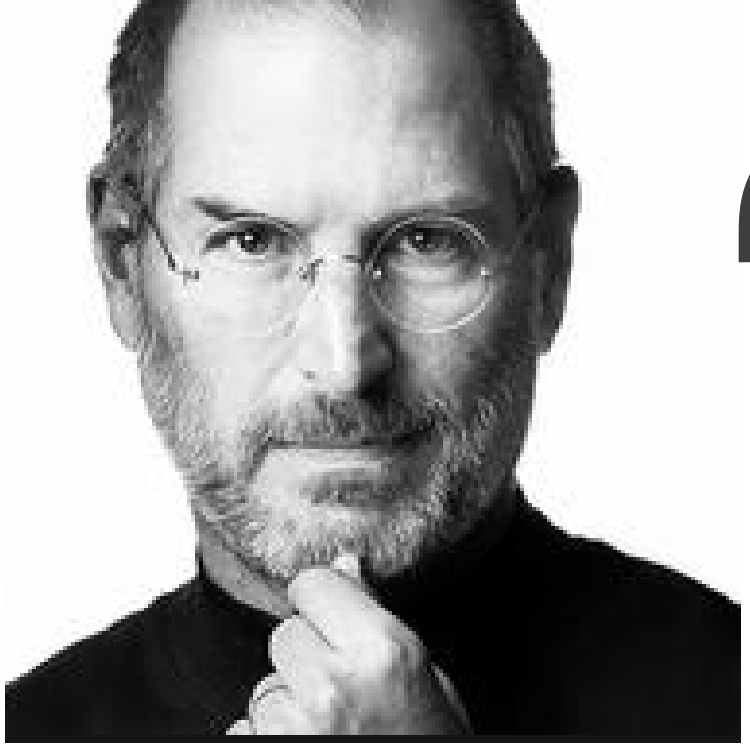
Code Challenge

GIC ISPVMF Global Retreat 2017

Yup, Everyone Can Code!



“Everybody in this country should learn how to program a computer…because it teaches you how to think.”

- Steve Jobs

“The core skills that programing brings out in every one of us is logical/analytical thinking and the ability to solve problems in a creative manner.”

- Hoe Yin

Aim

Basic Introduction To Coding

- Basic Python
- Web scraping

Most Importantly

HAVE FUN AND KEEP SMILING

Get to Know One Another Better

Plans

Appetiser

- 11:15-11:30 - Introduction
- 11:30-12:30 - Baby Python
- 12:30-1:00 - Lunch Break

Main Course

- 1:00-1:30 – Say Hello to ReBot
- 1:30-2:30 – Taming the Python
- 2:30-4:00 – Rebot Challenge
- 4:00-4:30 – Short Break

Dessert

- 4:30-5:15 – Idea Generation and Pitch preparation
- 5:15-5:30 – Move to 38 Auditorium
- 5:30-6:15 – Pitch, Awards and Closing

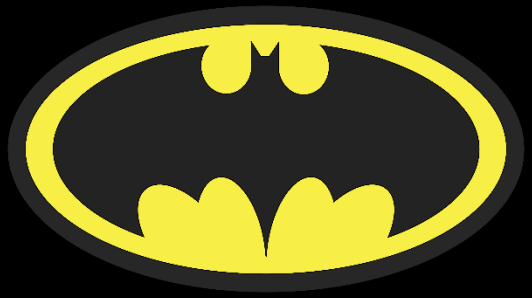
Get Ready!

Why Code?

10 minutes

CODING

IT MAY BE THE CLOSEST THING WE HAVE TO A SUPERPOWER



Stop. Think of Logical Blocks

Let’s get a computer to do things.

But,

- The machine is a fast worker but very stupid
  - “Please, make me a sandwich!”
  - Will this work?

