfB: Ch. 4-5
3	1/31	The Shell - Part 2	PCfB: Ch. 6, 21
4	2/7	Python Programming - Part 1	PCfB: Ch. 7-8 Jupyter Tutorial
5	2/14	Python Programming - Part 2	PCfB: Ch. 9
6	2/21	Python Programming - Part 3	PCfB: Ch. 10-11
7	2/28	Python Programming - Part 4	PCfB: Ch. 12
8	3/7	Python Programming - Part 4	PCfB: Ch. 13-14
9	3/21	Graphical concepts: vectors vs. pixels	PCfB: Ch. 17-19
10	3/28	Making Figures in Python - Part 1 (*Project proposal due)	Matplotlib overview
11	4/4	Making Figures in Python - Part 2	
12	4/11	Work/Troubleshoot Day #1	
13	4/18	Work/Troubleshoot Day #2	
14	4/25	Project Presentations - Part 1	
15	5/2	Project Presentations - Part 2	
Finals	5/7	*Final project due	

Computer setup

- Text Editor
- Command line terminal
- GitHub Repository

https://github.com/jtladner/BIO682_Spring2025/tree/main/Getting%20Started

Plain text files

Plain text file

- Pure sequence of character codes
- No formatting (e.g., text size, color, font, spacing)
- Human and machine readable
- Standardized

Which of these formats are NOT plain text?

Excel (.xlsx)

html

OpenOffice (.odf)

Google Sheet

text (.txt)

markdown

fasta

nexus

xml

Google Doc

json

Word (.doc)

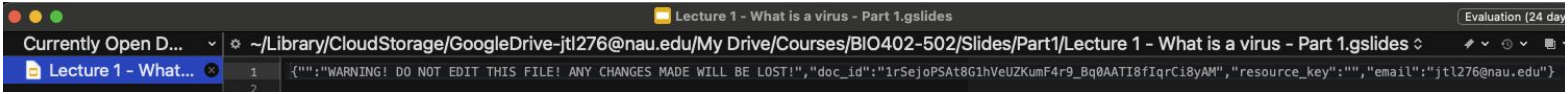
rich text (.rtf)

python script (.py)

tab-separated (.tsv)

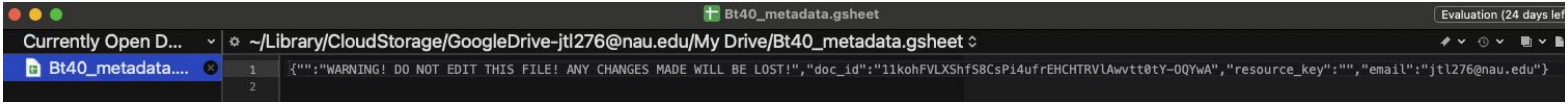
Viewing non-plain text in text editor

Google Slides


Lecture 1 - What is a virus - Part 1.gslides
Evaluation (24 day
Currently Open D...
~/Library/CloudStorage/GoogleDrive-jtl276@nau.edu/My Drive/Courses/BIO402-502/Slides/Part1/Lecture 1 - What is a virus - Part 1.gslides
Lecture 1 - What...

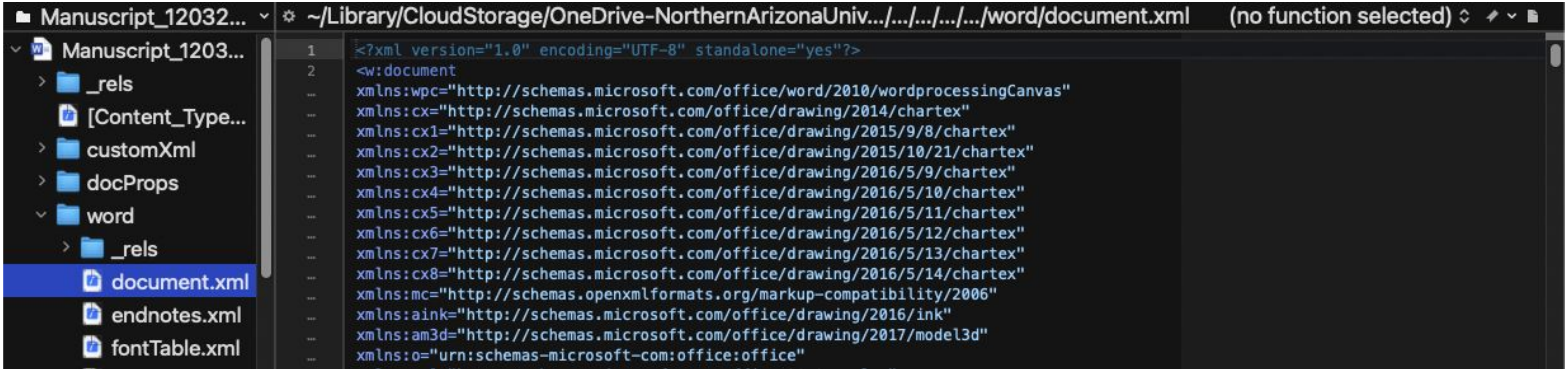
```
{"":"WARNING! DO NOT EDIT THIS FILE! ANY CHANGES MADE WILL BE LOST!","doc_id":"1rSejoPSAt8G1hVeUZKumF4r9_Bq0AATI8fIqrCi8yAM","resource_key":"","email":"jtl276@nau.edu"}
```

Google Sheet


Bt40_metadata.gsheet
Evaluation (24 days lef
Currently Open D...
~/Library/CloudStorage/GoogleDrive-jtl276@nau.edu/My Drive/Bt40_metadata.gsheet
Bt40_metadata....

```
{"":"WARNING! DO NOT EDIT THIS FILE! ANY CHANGES MADE WILL BE LOST!","doc_id":"11kohFVLXShfS8CsPi4ufrEHCHTRVlAwvtt0tY-0QYwA","resource_key":"","email":"jtl276@nau.edu"}
```

