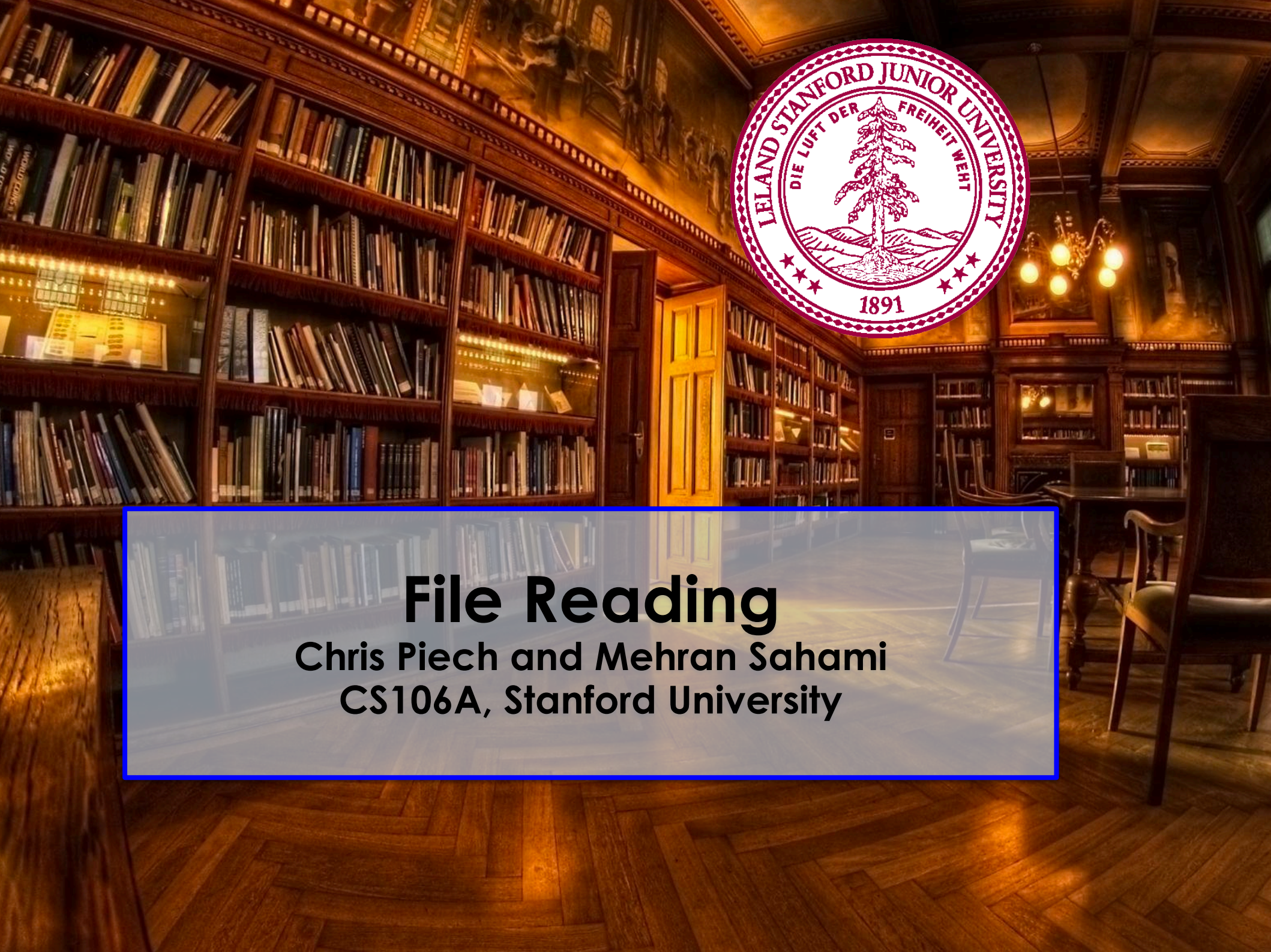
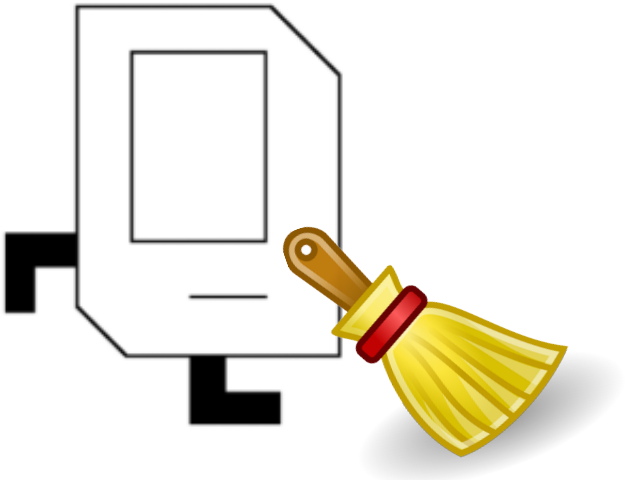


File Reading

Chris Piech and Mehran Sahami
CS106A, Stanford University

Housekeeping



- Happy Monday
- Mehran made an extra example video!
- Mehran AMA Thursday 14th May 4pm PDT
- Diagnostic review sessions
  - First one is tonight 5:30p. Strategies for success in CS106A

Diagnostic Review Sessions

We are going to be holding some additional optional help sessions in week 6 reviewing the material from earlier this quarter. These sessions will be recorded and posted on Canvas, but we suggest you come in person as they will be interactive. Details of these sessions are below:

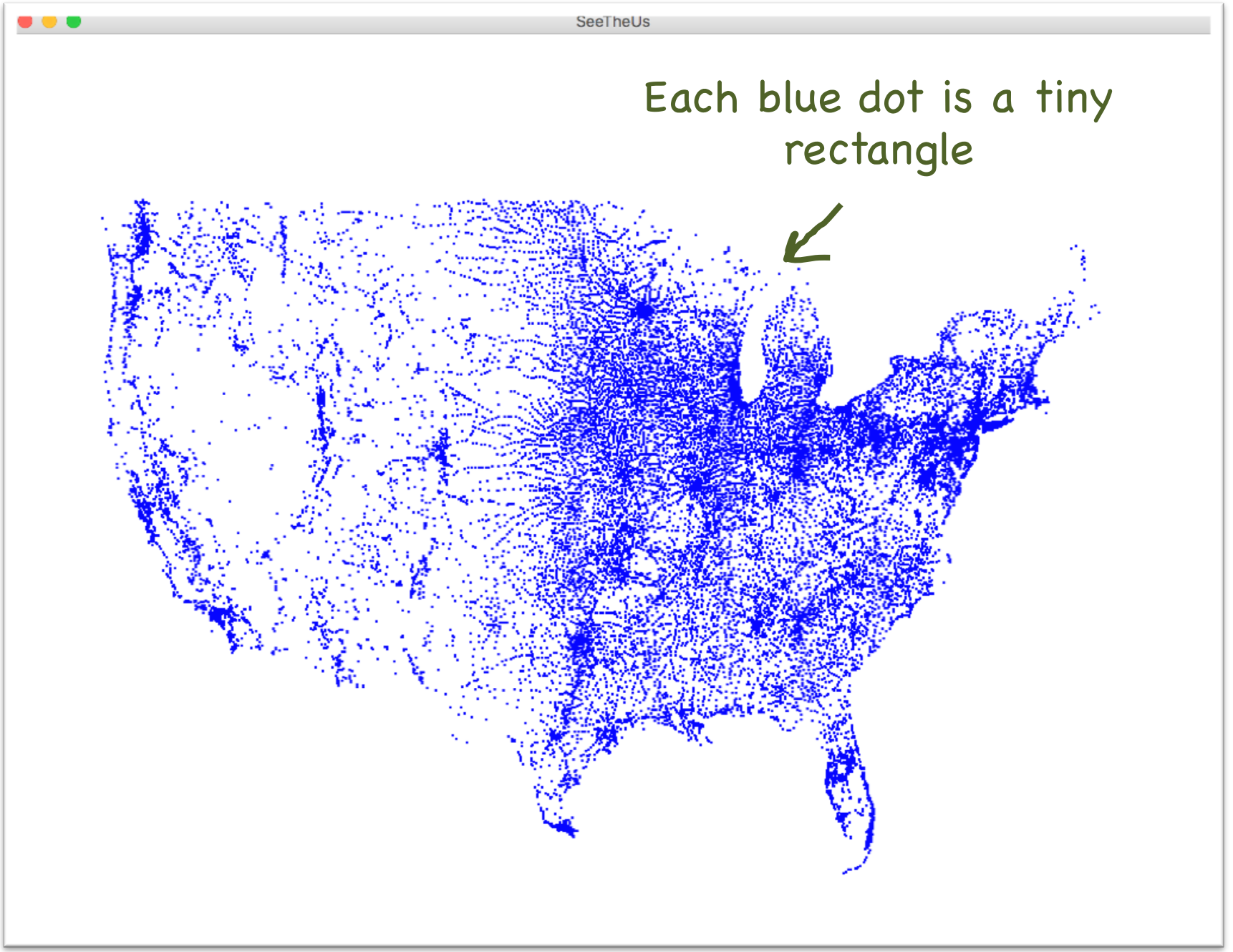
Topic	Time in PDT	Meeting ID
Strategies for Success in CS 106A	Monday, May 11, 5:30pm	96260861110
Control Flow Review	Tuesday May 12, 11:15am	97740758115 (password: **043119**)
Variables Review	Wednesday May 13, 8pm	94788621546
Functions, Parameters & References Review	Thursday May 14, 11am	96342394235

- In the **Strategies for Success in CS 106A** session, Brahm will discuss effective techniques to make sure you internalize the material in 106A, and how to assess your understanding as you go through the quarter. This session will be particularly helpful for you if you are considering reevaluating your approach to the class, if you feel like it's been too time consuming, or if you're not sure how best to consolidate all the material in the class.
- In the **Control Flow** session, we'll review key control flow structures such as conditions, loops, top-down design and incremental testing. This section will be particularly helpful for you if you had issues in problem 2 or 3 of the diagnostic, or if you found assignment 1 tricky.
- In the **Variables** session, we'll review variable usage in programs, doing arithmetic on variables and how they interact with control flow structures like conditions, loops and functions. This section will be particularly helpful for you if you had issues in problem 4 of the diagnostic, or found assignment 2 tricky.
- In the **Functions, Parameters and References** session, we'll review the mechanics of parameter passing and returning between functions. This section will be particularly helpful for you if you found problem 1 of the diagnostic challenging, or found the decomposition in assignment 3 tricky.

