# Python Text Basics

# Natural Language Processing Bootcamp

- Section Goals
  - Understand how to open normal .txt and .pdf files with basic Python libraries
  - Learn some basic regular expressions
  - Test skills with an assessment exercise.

# Working with Text Files

PART ONE

# Natural Language Processing Bootcamp

- Let’s go over some basic print formatting (f-string literal).
- We’ll also discuss alignment options with f-string literals.
- Let’s get started!

# Working with Text Files

PART TWO

# Natural Language Processing Bootcamp

- Let’s go over how to read and write to text files with Python!

# Working with PDF Files

# Natural Language Processing Bootcamp

- Often you may need to read in text data from a PDF file.
- We can use the PyPDF library to read in text data from a PDF file.
- **Keep in mind: NOT ALL PDFS HAVE TEXT THAT CAN BE EXTRACTED!**

# Natural Language Processing Bootcamp

- Some PDFs are created through scanning, instead of being exported from a text editor like Word.
- These scanned PDFs are more like image files, making it much harder to extract the text.
- Often this requires specialized software!

# Natural Language Processing Bootcamp

- To begin, make sure you are using our environment file

# Natural Language Processing Bootcamp

- To install PyPDF , simply open up your command line and directly type:
  - **pip install pypdf**

- Let’s get started!

# Regular Expressions

# Natural Language Processing Bootcamp

- Imagine you needed to search a string for a term, such as “phone”. You can use the in keyword to do this:

**“phone” in “Is the phone here?”**

**>>> True**

# Natural Language Processing Bootcamp

- Now imagine you need to find a telephone number, such as “408-555-1234”, you could do the same:

  **“408-555-1234” in “Her phone is 408-555-1234”**

  **>>> True**

# Natural Language Processing Bootcamp

- But what if you didn't know the exact number?
- If all you knew was the format of the number: **###-###-#####** you would need regular expressions to search through the document for this pattern.

# Natural Language Processing Bootcamp

- Regular expressions allow for pattern searching in a text document.
- The syntax for regular expressions can be very intimidating at first:
  - `r'\d{3}-\d{3}-\d{4}'`

# Natural Language Processing Bootcamp

- The key thing to keep in mind is that every character type has a corresponding pattern code.
- For example, digits have the placeholder pattern code of **\d**
- The use of backslash allows python to understand that it is a special code and not the letter **“d”**.

# Regular Expressions

Continued

# Python Text Basics Assessment

Overview