Word Doc

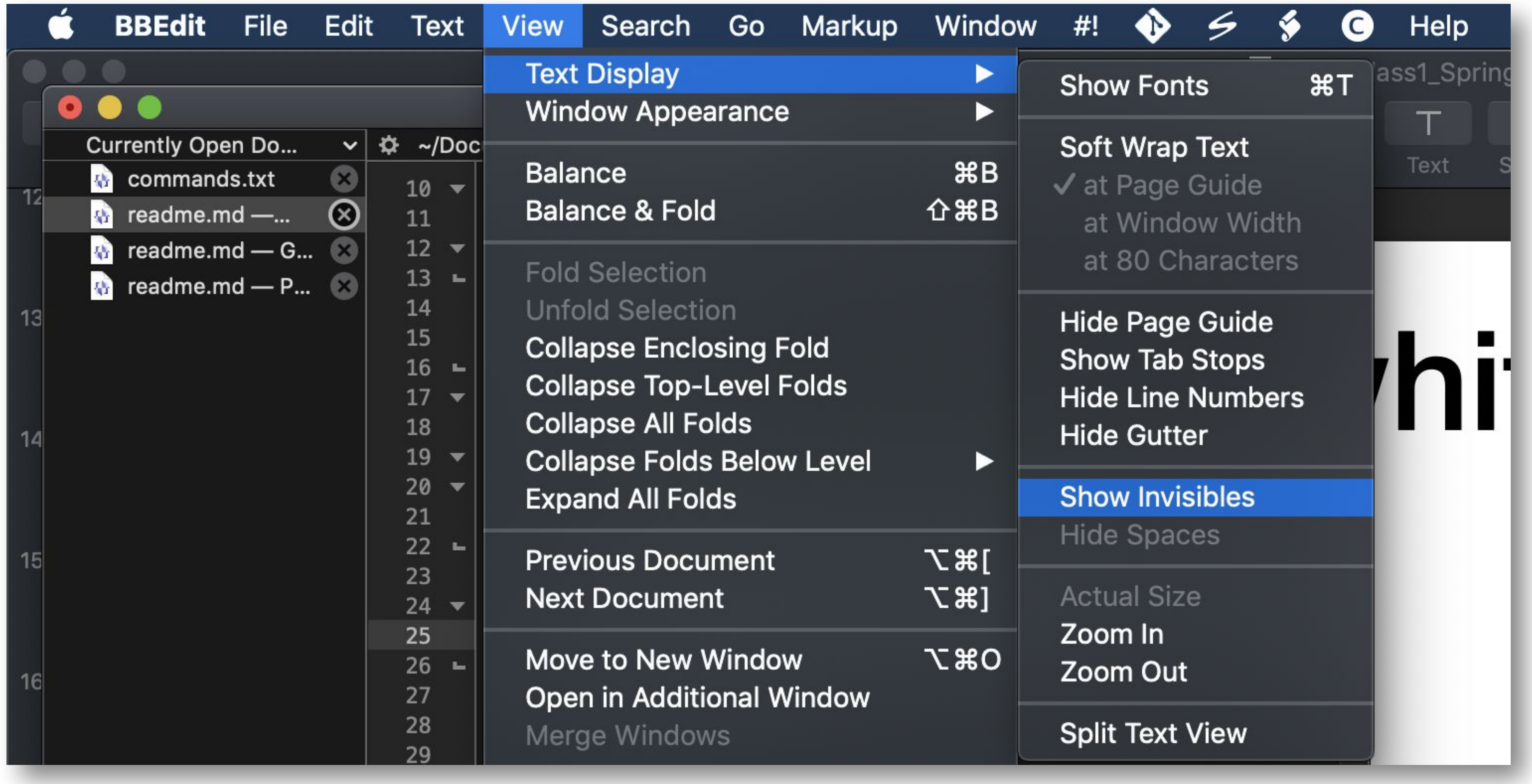


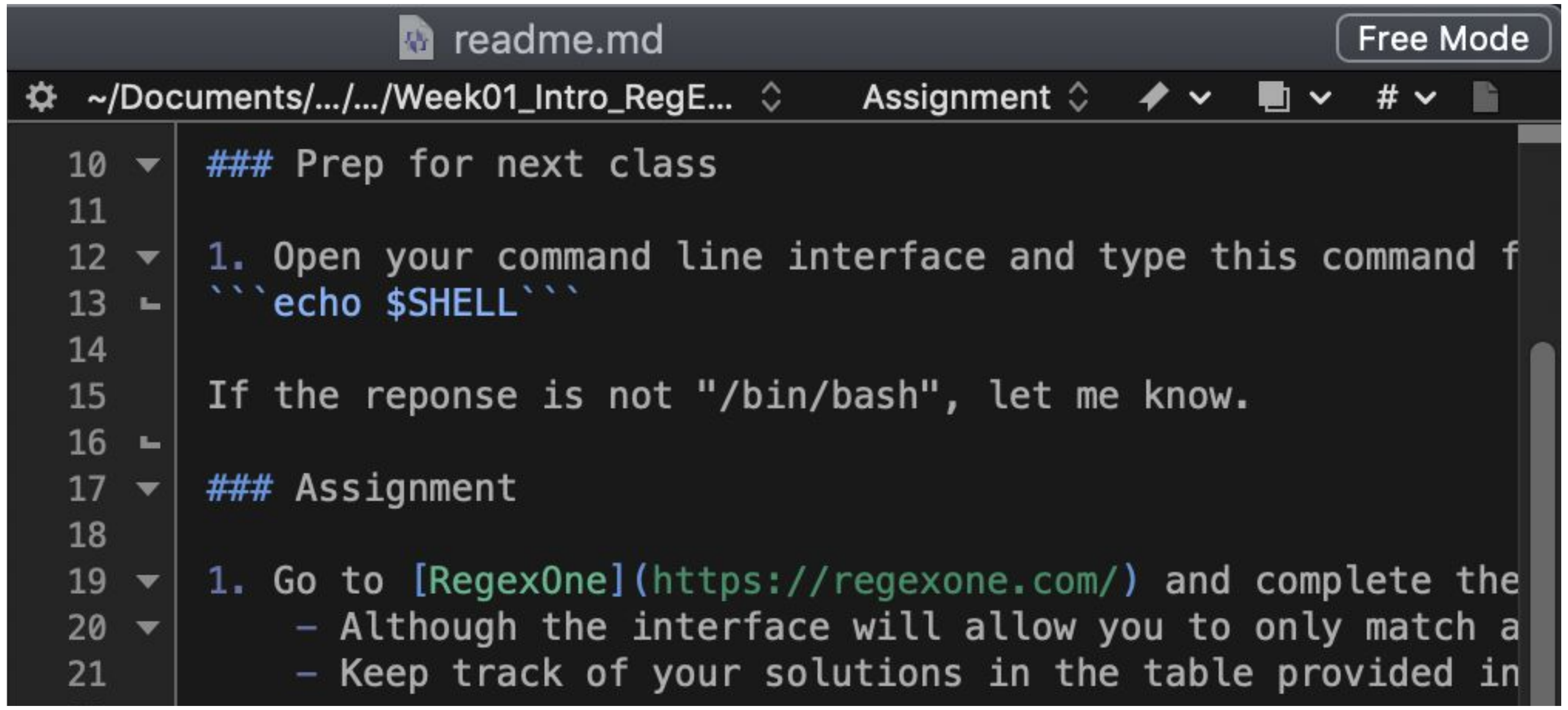
```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<w:document
xmlns:wpc="http://schemas.microsoft.com/office/word/2010/wordprocessingCanvas"
xmlns:cx="http://schemas.microsoft.com/office/drawing/2014/chartex"
xmlns:cx1="http://schemas.microsoft.com/office/drawing/2015/9/8/chartex"
xmlns:cx2="http://schemas.microsoft.com/office/drawing/2015/10/21/chartex"
xmlns:cx3="http://schemas.microsoft.com/office/drawing/2016/5/9/chartex"
xmlns:cx4="http://schemas.microsoft.com/office/drawing/2016/5/10/chartex"
xmlns:cx5="http://schemas.microsoft.com/office/drawing/2016/5/11/chartex"
xmlns:cx6="http://schemas.microsoft.com/office/drawing/2016/5/12/chartex"
xmlns:cx7="http://schemas.microsoft.com/office/drawing/2016/5/13/chartex"
xmlns:cx8="http://schemas.microsoft.com/office/drawing/2016/5/14/chartex"
xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
xmlns:aink="http://schemas.microsoft.com/office/drawing/2016/ink"
xmlns:am3d="http://schemas.microsoft.com/office/drawing/2017/model3d"
xmlns:o="urn:schemas-microsoft-com:office:office"
```