    Step 1: Take a slice of bread, put it on the board

    Step 2: Apply a 1mm layer of butter on the bread

    Step 3: Take a slice of ham, put it on the bread
  - Probably closer to this,

    Step 1: Move arm to coordinates (49, 50, 0) with power 1W

    Step 2: Move arm down to (49, 50, -40) with power 1W

    Step 3: Tighten fingers with power 2W

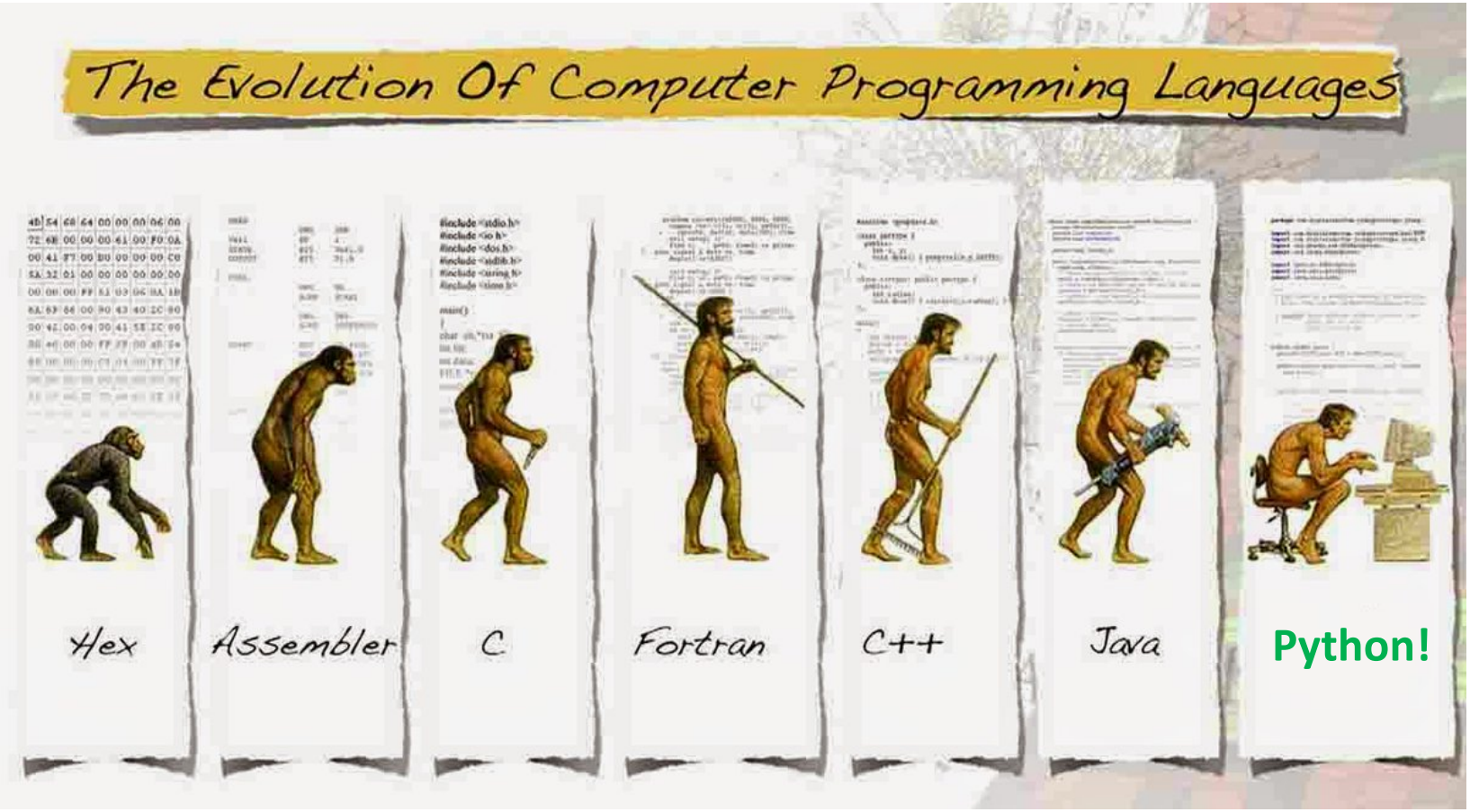
Python?

Uh.. Eating a snake?

45 minutes

What Exactly is Python?

- Considered a Fourth Generation Language
- Simple & mighty for mathematical libraries
- Relatively “natural” to read



Hello to Rebot Training!
Be Bold. Be Inspired.
Help bring new possibilities to conversational UX, create ideas to drive innovation!

Visit Rebot.Chat

Type rebot.chat in your web browser

Code the Future!

