Iteration. Iteration. Iteration. Iteration.

Chapter 7

Today's Outline

- Variable assignment
- while loop! (and how they are different from if/else/elif)
- Brief nod to web scraping
- Ethics of web scraping!

What is Iteration?

- The ability to run a block of statements repeatedly.
- What are examples we've seen of this already?
  - Recursion (`if/elif/else`) in chapter 5 and `for` loops in chapter 4.
- Why is this important?
  - Think about any process that's either really big and needs to be broken up (e.g. big data) or something that involves a long list… changing file names, for example.

Reassignment

- What is it?
  - New assignment to existing variables to give them *new* values.
  - A common form of reassignment is the update and it depends on the old value. E.g. `x = x + 1`
  - This can be done *multiple times*.
- Why is this important?
  - When you need to update a value… think of averages.

Swapping Values

- A shell game with numbers.
- Understand that values can change.

```
intX = -128
intY = 127
print(intX)
print(intY) # so far so good

temp = intX # use a temporary variable to hold value of intX.
intX = intY # We then put the value of intY in intX
intY = temp # We then put the value of temp in intY

print(intX)
Print(intY)
```

Revisiting “Blastoff”

- Using `while` statement

```
def countdownWhile(n):
    while n > 0:
        print(n)
        n = n-1
    print("Blastoff!")

countdownWhile(10)
```

While Statement

- Explain to me, in plain English, what is happening here.

```
def countdown(n):
    while n > 0:
        print(n)
        n = n-1
    print('Blastoff!')
```

- "While n is greater than 0, display the value of n and then **decrement** n. When you get to zero (n is greater than zero), print "Blastoff."


SOOOOOO POWERFUL!

While Statement

- Why is this important?
  - We can use this to automate repetitive tasks!!!
- Formal flow of execution for a `while` statement.
  - Determine whether a condition is True or False.
  - If False, exit the while statement and continue execution at the next statement.
  - If the condition is True, run the body and go back to step 1.
- This type of flow is called a loop because it goes back around to the top.

Revisiting "Blastoff"

- Using `if` statement

```
def countdownIf(n):
    if n == 0: #base case
        print("Blastoff!")
    else:    #general case
        print(n)
        countdown(n-1)
```

- Using `while` statement

```
def countdownWhile(n):
    while n > 0:
        print(n)
        n = n-1
    print("Blastoff!")
```

- What's the difference here?

While versus If

- An **if** statement checks if an expression is **true or false**, and then runs the code inside the statement only if it is true.
  - The code inside the loop is only run once...
- A **while** statement is a **loop**. It continues executing the code in the while statement for however long the expression is true.

While + Reassignment

- Write code that sums all numbers from 1 through 99.

```
n = 1
sum = 0
while n <= 99:
      sum = sum + n
      n = n + 1
print(sum)
```

While + Reassignment

- Write code that sums all *even* numbers from 1 - 99.
- How can I check if a number is even?

```
n = 1
sum = 0
while n <= 99:
    if n % 2 == 0:
        sum = sum + n
    n = n + 1
print(sum)
```

Breaks

- What is it?
  - A way to exit a loop!

- Take input from a user until they type the word “done”

```
while True:
        line = str(raw_input("Type Here > "))
        if line == “done”:
                break
        print(line)
```

Notice `while True` which makes the loop condition always true. It runs until it hits the `break`.

While Statement

- What's the catch?
- Infinite loop! What is it and how do you avoid it?
  - The body of a loop needs to change the value of one or more variables so the the condition becomes False and eventually terminates.
  - Doing these slides I accidentally crashed my machine with a misplaced indent…

Quick Exercise

- Use a while loop to add all numbers from 1 to 20 that can be divided by 5. Below is the code for adding even numbers between 1 to 99

```
n = 1
sum = 0
while n <= 99:
    if n % 2 == 0:
        sum = sum + n
    n = n + 1
print(sum)
```

Exercise Answer

- Use a while loop to add all numbers from 1 to 20 that can be divided by 5.

```
n = 1
sum = 0
while n <= 20: #Changed this to stop at 20
     if n % 5 == 0: #Divisible by 5.
          sum = sum + n
     n = n + 1
print(sum)
```

Web Scraping

- The use of a program or algorithm to extract and process data from the web.
- Specifically: downloading structured data, selecting some of that data, and passing along what you selected to another process.
- Why might this be important?