Whitespace

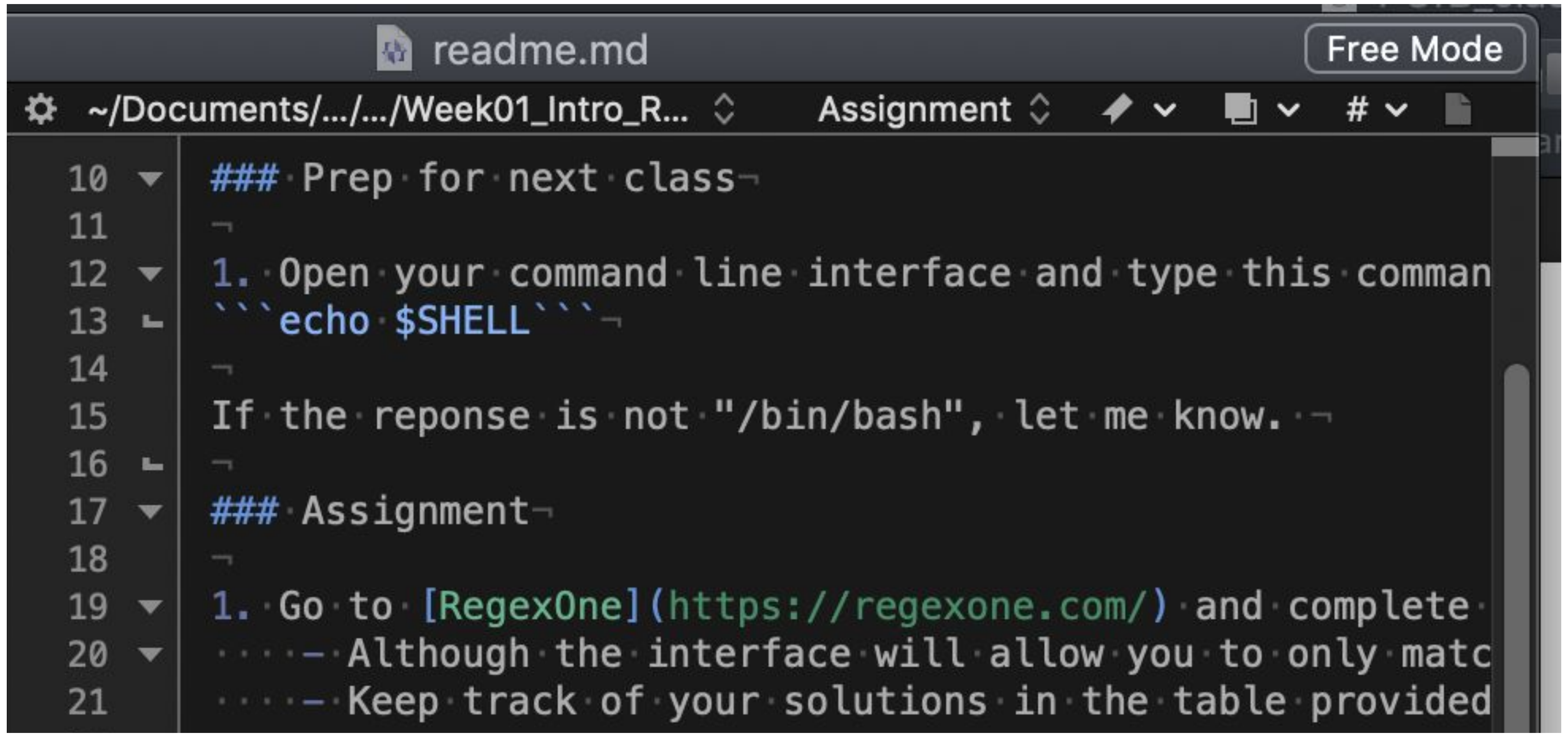
- Space
- Tab
- End of line

Visualizing white space (BBEdit)