Lets get creative with Python. Python is a very simple programming language that can help you get started quickly!

Download the training notes here, and follow the lesson plan below.

Note: Are you still using IE? If so, please take 1 min to install Google Chrome to follow the lesson effectively.

Level 01: Baby Python

In this part of the course, you will learn to write and run basic python using a simple online python editor.

START PYTHON

Level 02: Hello Rebot

Here, you will write chatbot code using our integrated python editor which connects to our chatbot on Telegram.

TRY IT NOW!

Level 03: Taming Python

Next, let's learn basic HTML and webscrapping to extract data from websites, which are rich with information.

OK, LET'S GO!

Level 04: Rebot Challenge

Alright! Put all the code you have learnt together and help ReBot extract real-time Yahoo Finance data.

CHALLENGE ACCEPTED!

REPL: Basic Online Python Editor

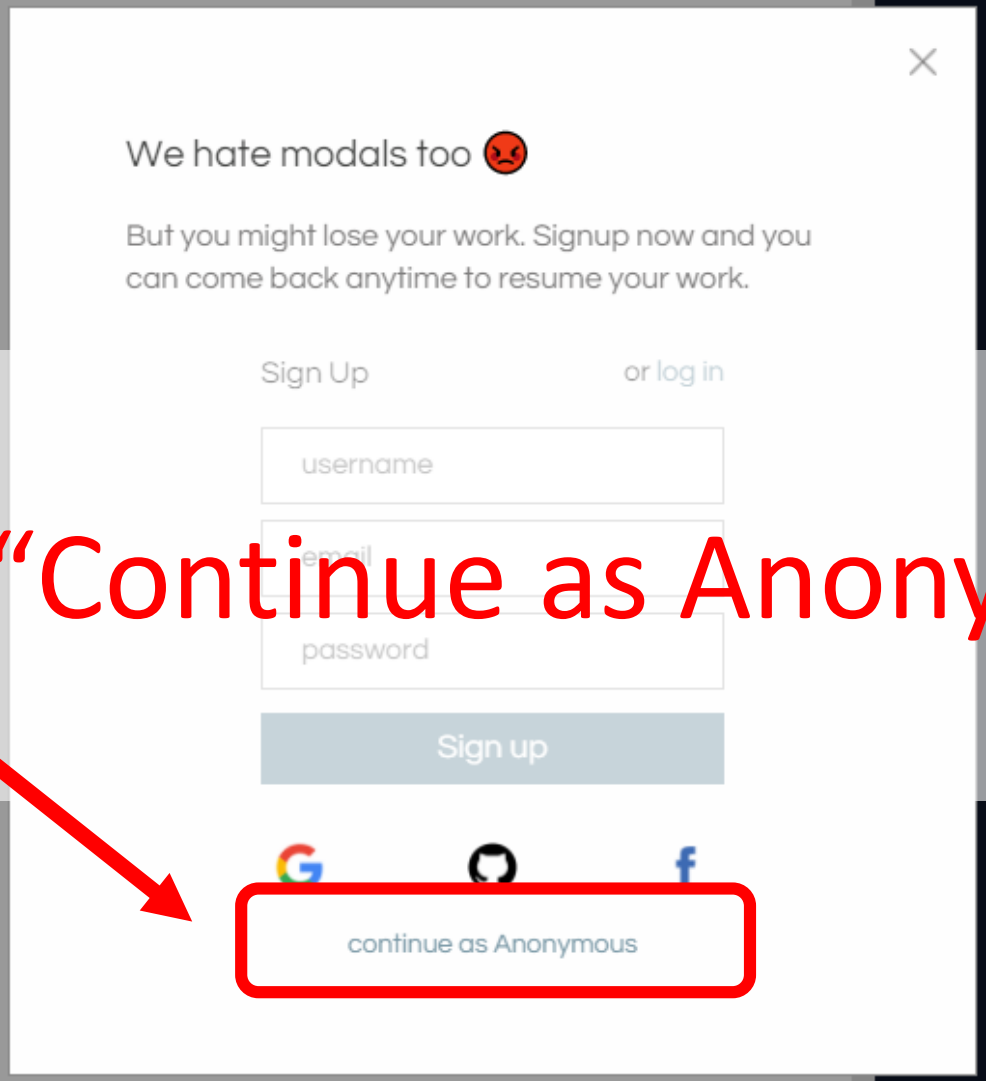
repl.it is **a cloud coding environment** for ES6