Web Scraping

- Getting data that are regularly updated: weather, stocks, etc.
- Web interaction?
  - Imagine wanting a proxy indicator for… popularity.
  - How might you do this?
- Working with "**user generated content**."

Web Scraping: basics

- Decide what you want to accomplish & identify the type of data needed!!!
- Import libraries to request information from the web and to process it.
- Write code to download data: This means getting familiar with web pages and HTML tags.
- Parse out what you downloaded, structure it, wrangle it into shape.

Basics of a Web Page

- HTML — language that contains the main content of the page.
- CSS — adds styling to make the page look nicer.
- Javascript files: add interactivity to web pages.
- Images — image formats, such as JPG and PNG allow web pages to show pictures.

Basics of a Web Page

- To find the "elements" that you want to capture (scrape) you need to know what they are called (how they are tagged) in the HTML.
- Do that by right-clicking on a web page and choosing "inspect element", which shows you the structure of the page!

Example

- Navigate to https://greenteapress.com
- Right-click on the page and choose “inspect element”.
- Use the inspector tools to select the sub-section called “Free Books.”
- Now look at the corresponding HTML code below, what “tag” is shown for this bit of text?

https://greenteapress.com/wp/

Most Visited Getting Started Clark Mail Moodle

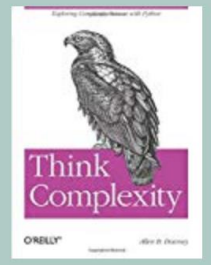
Green Tea Press

Free books by Allen B. Downey

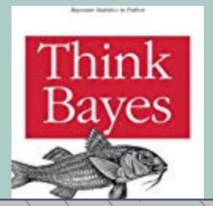
Search ...

GREEN TEA BOOKS ALSO PUBLISHED BY O'REILLY MEDIA

Think Complexity 2e



Think Bayes



Home

Welcome to Green Tea Press, publisher of *Think Python*, *Think Bayes*, and other books by Allen Downey.

Read our Textbook Manifesto.

Free Books!

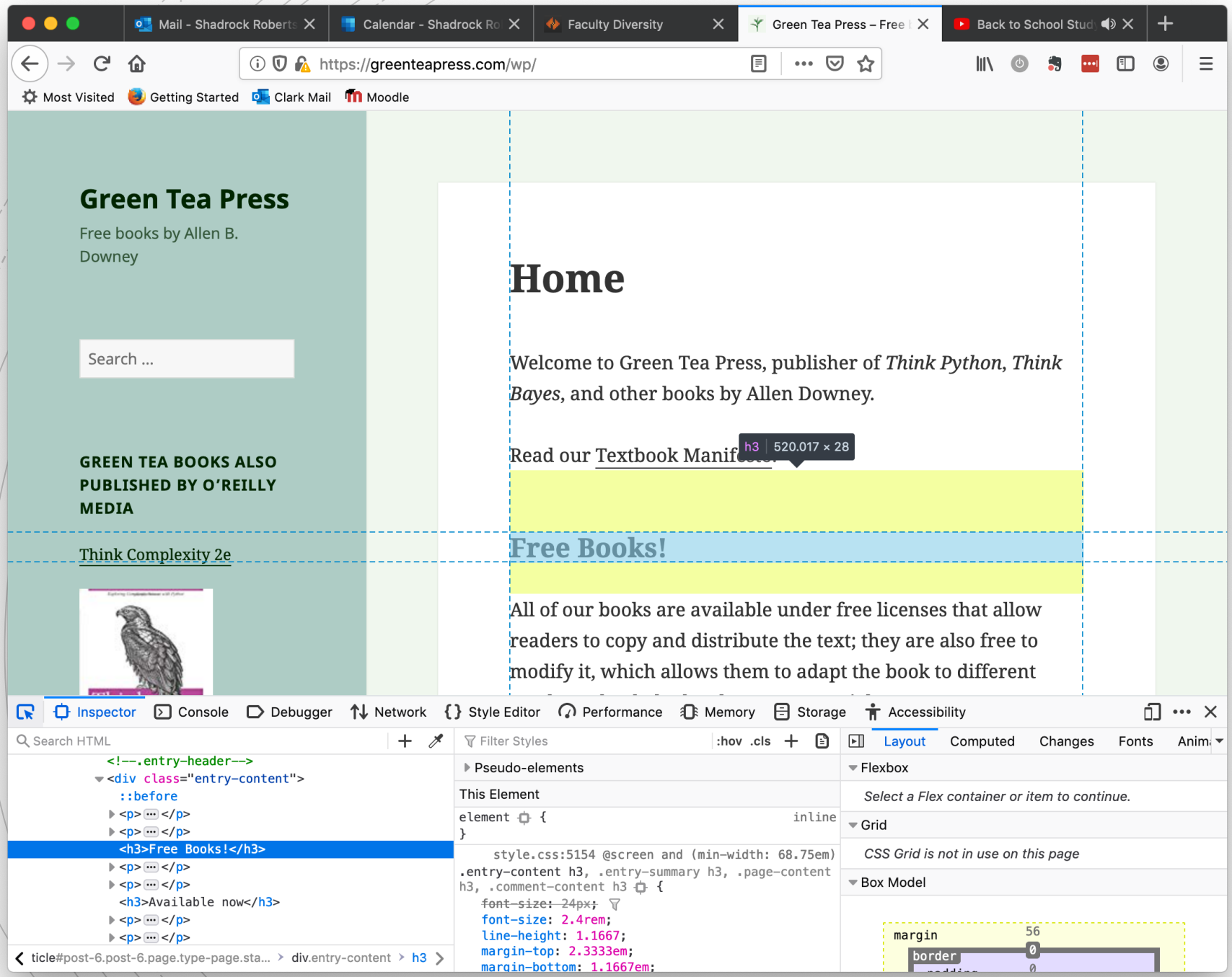
All of our books are available under fre
readers to copy and distribute the text;
modify it, which allows them to adapt t
needs, and to help develop new materia