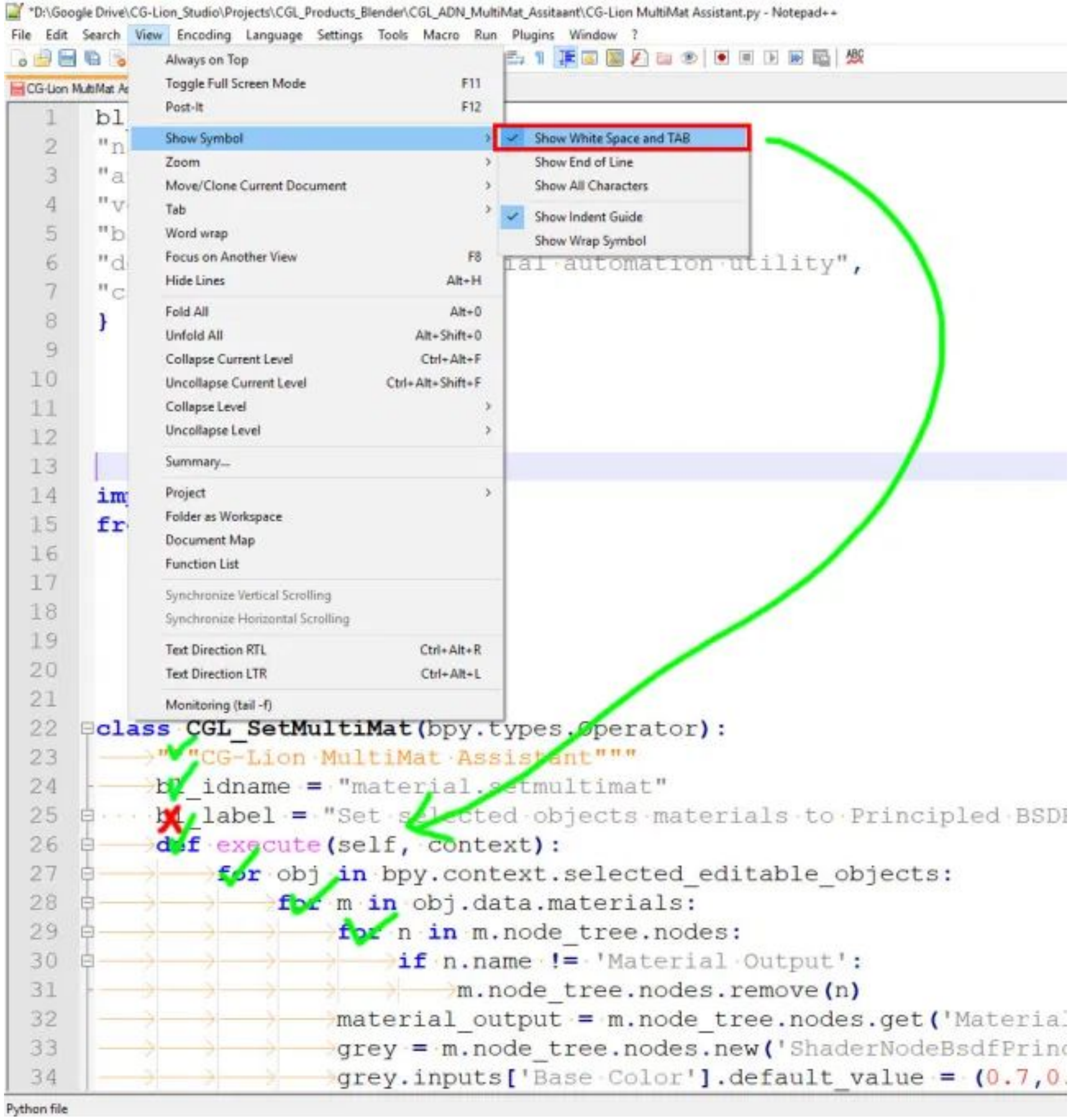


~~~
### Prep for next class

1. Open your command line interface and type this command f
```echo $SHELL```

If the reponse is not "/bin/bash", let me know.

### Assignment

1. Go to [RegexOne](https://regexone.com/) and complete the
    - Although the interface will allow you to only match a
    - Keep track of your solutions in the table provided in
~~~



~~~
###·Prep·for·next·class¬
¬
1.·Open·your·command·line·interface·and·type·this·comman
```echo·$SHELL```¬
¬
If·the·reponse·is·not·"/bin/bash",·let·me·know.·¬
¬
###·Assignment¬
¬
1.·Go·to·[RegexOne](https://regexone.com/)·and·complete·
····-·Although·the·interface·will·allow·you·to·only·matc
····-·Keep·track·of·your·solutions·in·the·table·provided
~~~

Visualizing white space (Notepad++)

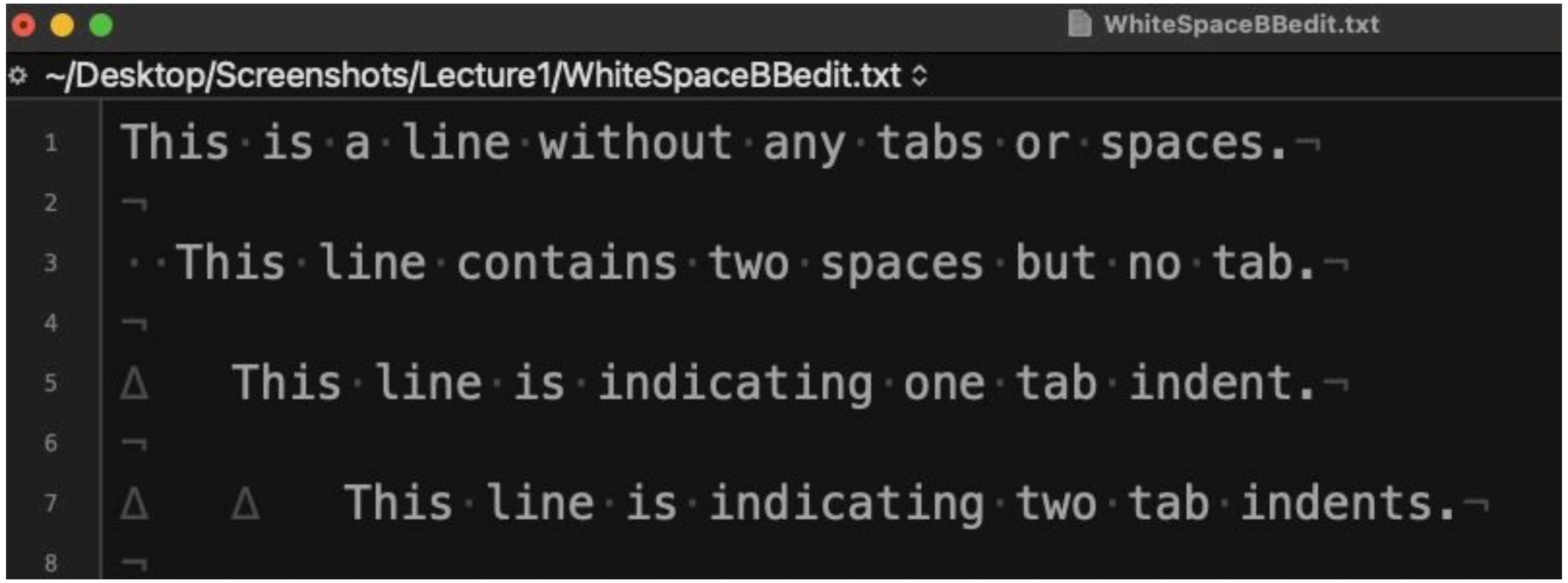


*D:\Google Drive\CG-Lion_Studio\Projects\CGL_Products_Blender\CGL_ADN_MultiMat_Assitaant\CG-Lion MultiMat Assistant.py - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
CG-Lion MultiMat As
Always on Top
Toggle Full Screen Mode F11
Post-It F12
Show Symbol
Zoom
Move/Clone Current Document
Tab
Word wrap
Focus on Another View F8
Hide Lines Alt+H
Fold All Alt+0
Unfold All Alt+Shift+0
Collapse Current Level Ctrl+Alt+F
Uncollapse Current Level Ctrl+Alt+Shift+F
Collapse Level
Uncollapse Level
Summary...
Project
Folder as Workspace
Document Map
Function List
Synchronize Vertical Scrolling
Synchronize Horizontal Scrolling
Text Direction RTL Ctrl+Alt+R
Text Direction LTR Ctrl+Alt+L
Monitoring (tail -f)
✓ Show White Space and TAB
Show End of Line
Show All Characters
✓ Show Indent Guide
Show Wrap Symbol
Python file