If there are other topics that you'd like to review, feel free to shoot Brahm an email - chances are, other students would like to review it as well!

Learning Goals

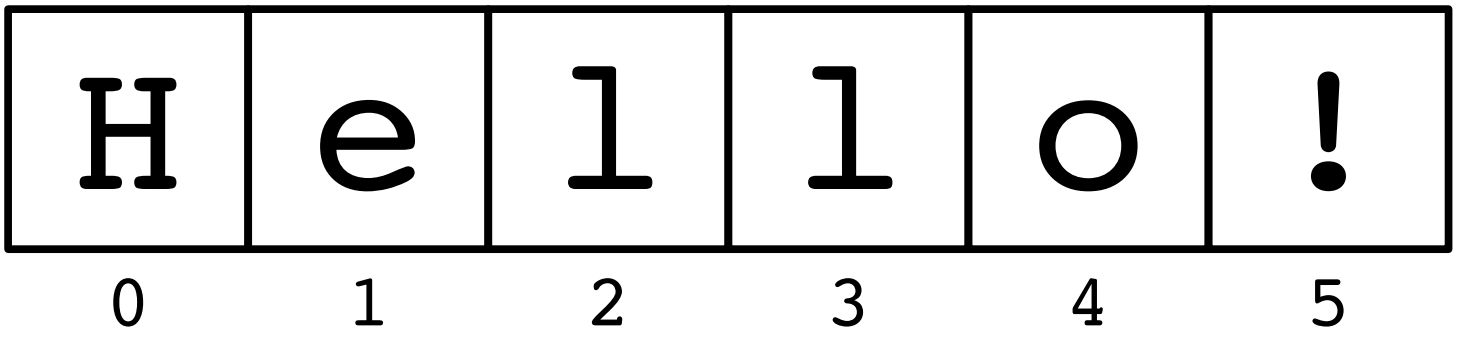
1. Know how to read a file line by line.


SeeTheUs
Each blue dot is a tiny rectangle

Review

Strings are similar to lists of chars

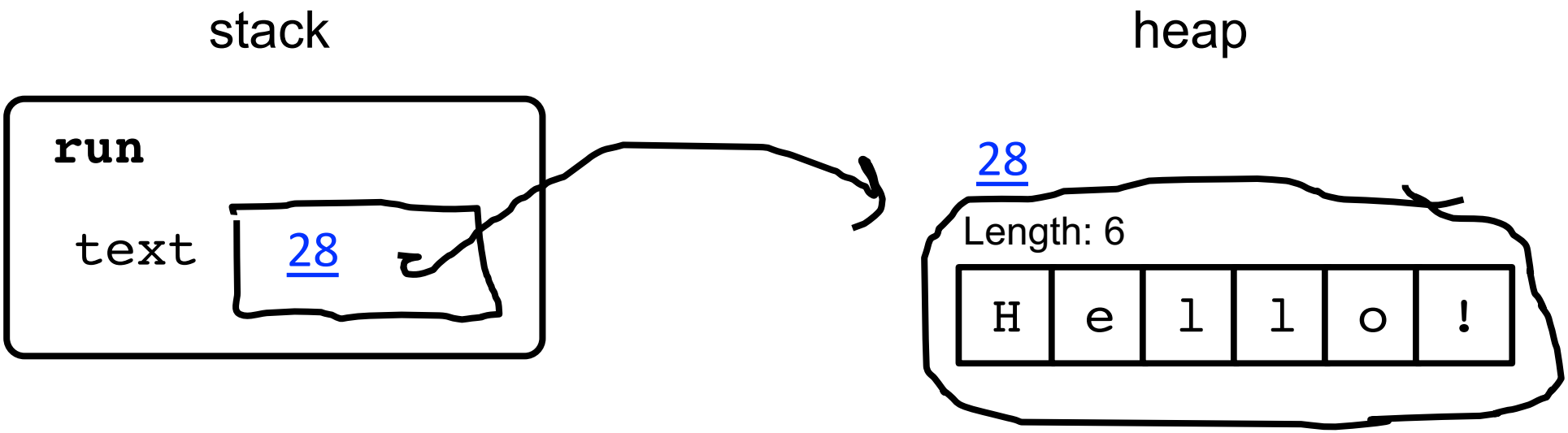
```
def main():
    text = "hello!"
```



***text*[*index*]**

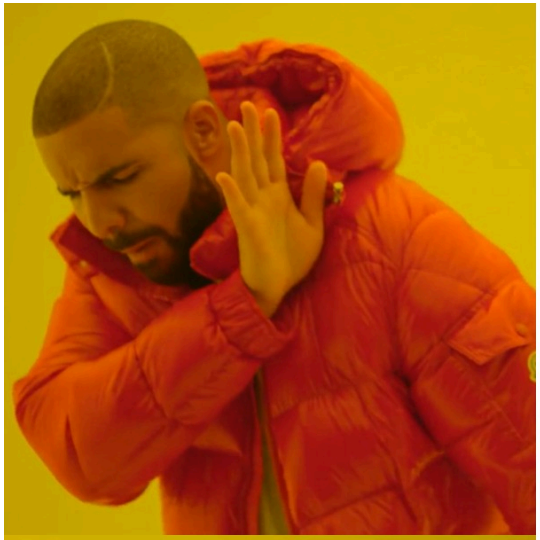
How it is actually stored

```
def main():
    text = "hello!"
```



Strings are Immutable

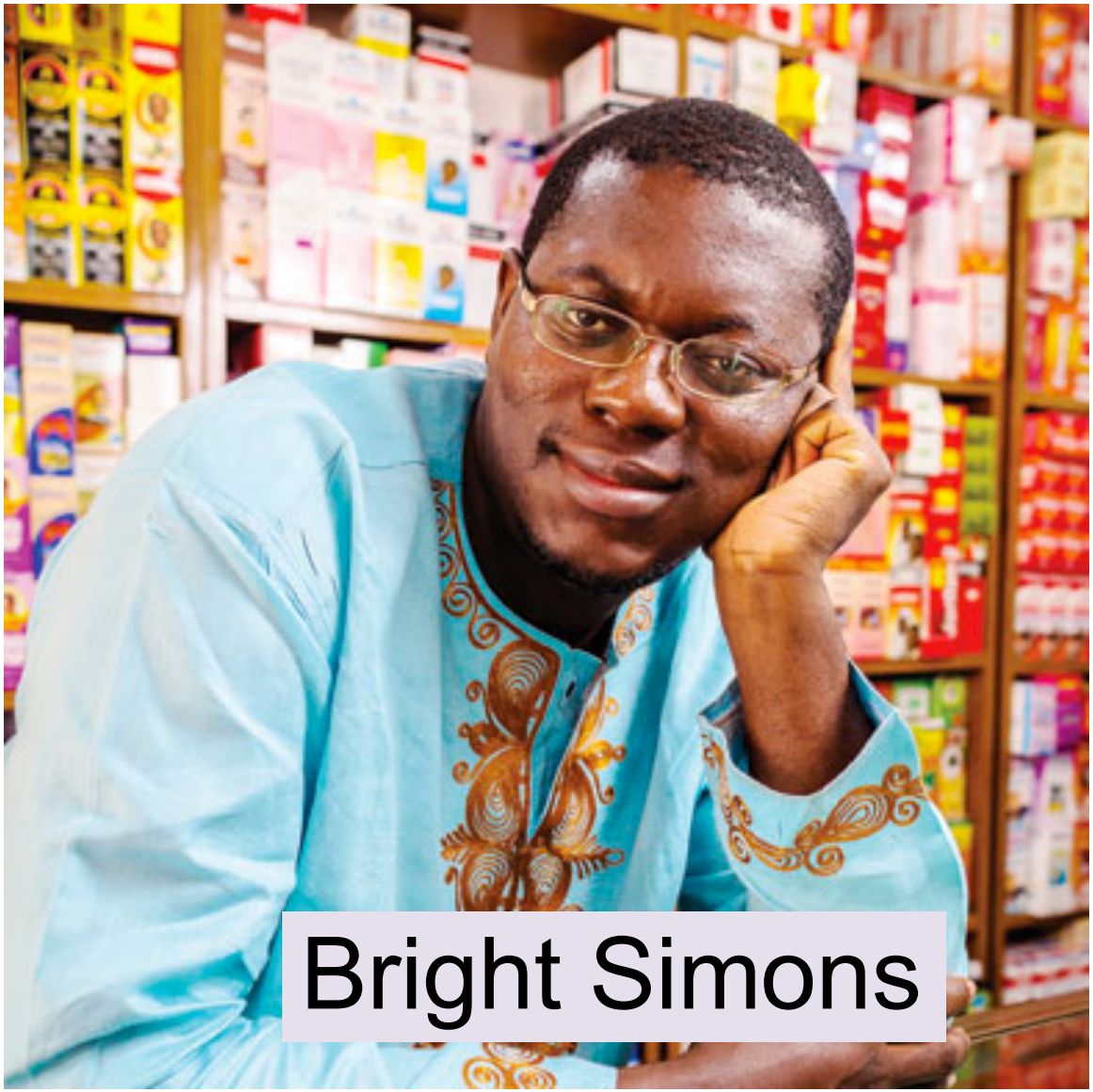
```
x = 'abc'
```



```
x[1] = 'z'
```

```
x = x[0] + 'z' + x[2]
```