These books are available in a variety c
some are also for sale in hard copy.

Available now

Think Python 2e: How To Think Like a Computer Scientist

Save Page As...
Save Page to Pocket
Send Page to Device
View Background Image
Select All
View Page Source
View Page Info
Inspect Element
Take a Screenshot
Start timer
Evernote Web Clipper
LastPass


Mail - Shadrock Roberts
Calendar - Shadrock Ro
Faculty Diversity
Green Tea Press – Free
Back to School Stud
https://greenteapress.com/wp/
Most Visited
Getting Started
Clark Mail
Moodle
Green Tea Press
Free books by Allen B. Downey
Search ...
GREEN TEA BOOKS ALSO PUBLISHED BY O'REILLY MEDIA
Think Complexity 2e
Home
Welcome to Green Tea Press, publisher of Think Python, Think Bayes, and other books by Allen Downey.
Read our Textbook Manifesto.
h3 | 520.017 × 28
Free Books!
All of our books are available under free licenses that allow readers to copy and distribute the text; they are also free to modify it, which allows them to adapt the book to different
Inspector
Console
Debugger
Network
Style Editor
Performance
Memory
Storage
Accessibility
Search HTML
<!--.entry-header-->

::before
 …
 …
<h3>Free Books!</h3>
 …
 …
<h3>Available now</h3>
 …
 …
ticle#post-6.post-6.page.type-page.sta… › div.entry-content › h3
Filter Styles
:hov .cls
Pseudo-elements
This Element
element { inline }
style.css:5154 @screen and (min-width: 68.75em)
.entry-content h3, .entry-summary h3, .page-content h3, .comment-content h3 {
font-size: 24px;
font-size: 2.4rem;
line-height: 1.1667;
margin-top: 2.3333em;
margin-bottom: 1.1667em;
Layout
Computed
Changes
Fonts
Flexbox
Select a Flex container or item to continue.
Grid
CSS Grid is not in use on this page
Box Model
margin
56
border
0

Web Scraping Ethics

- Can overload servers – there are ways to spread your requests out over time.
  - Who knows what a Distributed Denial of Service (DDoS) attack is?
- Licensing, Terms of Service… are you allowed to scrape the pages?

Summary

- Variable assignment: important for allowing variable to be changed.
- while loop: powerful form of iteration that "loops" through code while a condition is true.
- Brief nod to web scraping – key takeaway is that you need to understand what you're looking for and how to find that in the structure of a web page.
- There are always ethical considerations!