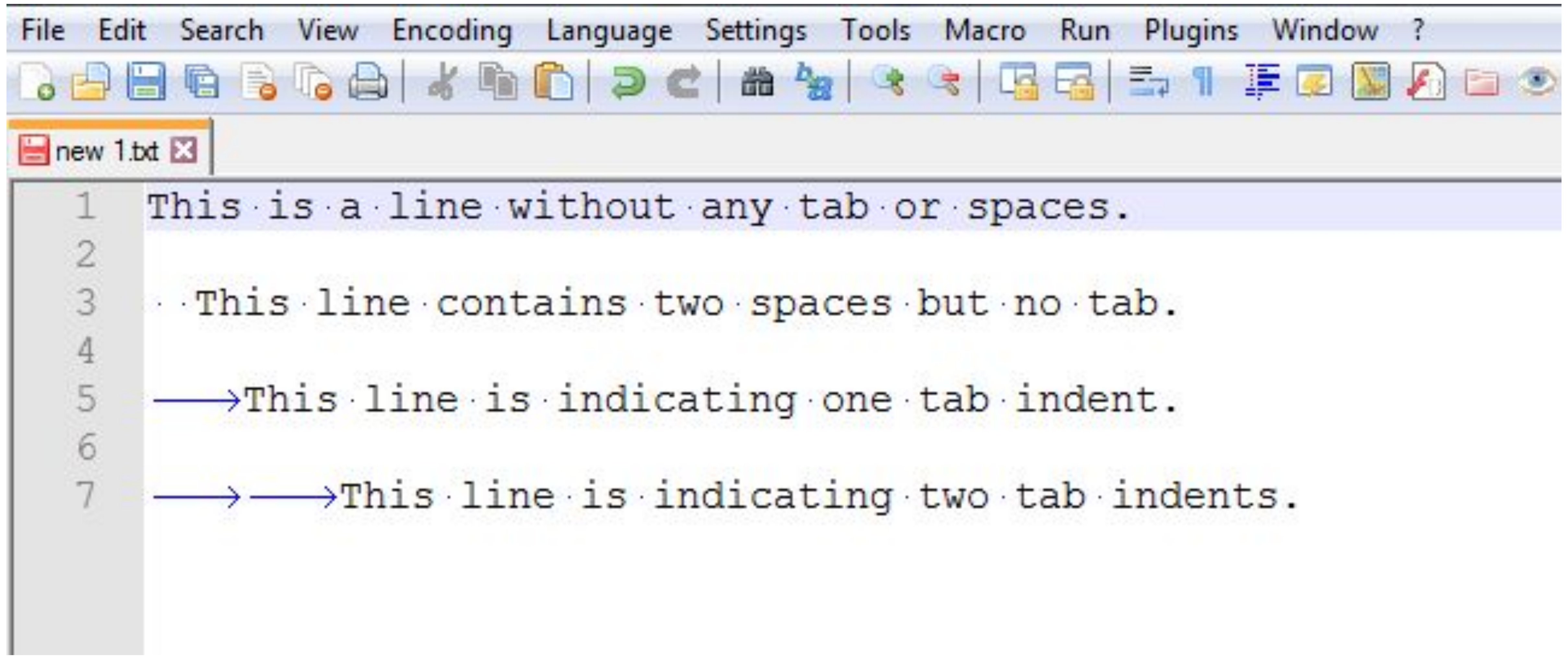
```
bl
"n
"a
"v
"b
"d                         ial automation utility",
"c
}



im
fr







class CGL_SetMultiMat(bpy.types.Operator):
	"""CG-Lion MultiMat Assistant"""
	bl_idname = "material.setmultimat"
    bl_label = "Set selected objects materials to Principled BSDI
	def execute(self, context):
		for obj in bpy.context.selected_editable_objects:
			for m in obj.data.materials:
				for n in m.node_tree.nodes:
					if n.name != 'Material Output':
						m.node_tree.nodes.remove(n)
				material_output = m.node_tree.nodes.get('Materia
				grey = m.node_tree.nodes.new('ShaderNodeBsdfPrinc
				grey.inputs['Base Color'].default_value = (0.7,0.
```

BBEdit

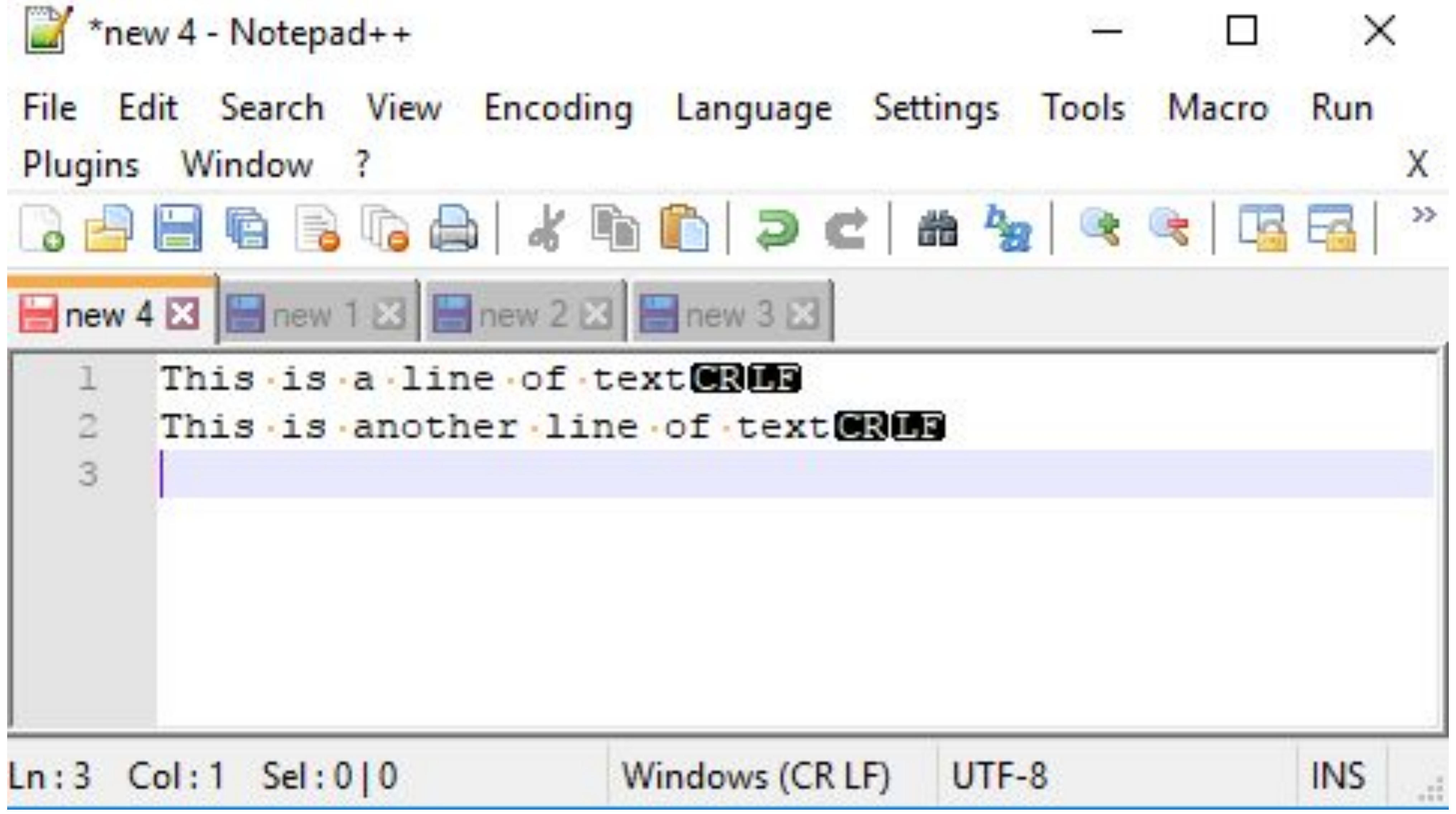
Notepad++



```
This·is·a·line·without·any·tabs·or·spaces.¬
¬
··This·line·contains·two·spaces·but·no·tab.¬
¬
Δ   This·line·is·indicating·one·tab·indent.¬
¬
Δ   Δ   This·line·is·indicating·two·tab·indents.¬
¬
```



```
This·is·a·line·without·any·tab·or·spaces.

··This·line·contains·two·spaces·but·no·tab.

→This·line·is·indicating·one·tab·indent.

→→This·line·is·indicating·two·tab·indents.
```