Chris' Favorite Program



Underlying Puzzle

Counterfeiter

User

You (Distributor)

Underlying Puzzle

Counterfeiter

User

You (Distributor)

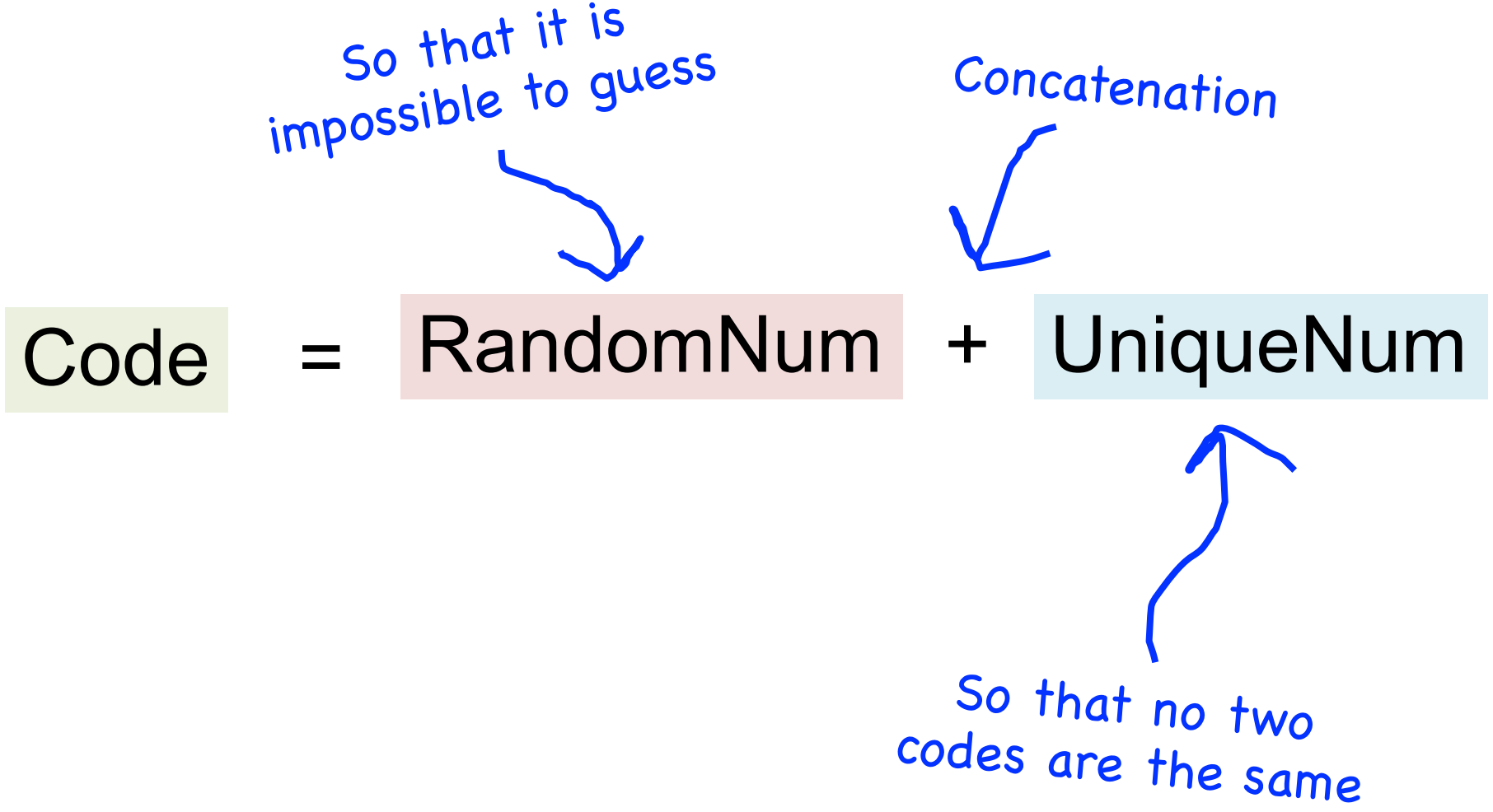
Make a code to
put on every box


Artelum®
Artemether Lumefantrin
40 /240 mg
12 Tablets
Anti-malarial agent
M&B
342302033
BlackBerry
151 / 1 of 6
To: 1393
To:
342302033

1. Unique

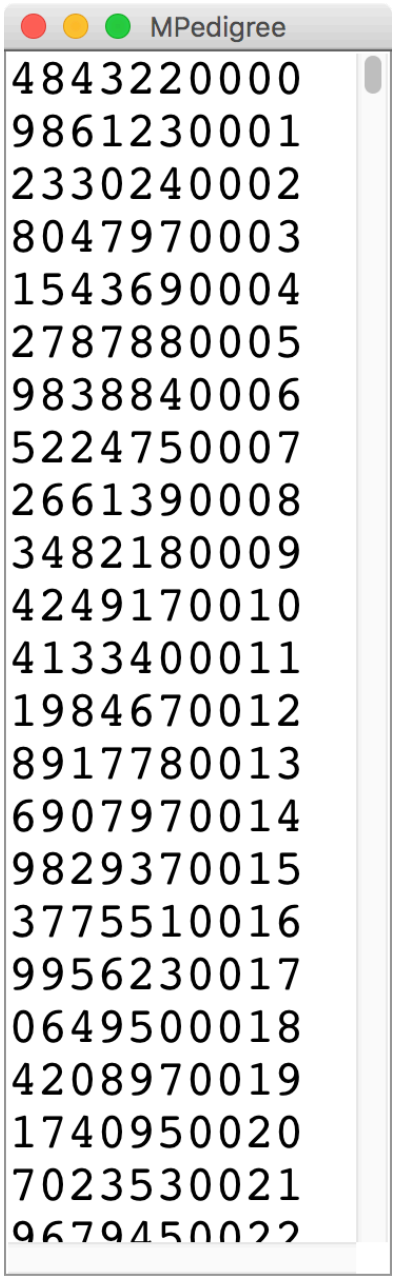
2. Impossible to guess

Insight


So that it is impossible to guess
Concatenation
Code = RandomNum + UniqueNum
So that no two codes are the same

M-Pedigree

Every string should be the same length




```
def main():
  # prints a set of unique labels
  for i in range(N_LABELS):
    rand_value = random.randint(0, MAX)
    rand_part = pad(rand_value, 6)
    unique_part = pad(i, 4)
    print(rand_part + unique_part)

# prepends 0s to a str version of a num
# until the str has goal_length len
def pad(number, goal_length):
  number_string = str(number)
  while len(number_string) < goal_length:
    number_string = '0' + number_string
  return number_string
```

End Review


ATGCTTAAACC..

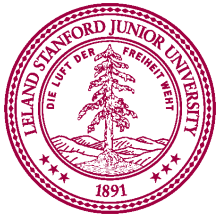
Human Genome Project

"ATGCCAGGAC"

"GGACTTACATTTTT"

"ATTTTTGGCCGGCC"

The human genome has 3 billion base pairs

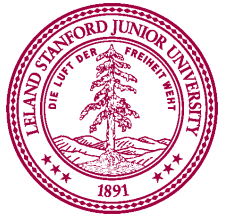


Compose Problem

```
strand1  "GCATCAT"

strand2  "CATTAC"
```

```
result   "GCATCATTAC"
```

The human genome has 3 billion base pairs



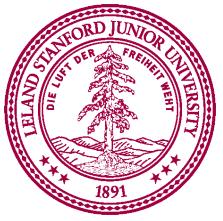
Ha. Gene was working on
The Genome project ☺

Did Gene Myers define all those little pieces as constants?

File Processing!

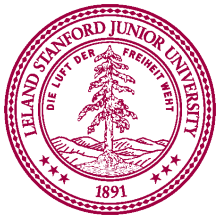
Getting Data into Programs

- Put it directly in the program:
  - Define constants holding your values.
- Get it from the user:
  - Mouse location, input, etc.
- Generate it randomly:
  - Use random.
- Get it from an external source.
  - Store it in a file and read it later.



Reading Files

- Virtually all programs that you've used at some point read files from disk:
  - Word processing (documents)
  - Web browser (cookies)
  - Games (saved progress)
  - PyCharm (Python files)
  - Music player (songs)



The structure of files

- A file is just a series of ***bits*** (ones and zeros).
- Those bits can have structure:
  - Plain-text: Bits represent characters.
  - JPEG: Bits encode information about the structure of an image.
  - MP3: Bits encode frequency information about music.
  - etc.

The structure of files

A file is just a series of ***bits*** (ones and zeros).

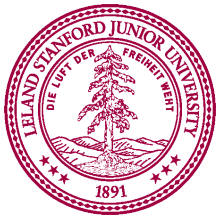
Those bits can have structure:

- Plain-text: Bits represent characters.

