

HOW TO REGEX

STEP 1: OPEN YOUR FAVORITE EDITOR

@GARABATOKID



STEP 2: LET YOUR CAT PLAY ON YOUR KEYBOARD



Welcome to 61A Lab!

We will begin at **5:10!**
Slides: cs61a.bencuan.me

Announcements

- Scheme
 - Parts 2-3 due tonight
 - Part 4 due 4/26
- HW7 due Thursday
- Final lab next week :o

The Plan

- Regex

Regular Expressions

how to regex

1. Use regex101.com or a similar website
2. Do not memorize anything!
 - a. Just know stuff exists & where to find it
 - b. Write onto cheat sheet

Regex is super useful!

- saves on having really clunky if/elif statements or crazy string manipulation
- super universal- nearly every language will support it
- used in many applications (linux commands, find/replace, data validation/cleaning, future cs+data courses...)

Using regex in python

- ▣ https://www.w3schools.com/python/python_regex.asp
- ▣ tl;dr

```
import re
re.search(r"YOUR_REGEX_HERE", str)
```
- ▣ **r(aw) string:** every character is taken literally (no need to escape special chars)
 - ▣ **Remember to put the r in front of the string!!!**

Regex cheat sheet

- `.` matches **any character**
- `[az]` matches a single **a or z**
- `[a-z]` matches **all lowercase letters**
- `[^az]` matches anything that's **not** a or z
- `+` matches **one or more**
- `*` matches **zero or more**
- `?` matches **zero or one**
- `\w = [a-zA-Z0-9_]`
- `\b = word boundary (punctuation, whitespace)`
- `\d = [0-9]`
- `\s = whitespace`
- `^...$ = from start to end`

Regex examples

- `[A-Z][a-z]+` matches capitalized words (A, Banana, Crab)
- `^[1-9]\d{2}-\d{3}-\d{4}` matches phone numbers ###-###-####
- `[A-Za-z_]\@berkeley\.edu` matches berkeley.edu emails

Things to pay attention to for lab

- What's the difference between (+, ?, and *) ?
- What does ^ and \$ do?
- What does \b do? How is this different from \w?
- How can you represent matching this “or” that in regex?

Work Time!

go.cs61a.org/ben-queue