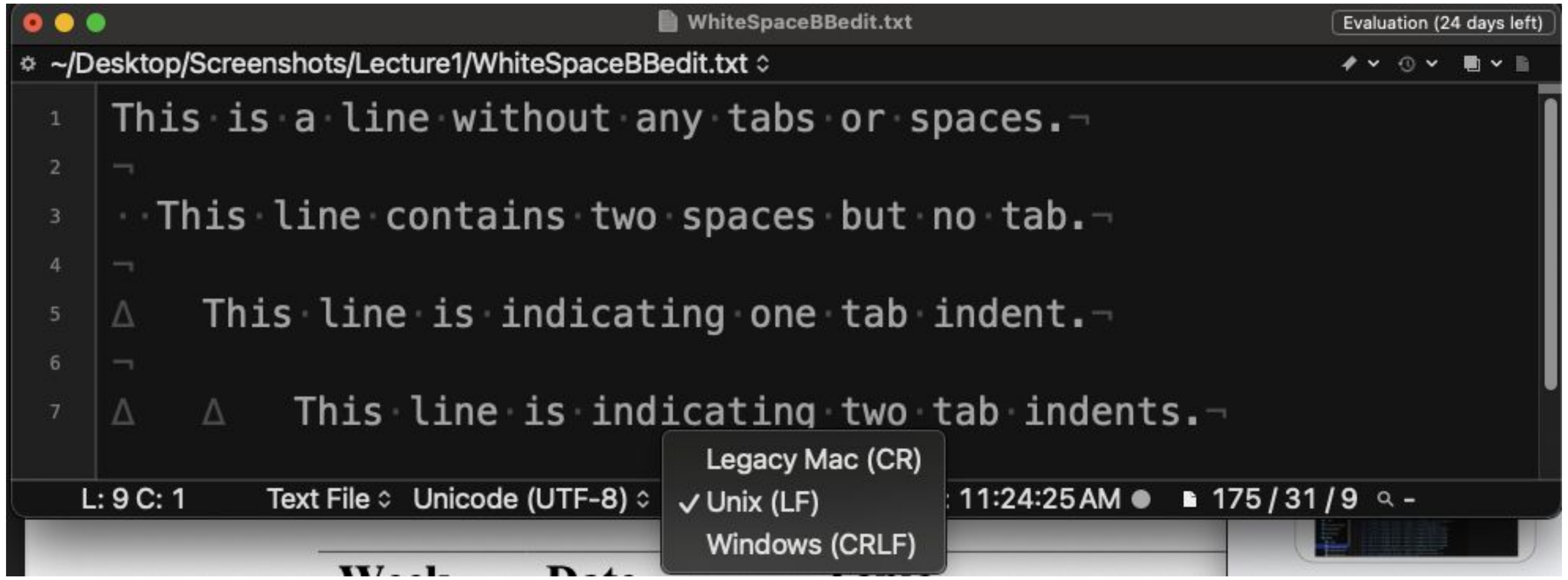
```
This·is·a·line·of·textCRLF
This·is·another·line·of·textCRLF

```

Ln : 3 Col : 1 Sel : 0 | 0 Windows (CR LF) UTF-8 INS

End of line characters differ by OS

- Line feed (LF) - Mac OSX, Linux
- Carriage return (CR) - Mac OS9 and earlier
- Carriage return + line feed (CRLF) - Windows