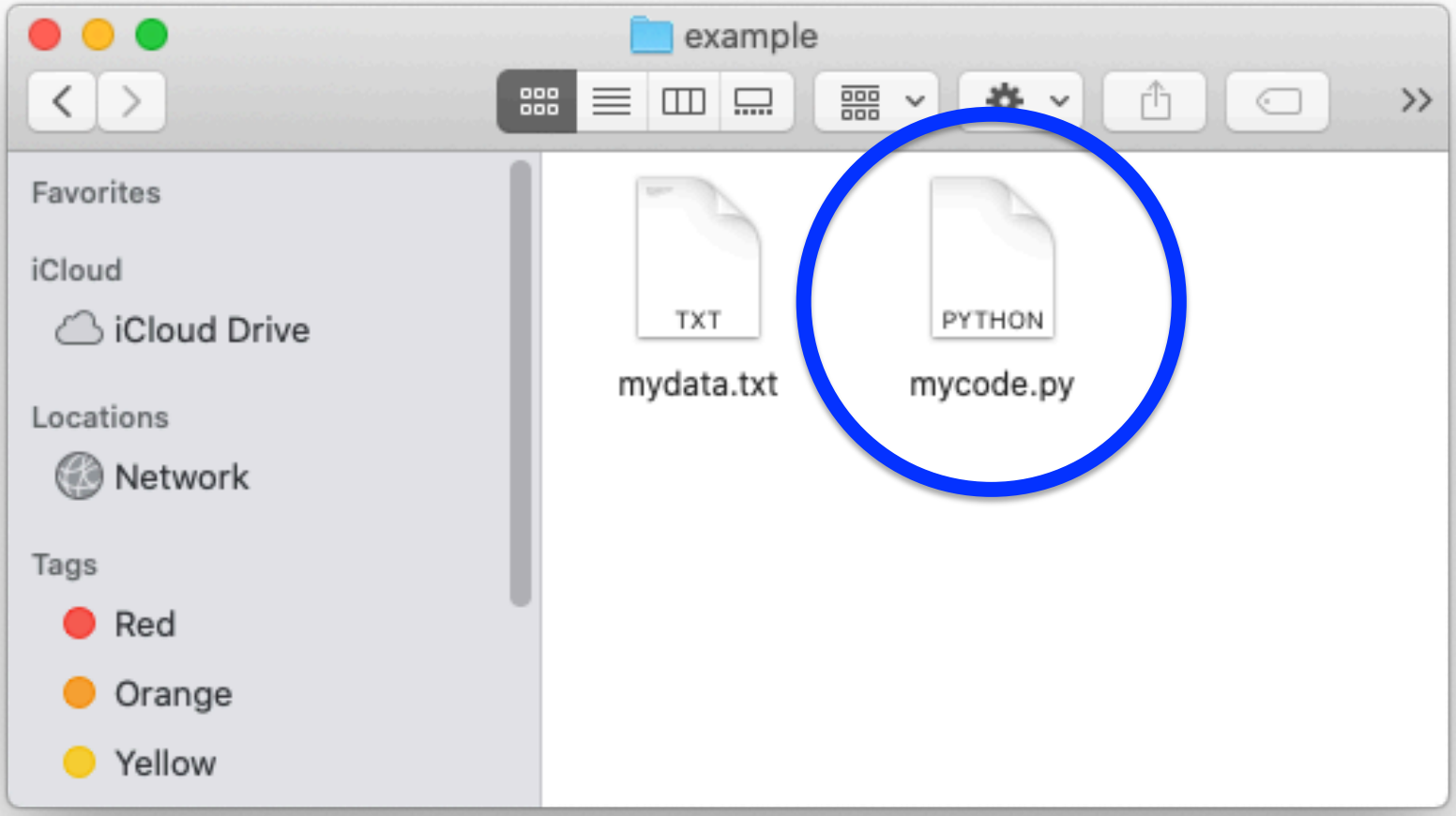
JPEG: Bits encode information about the structure of an image.

MP3: Bits encode frequency information about music.

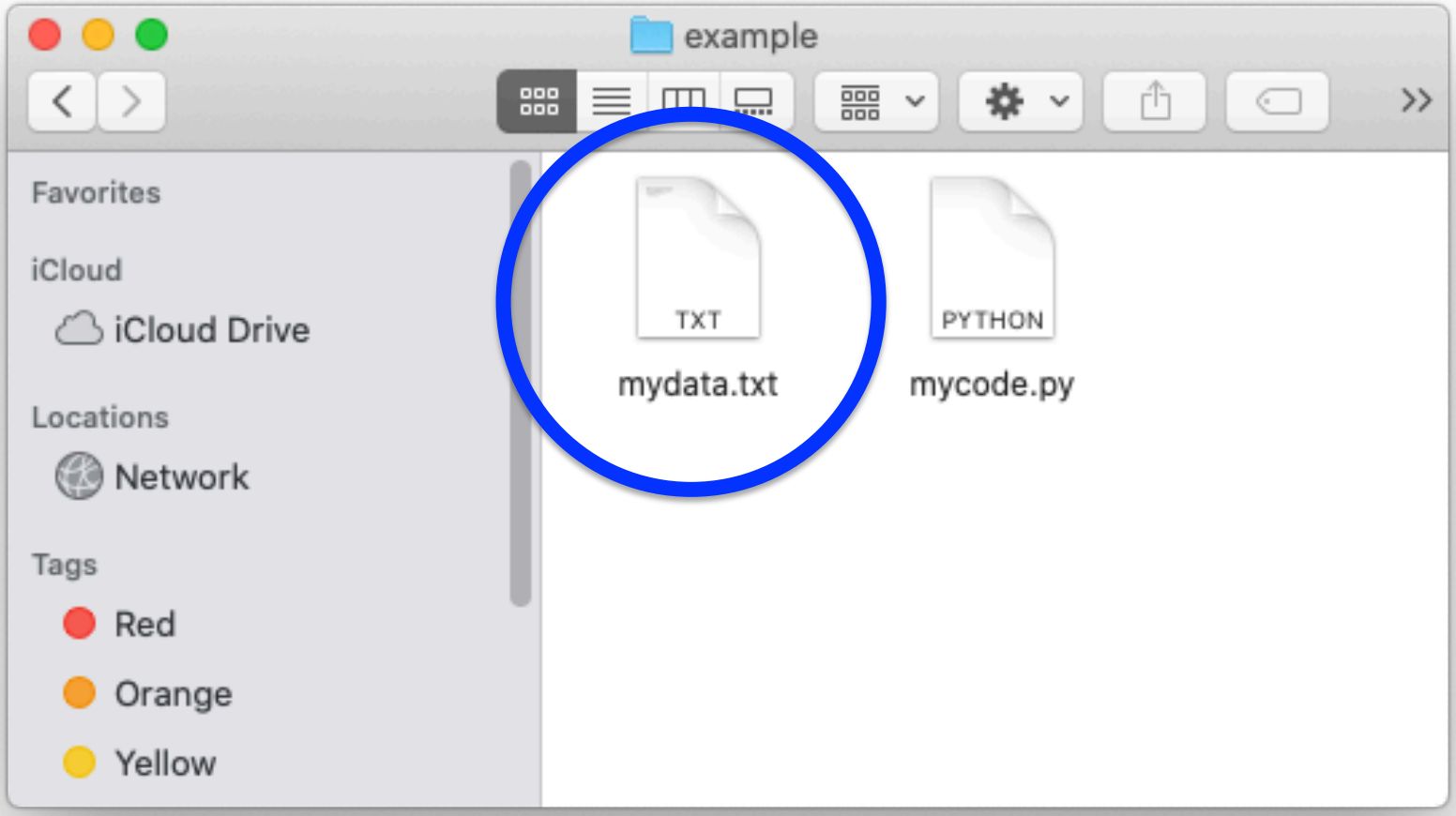
etc.



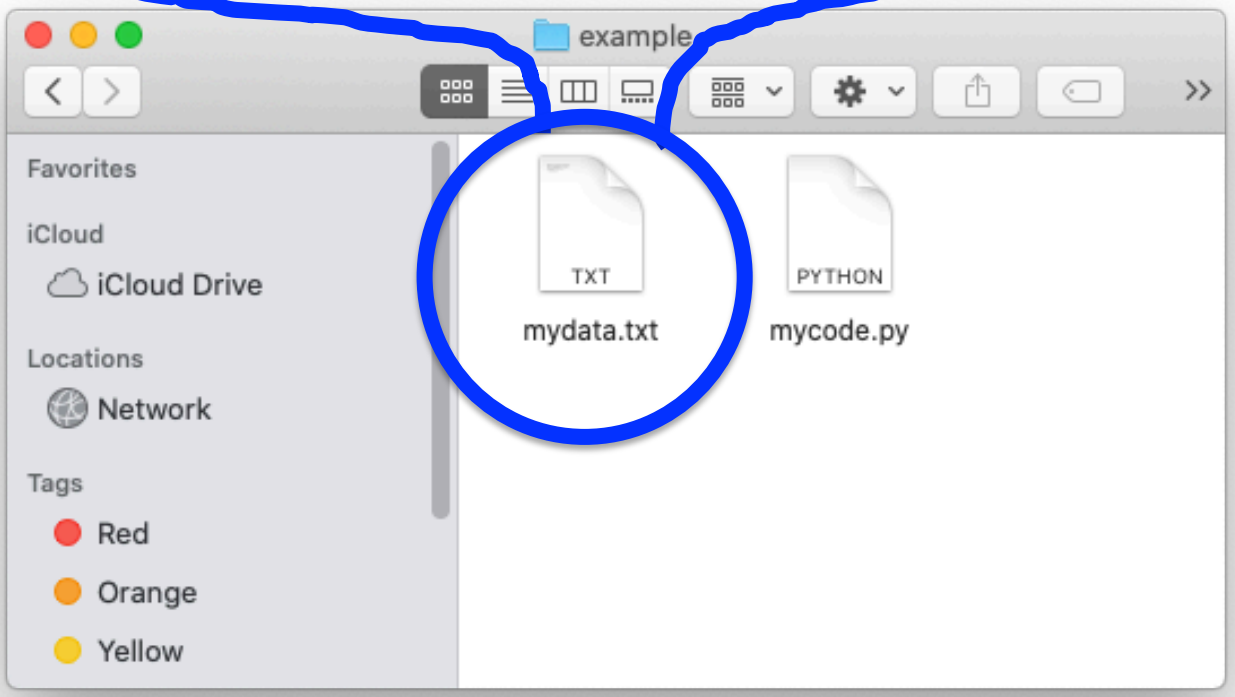
Whats a file?


example
Favorites
iCloud
iCloud Drive
Locations
Network
Tags
Red
Orange
Yellow
TXT
mydata.txt
PYTHON
mycode.py

Whats a file?