join a community of

engineers, teachers, and students

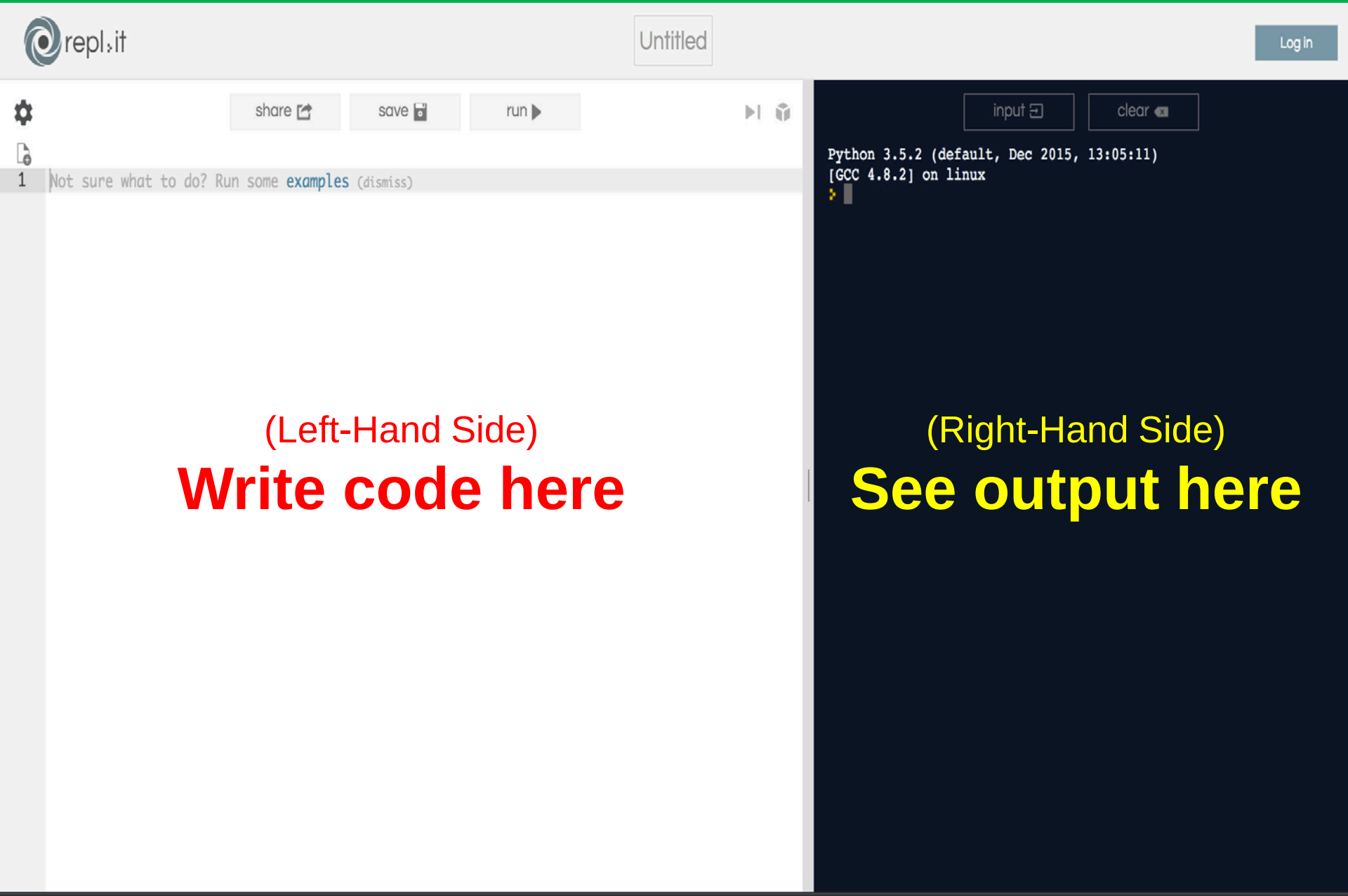
Select “Python3”