BBEdit



WhiteSpaceBBedit.txt

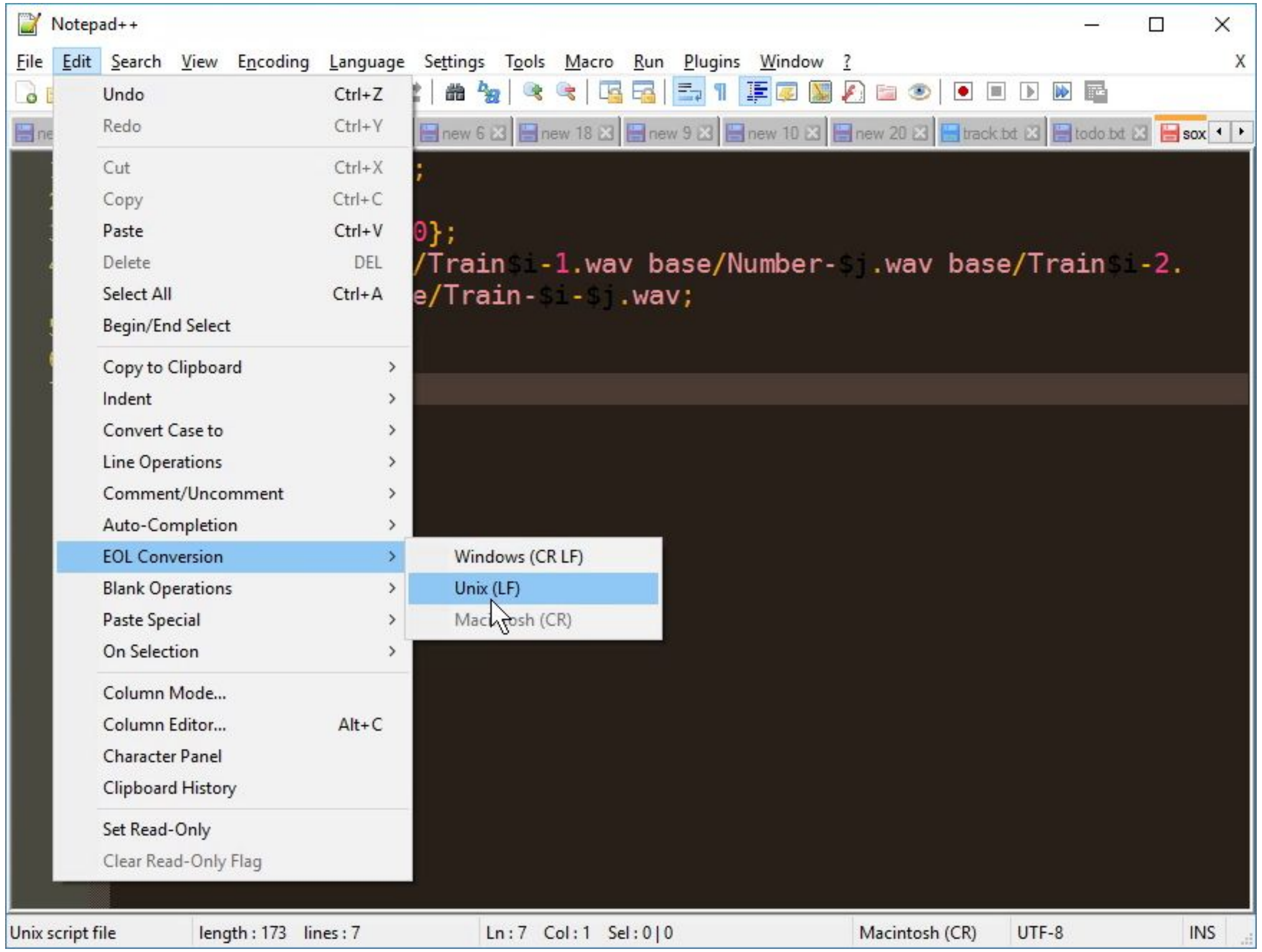
Evaluation (24 days left)

~/Desktop/Screenshots/Lecture1/WhiteSpaceBBedit.txt

```
This·is·a·line·without·any·tabs·or·spaces.¬
¬
··This·line·contains·two·spaces·but·no·tab.¬
¬
Δ   This·line·is·indicating·one·tab·indent.¬
¬
Δ   Δ   This·line·is·indicating·two·tab·indents.¬
```

L: 9 C: 1   Text File   Unicode (UTF-8)   11:24:25 AM   175 / 31 / 9

Notepad++



```
;

0};
/Train$i-1.wav base/Number-$j.wav base/Train$i-2.
e/Train-$i-$j.wav;
```

Unix script file   length : 173   lines : 7   Ln : 7   Col : 1   Sel : 0 | 0   Macintosh (CR)   UTF-8   INS

Regular expressions

Regular expressions

(a.k.a. regex, regexp)

- Powerful find and replace toolkit
- Understood by many text editors, programming languages and even search engines
- Power comes from wildcard operators

```
\d

\w

\s

 .
```

```
\w+

\w*

\w?
```

```
[ABC]

[^ABC]

[A-C]
```

(ABC)

(AB)C

((AB)C)