example
Favorites
iCloud
iCloud Drive
Locations
Network
Tags
Red
Orange
Yellow
TXT
mydata.txt
PYTHON
mycode.py

Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"



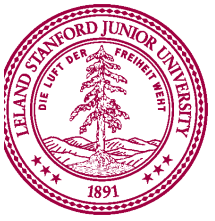
```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```

```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```

```
file = open('mydata.txt')
for line in file:
    print(line)
```

Step One:

Open the file

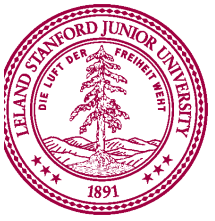


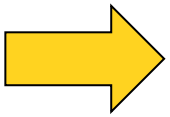
```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```

```
file = open('mydata.txt')
for line in file:
    print(line)
```

Step One:

Open the file



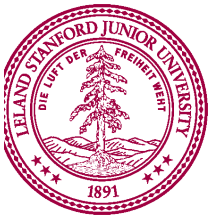


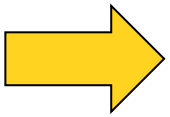
```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```

```
file = open('mydata.txt')
for line in file:
    print(line)
```

Step Two:

Read the file one line at a time





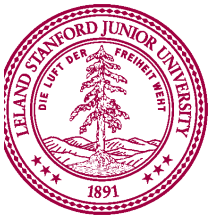
```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```

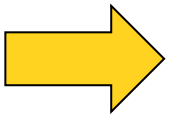
console

```
file = open('mydata.txt')
for line in file:
    print(line)
```

Step Two:

Read the file one line at a time





```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```

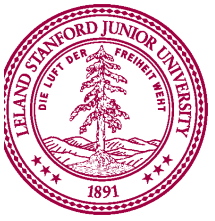
```
file = open('mydata.txt')
for line in file:
    print(line)
```

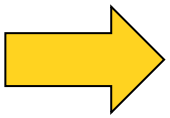
console

```
Neither be cynical about love;
```

Step Two:

Read the file one line at a time





```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```

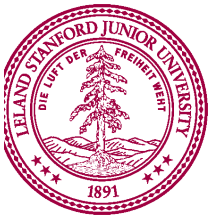
```
file = open('mydata.txt')
for line in file:
    print(line)
```

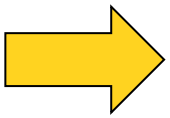
console

```
Neither be cynical about love;
```

Step Two:

Read the file one line at a time





```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```

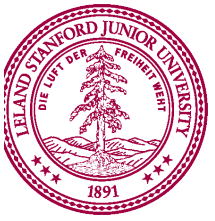
```
file = open('mydata.txt')
for line in file:
    print(line)
```

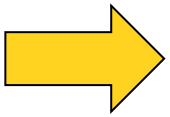
console

```
Neither be cynical about love;

for in the face of all aridity
```

Step Two:

Read the file one line at a time





```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```

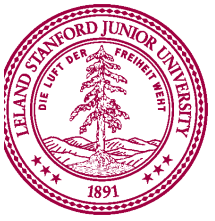
```
file = open('mydata.txt')
for line in file:
    print(line)
```

console

```
Neither be cynical about love;

for in the face of all aridity
```

Step Two:

Read the file one line at a time



```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```

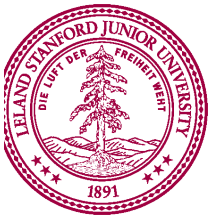
```
file = open('mydata.txt')
for line in file:
    print(line)
```

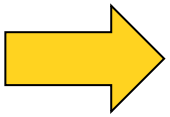
console

```
Neither be cynical about love;

for in the face of all aridity

and disenchantment it is as
```

Step Two:

Read the file one line at a time





```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```

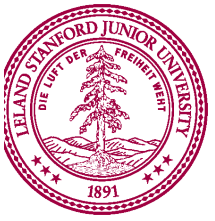
```
file = open('mydata.txt')
for line in file:
    print(line)
```

console

```
Neither be cynical about love;

for in the face of all aridity

and disenchantment it is as
```

Step Two:

Read the file one line at a time



```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```

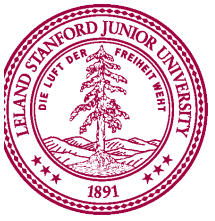
```
file = open('mydata.txt')
for line in file:
    print(line)
```

console

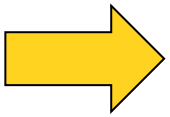
```
Neither be cynical about love;

for in the face of all aridity

and disenchantment it is as

perennial as the grass.
```

Step Two:

Read the file one line at a time



```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```



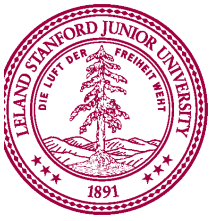
```
file = open('mydata.txt')
for line in file:
    print(line)
```

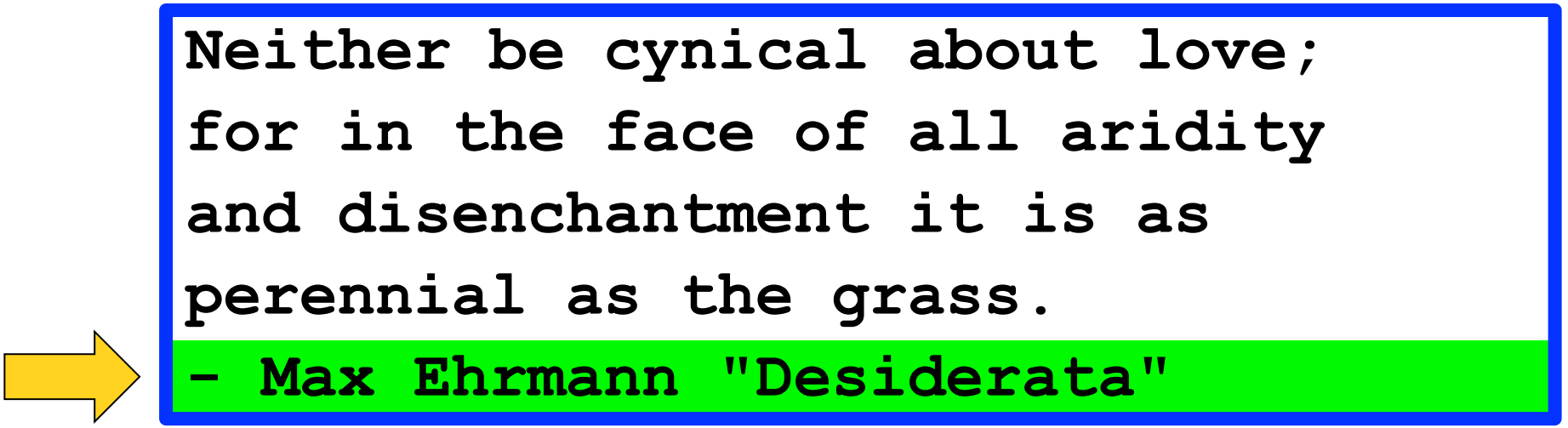
console

```
Neither be cynical about love;

for in the face of all aridity

and disenchantment it is as

perennial as the grass.
```

Step Two:

Read the file one line at a time





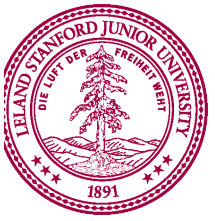

```
file = open('mydata.txt')
for line in file:
    print(line)
```

console

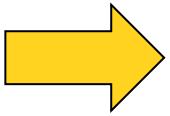
```
Neither be cynical about love;

for in the face of all aridity

and disenchantment it is as

perennial as the grass.

- Max Ehrmann "Desiderata"
```

Step Two:

Read the file one line at a time



```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```



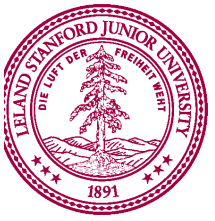
```
file = open('mydata.txt')
for line in file:
    print(line)
```

console

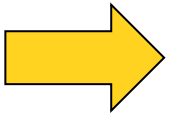
```
Neither be cynical about love;

for in the face of all aridity

and disenchantment it is as

perennial as the grass.

- Max Ehrmann "Desiderata"
```

Step Two:

Read the file one line at a time



```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```



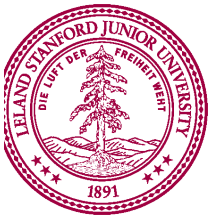
```
file = open('mydata.txt')
for line in file:
    print(line)
```

console

```
Neither be cynical about love;

for in the face of all aridity

and disenchantment it is as

perennial as the grass.

- Max Ehrmann "Desiderata"
```

Step Two:

Read the file one line at a time



Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"

```
file = open('mydata.txt')
for line in file:
    print(line)
```

console

```
Neither be cynical about love;

for in the face of all aridity

and disenchantment it is as

perennial as the grass.

- Max Ehrmann "Desiderata"
```

line

```
'perennial as the grass.\n'
```

```
file = open('mydata.txt')
for line in file:
    print(line)
```

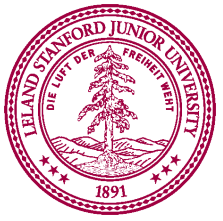
console

```
Neither be cynical about love;

for in the face of all aridity

and disenchantment it is as

perennial as the grass.
```

Take 2

line `'perennial as the grass.'`

```
file = open('mydata.txt')
for line in file:
    line = line.strip()
    print(line)
```

```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```

```
file = open('mydata.txt')
for line in file:
    line = line.strip()
    print(line)
```

console

```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass.
- Max Ehrmann "Desiderata"
```

What does this do?

```
f = open('mydata.txt')
for line in f:
    print(line.strip())
print('-----')
for line in f:
    print(line.strip())
```

Option A

```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass
- Max Ehrmann "Desiderata"
-----
```

Option B

```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass
- Max Ehrmann "Desiderata"
-----
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass
- Max Ehrmann "Desiderata"
```

What does this do?

```
f = open('mydata.txt')
for line in f:
    print(line.strip())
print('-----')
for line in f:
    print(line.strip())
```

The file object doesn't reset after the for loop

Don't read from a file twice…

Option A

```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass
- Max Ehrmann "Desiderata"
-----
```

Option B

```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass
- Max Ehrmann "Desiderata"
-----
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass
- Max Ehrmann "Desiderata"
```

Round 2

What does this do?

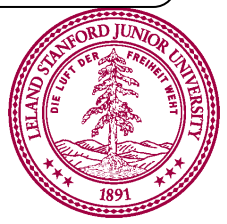
```
f = open('mydata.txt')
next(f)
for line in f:
    print(line.strip())
```

Option A

```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass
- Max Ehrmann "Desiderata"
```

Option B

```
for in the face of all aridity
and disenchantment it is as
perennial as the grass
- Max Ehrmann "Desiderata"
```



What does this do?

next skips a line (and returns it)

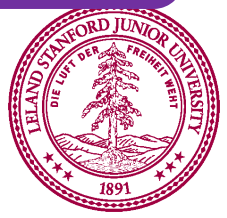
```
f = open('mydata.txt')
next(f)
for line in f:
    print(line.strip())
```

Option A

```
Neither be cynical about love;
for in the face of all aridity
and disenchantment it is as
perennial as the grass
- Max Ehrmann "Desiderata"
```

Option B

```
for in the face of all aridity
and disenchantment it is as
perennial as the grass
- Max Ehrmann "Desiderata"
```




Karel
A robot who
can't turn right!

with or without you?

How edgy are you feeling?

This is the proper way to do it…

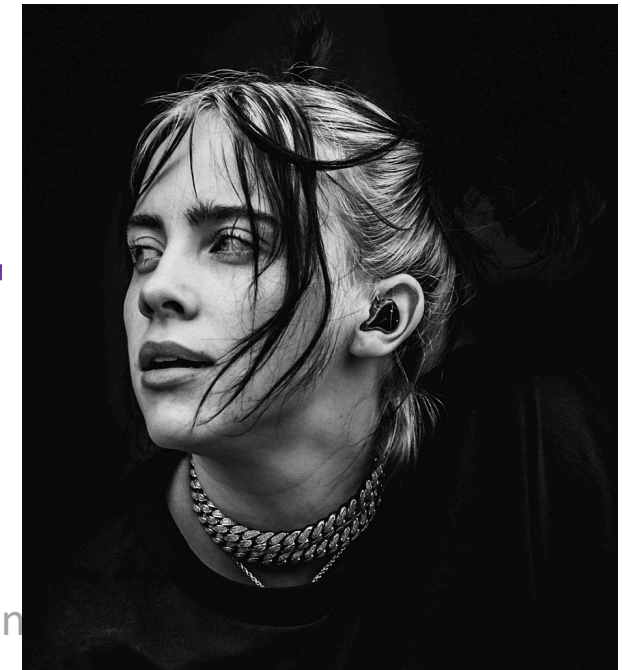
```
with open('mydata.txt') as f:
    for line in file:
        line = line.strip()
        print(line)
```

with *recruits the "file manager" to help python know when it can "close" the file*

How edgy are you feeling?

```
for line in open('mydata.txt'):
    line = line.strip()
    print(line)
```

You actually don't need **with**. The garbage collector will close your file for you...



How edgy are you feeling?

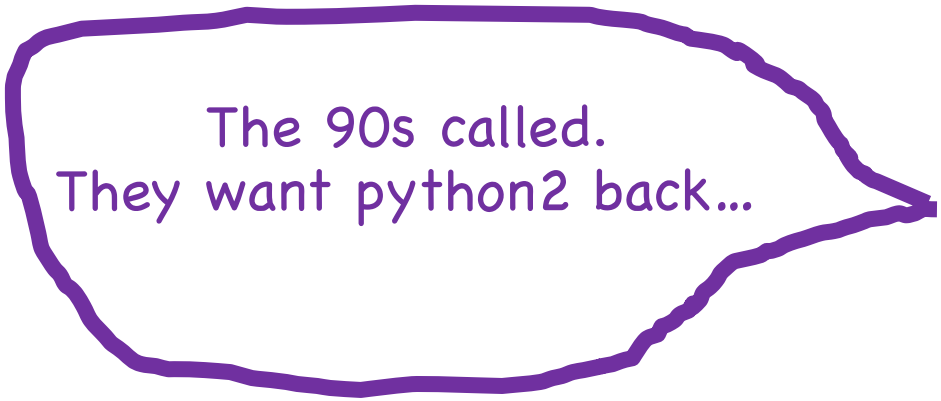
```
with open('mydata.txt') as f:
    for line in file:
        line = line.strip()
        print(line)
```

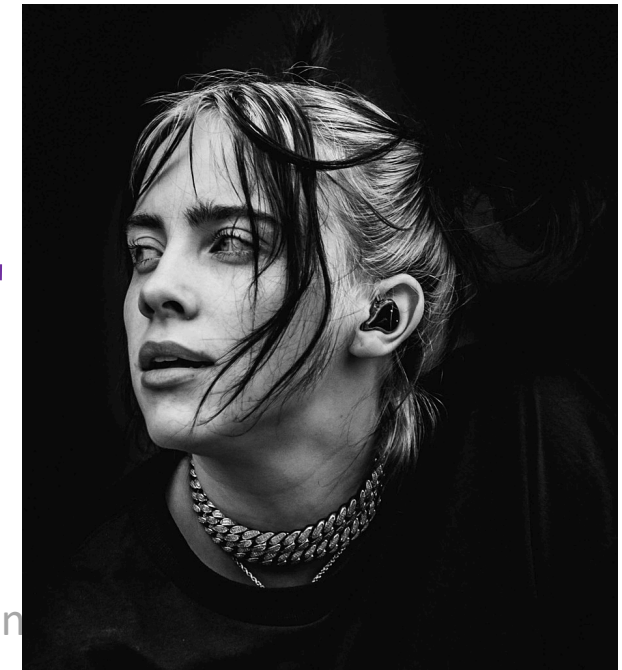
Semisonic wrote a song called "Closing Time"

But some old versions of Python don't garbage collect and then the file is open till the program ends wasting resources

Im a little edgy when it comes to files

```
for line in open('mydata.txt'):
    line = line.strip()
    print(line)
```

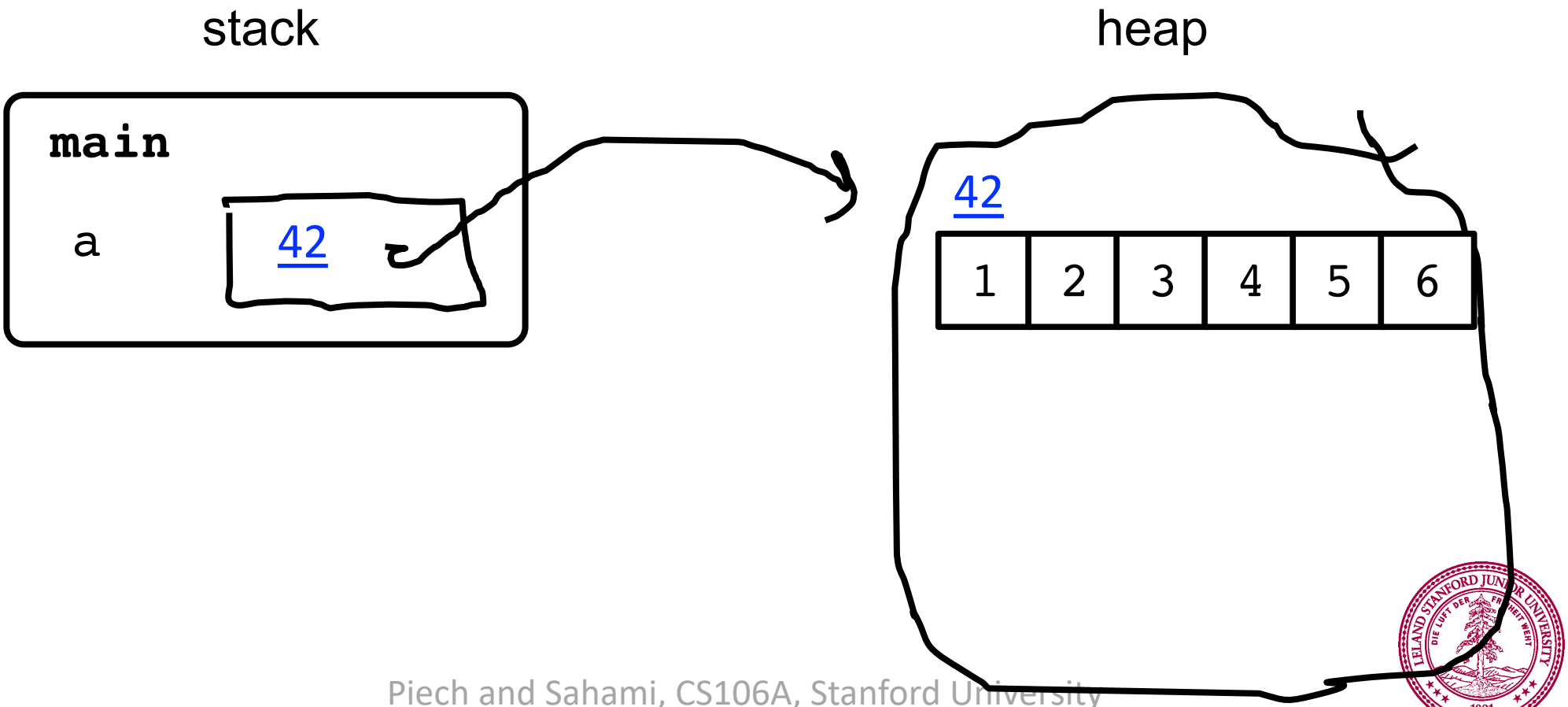




Aside: What is the Garbage Collector?