Anchors

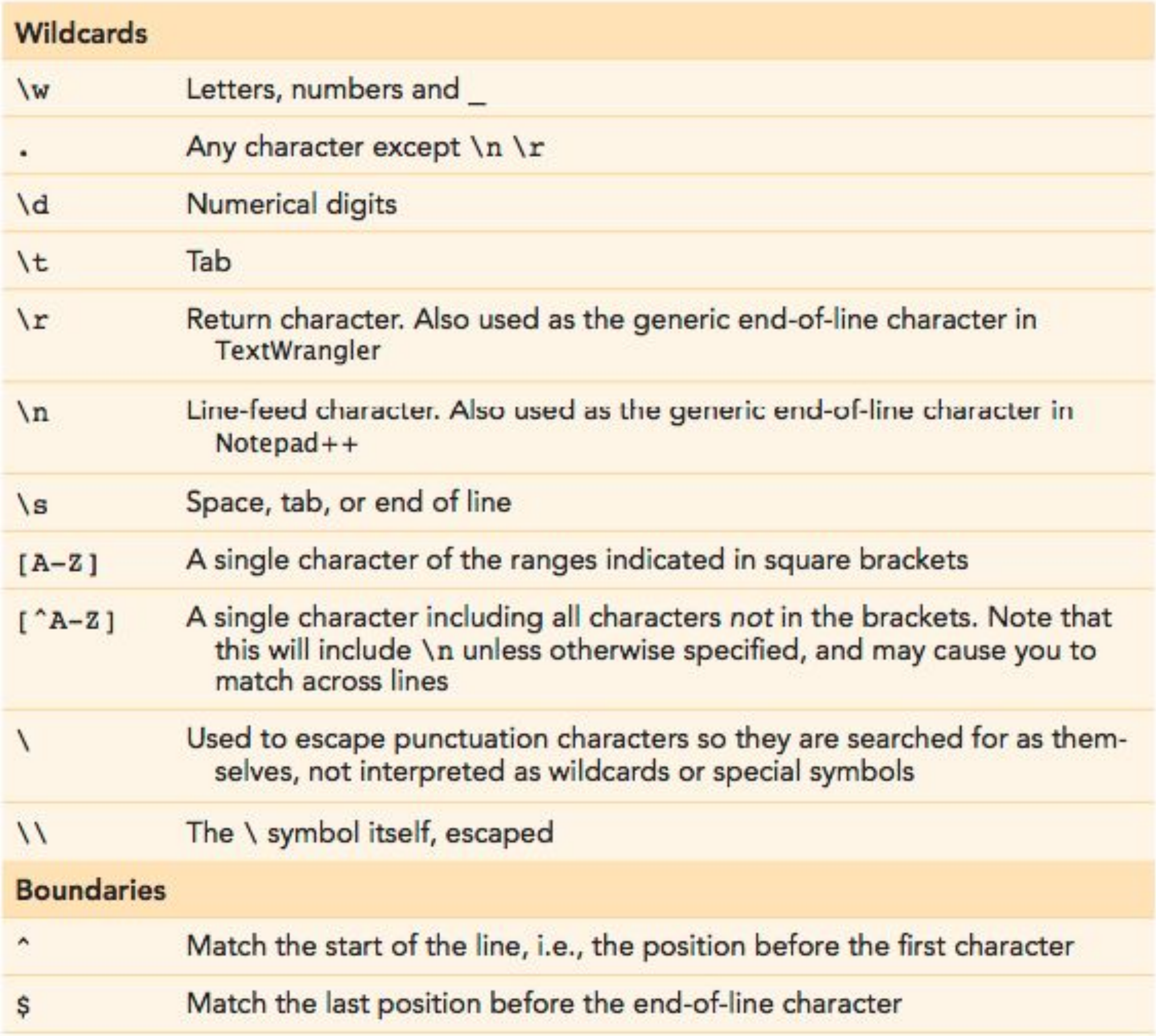
^

$

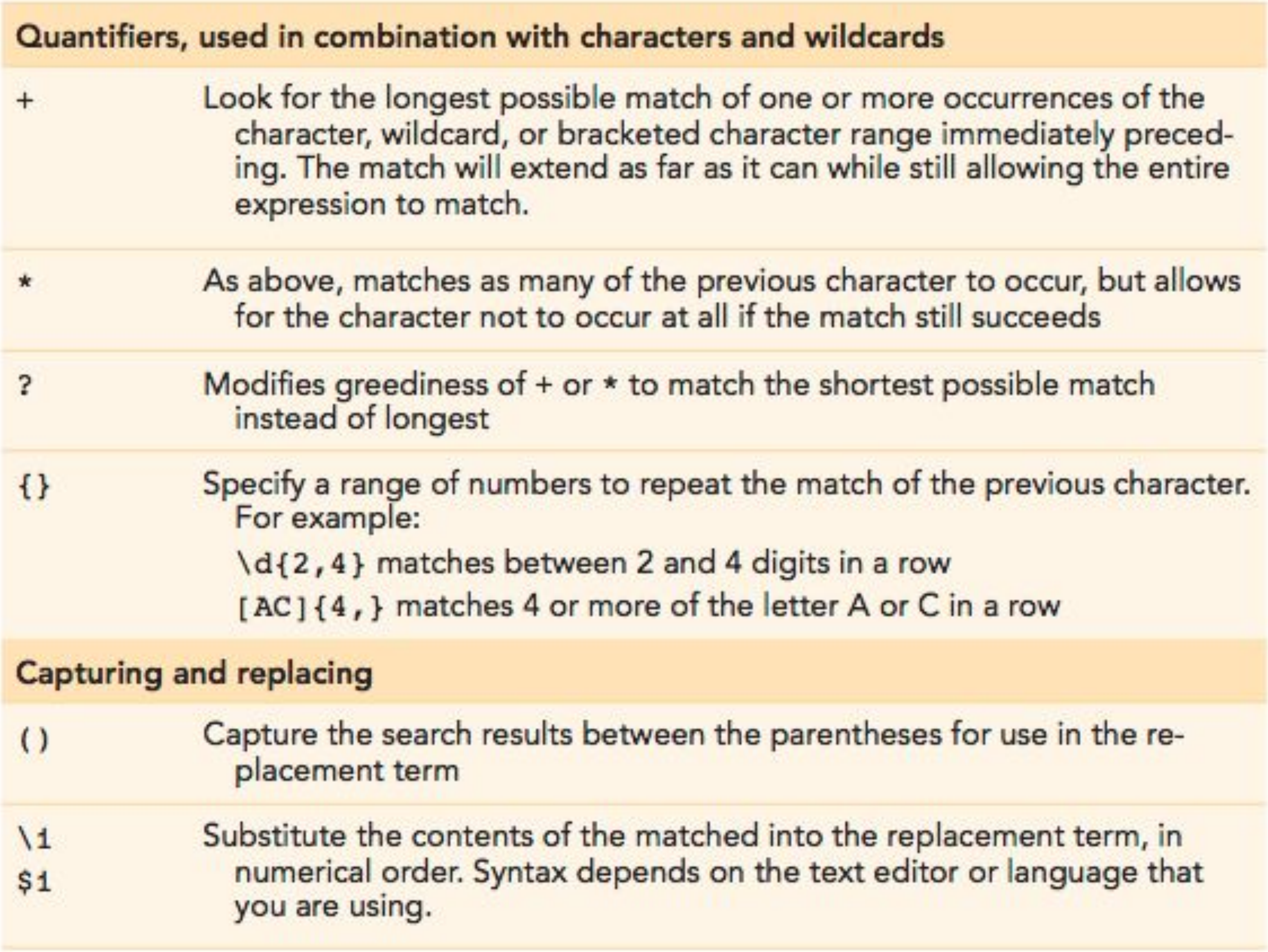
Tips

- Try PCfB methodology
  - copy (1 version of) target text into search dialog
  - replace text with wildcards, piece by piece
- Be as specific as possible
- Build in redundancies

Regexp reference tables



Wildcards	
\w	Letters, numbers and _
.	Any character except \n \r
\d	Numerical digits
\t	Tab
\r	Return character. Also used as the generic end-of-line character in TextWrangler
\n	Line-feed character. Also used as the generic end-of-line character in Notepad++
\s	Space, tab, or end of line
[A-Z]	A single character of the ranges indicated in square brackets
[^A-Z]	A single character including all characters *not* in the brackets. Note that this will include \n unless otherwise specified, and may cause you to match across lines
\	Used to escape punctuation characters so they are searched for as themselves, not interpreted as wildcards or special symbols
\\	The \ symbol itself, escaped
Boundaries	
^	Match the start of the line, i.e., the position before the first character
$	Match the last position before the end-of-line character



Quantifiers, used in combination with characters and wildcards	
+	Look for the longest possible match of one or more occurrences of the character, wildcard, or bracketed character range immediately preceding. The match will extend as far as it can while still allowing the entire expression to match.
*	As above, matches as many of the previous character to occur, but allows for the character not to occur at all if the match still succeeds
?	Modifies greediness of + or * to match the shortest possible match instead of longest
{}	Specify a range of numbers to repeat the match of the previous character. For example: \d{2,4} matches between 2 and 4 digits in a row [AC]{4,} matches 4 or more of the letter A or C in a row
Capturing and replacing	
()	Capture the search results between the parentheses for use in the replacement term
\1 $1	Substitute the contents of the matched into the replacement term, in numerical order. Syntax depends on the text editor or language that you are using.

http://practicalcomputing.org/files/PCfB_Appendices.pdf