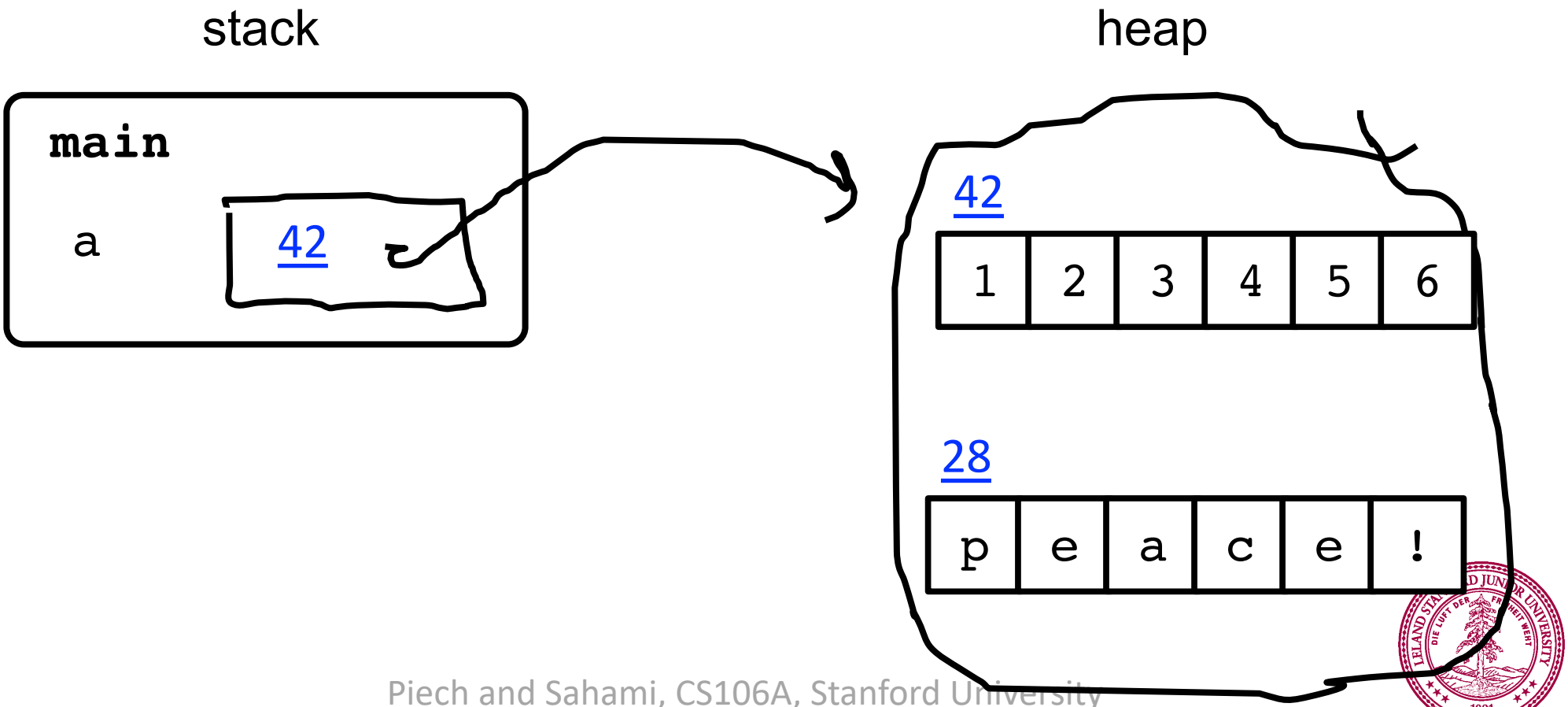
Aside: What is the Garbage Collector?

```
a = [1, 2, 3, 4, 5, 6]
a = 'peace!'
```



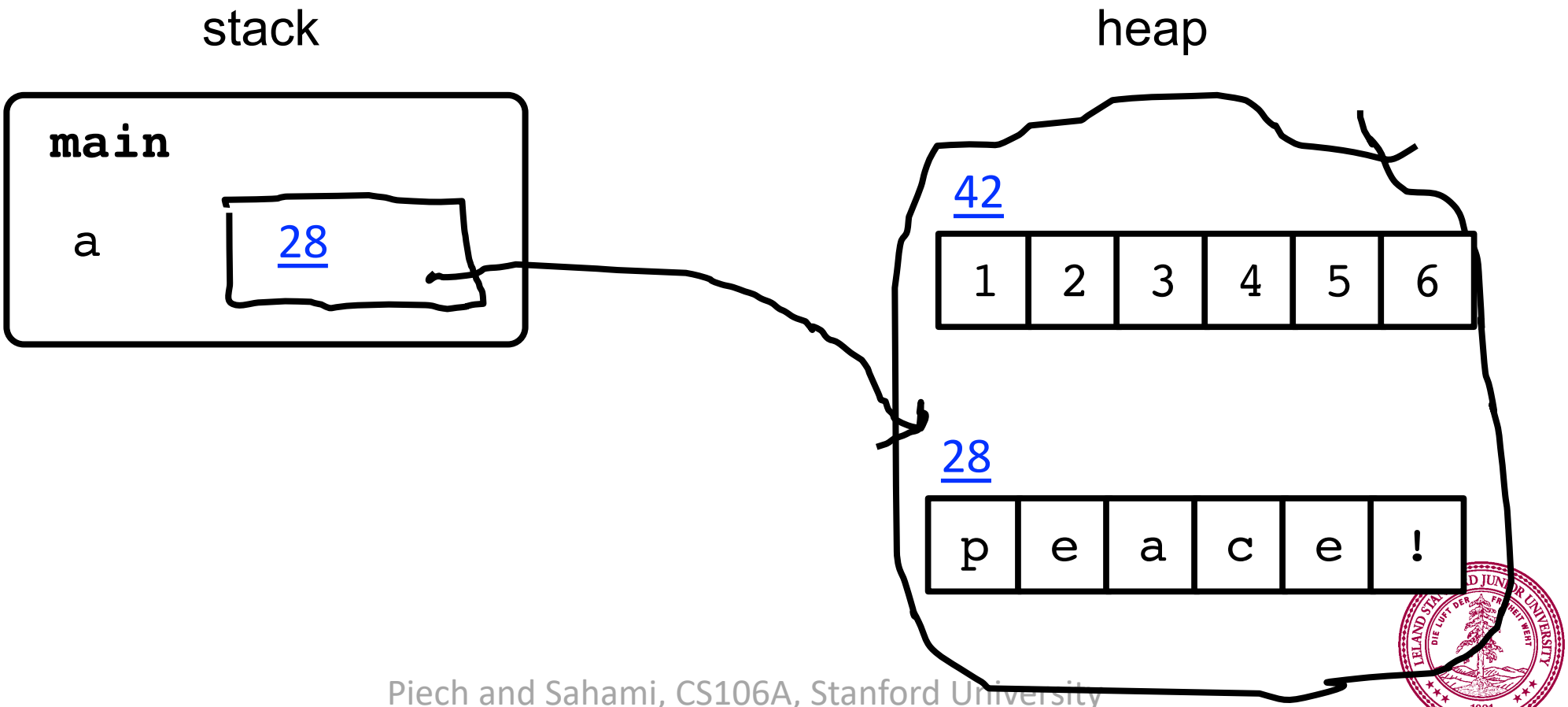
Aside: What is the Garbage Collector?

```
a = [1, 2, 3, 4, 5, 6]
a = 'peace!'
```



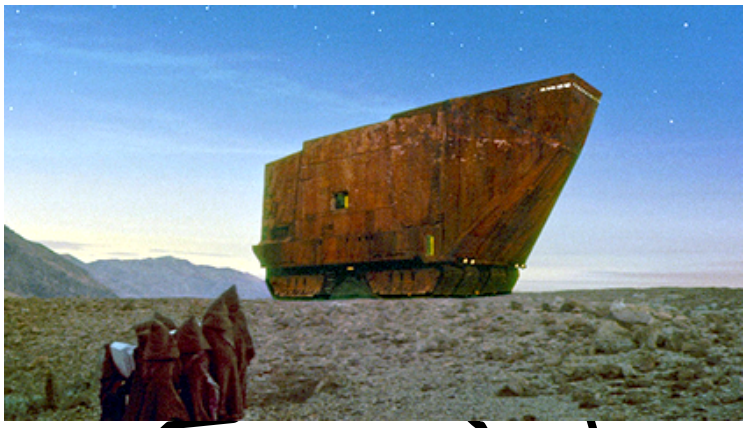
Aside: What is the Garbage Collector?

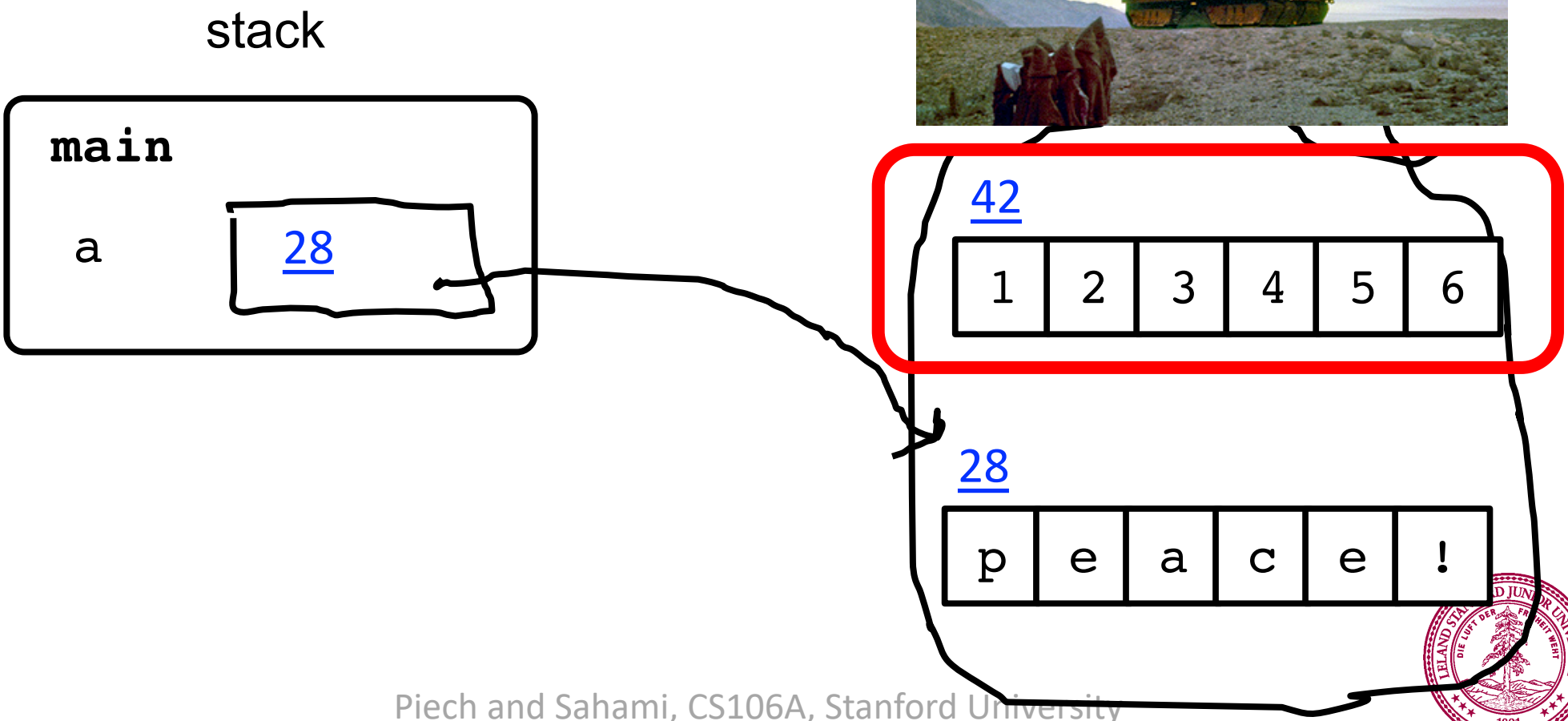
```
a = [1, 2, 3, 4, 5, 6]
a = 'peace!'
```



Aside: What is the Garbage Collector?

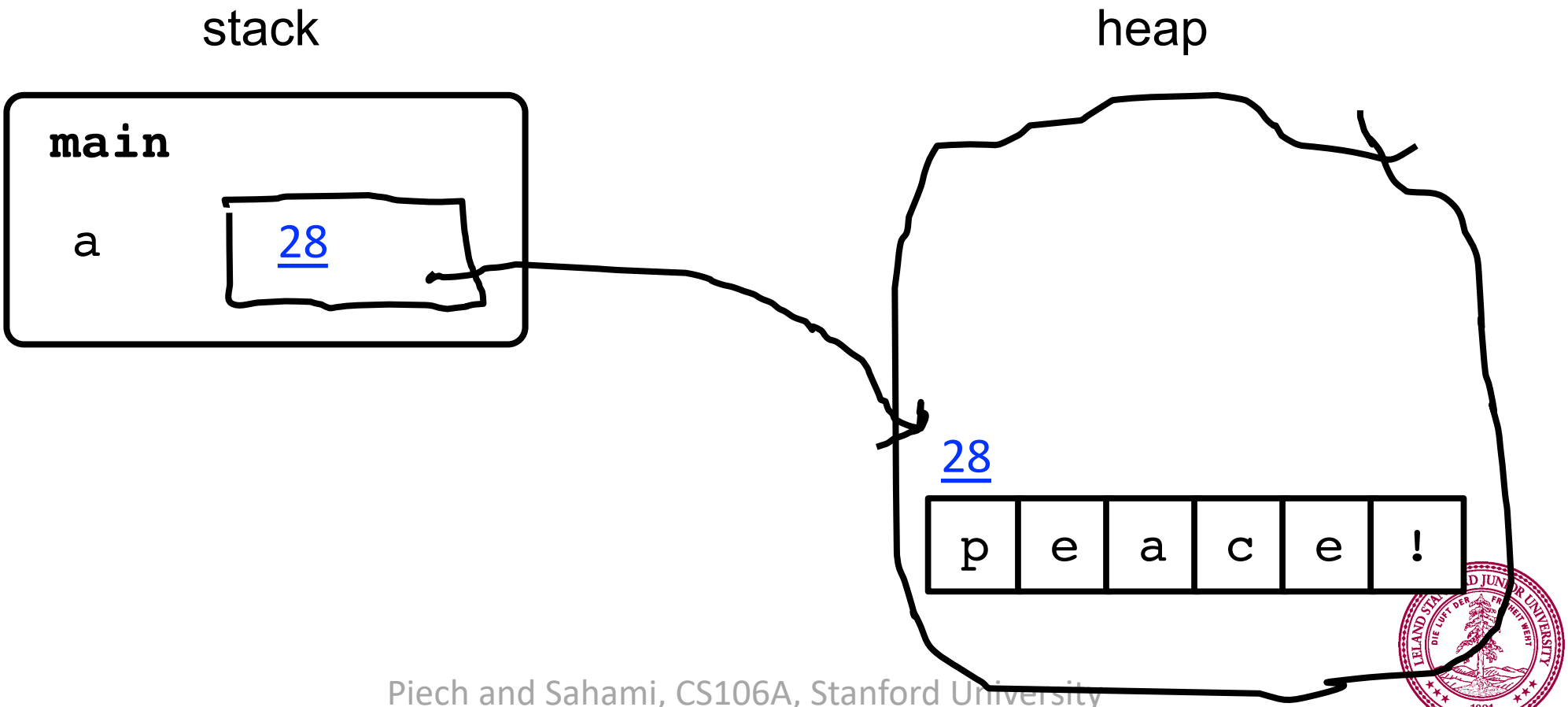
```
a = [1, 2, 3, 4, 5, 6]
a = 'peace!'
```





Aside: What is the Garbage Collector?

```
a = [1, 2, 3, 4, 5, 6]
a = 'peace!'
```



with or without you?

6 song references later…

a lesson on garbage collection…

My Take


```
with open('mydata.txt') as f:
    for line in file:
        line = line.strip()
        print(line)
```

```
for line in open('mydata.txt'):
    line = line.strip()
    print(line)
```

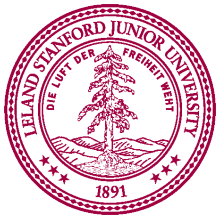
People bother you less if you use the first one

But I would guess in the future the second one takes over…

Files + Strings = DataScience!

dataset.csv

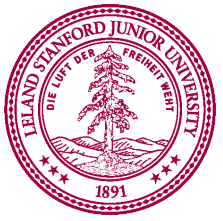
```
Kenya,100,50
Malaysia,50,100

...

Turkey,20,20
Spain,95,95
```



dataset.csv

```
Kenya,100,50
Malaysia,50,100

...

Turkey,20,20
Spain,95,95
```

```
for line in file:
    line = line.strip()
    values = line.split(',')
    print(values[1])
```



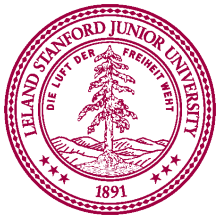
dataset.csv

```
Kenya,100,50
Malaysia,50,100

...

Turkey,20,20
Spain,95,95
```

```
for line in file:
    line = line.strip()
    values = line.split(',')
    print(values[1])
```

line

```
'Kenya,100,50\n'
```



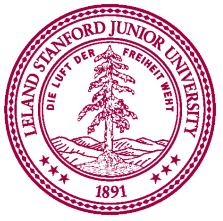
dataset.csv

```
Kenya,100,50
Malaysia,50,100

...

Turkey,20,20
Spain,95,95
```

```
for line in file:
    line = line.strip()
    values = line.split(',')
    print(values[1])
```

line

```
'Kenya,100,50\n'
```



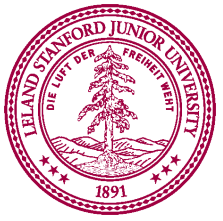
dataset.csv

```
Kenya,100,50
Malaysia,50,100

...

Turkey,20,20
Spain,95,95
```

```
for line in file:
    line = line.strip()
    values = line.split(',')
    print(values[1])
```

line

```
'Kenya,100,50'
```



dataset.csv

```
Kenya,100,50
Malaysia,50,100

...

Turkey,20,20
Spain,95,95
```

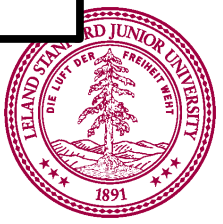
```
for line in file:
    line = line.strip()
    values = line.split(',')
    print(values[1])
```

line

```
'Kenya,100,50'
```

values

'Kenya'	'100'	'50'



dataset.csv

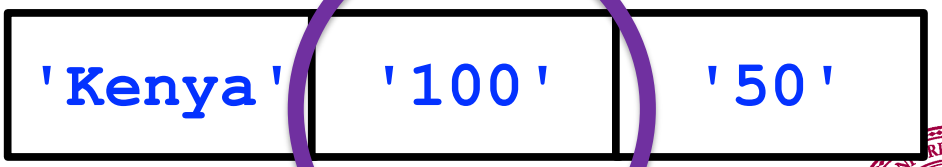
```
Kenya,100,50
Malaysia,50,100

...

Turkey,20,20
Spain,95,95
```

```
for line in file:
    line = line.strip()
    values = line.split(',')
    print(values[1])
```

line

```
'Kenya,100,50'
```

values



US Census Data

Thanks Keith for the cool dataset