REPL: Basic Online Python Editor

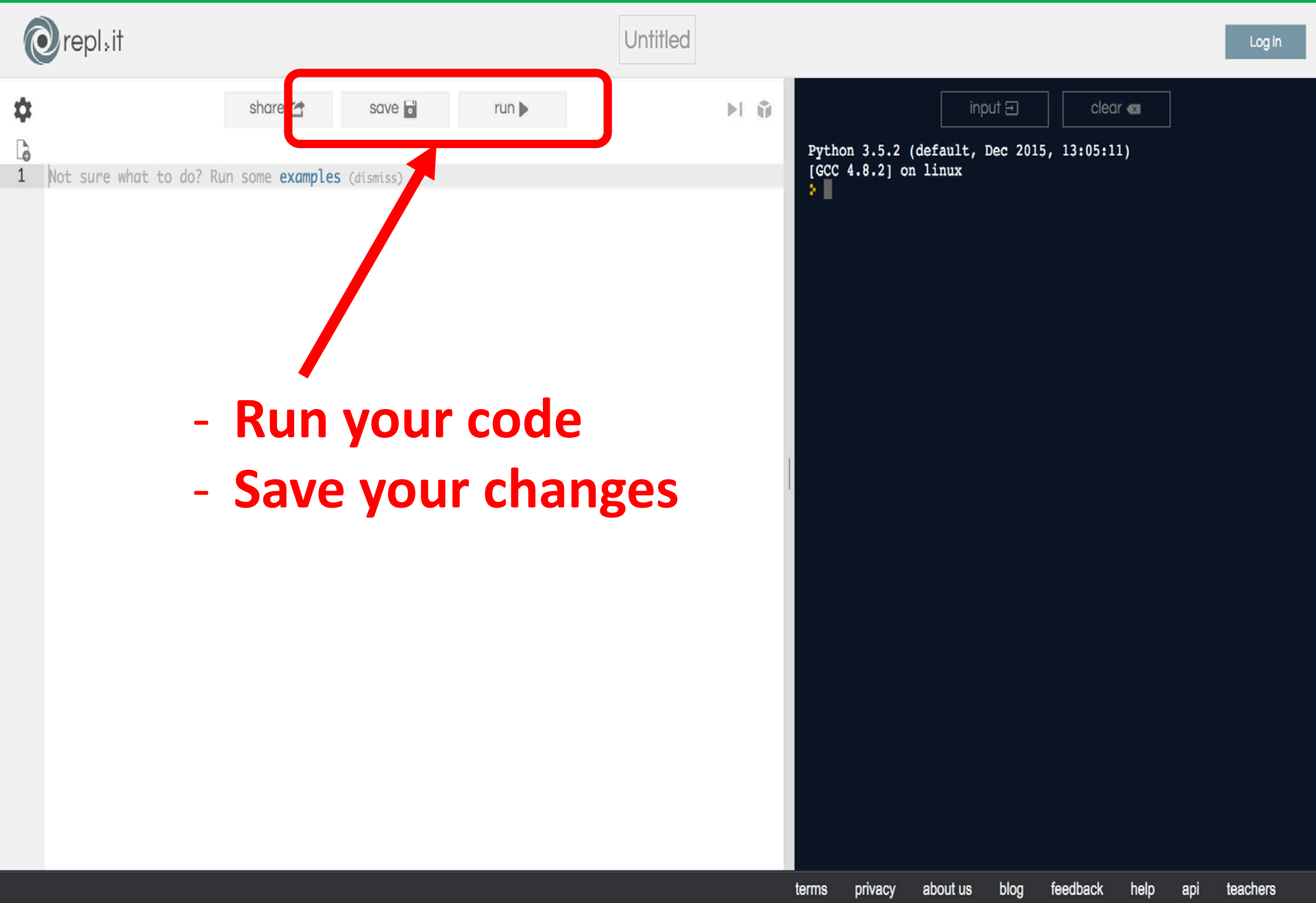


Select “Continue as Anonymous”

REPL: Basic Online Python Editor



REPL: Basic Online Python Editor



- **Run your code**
- **Save your changes**

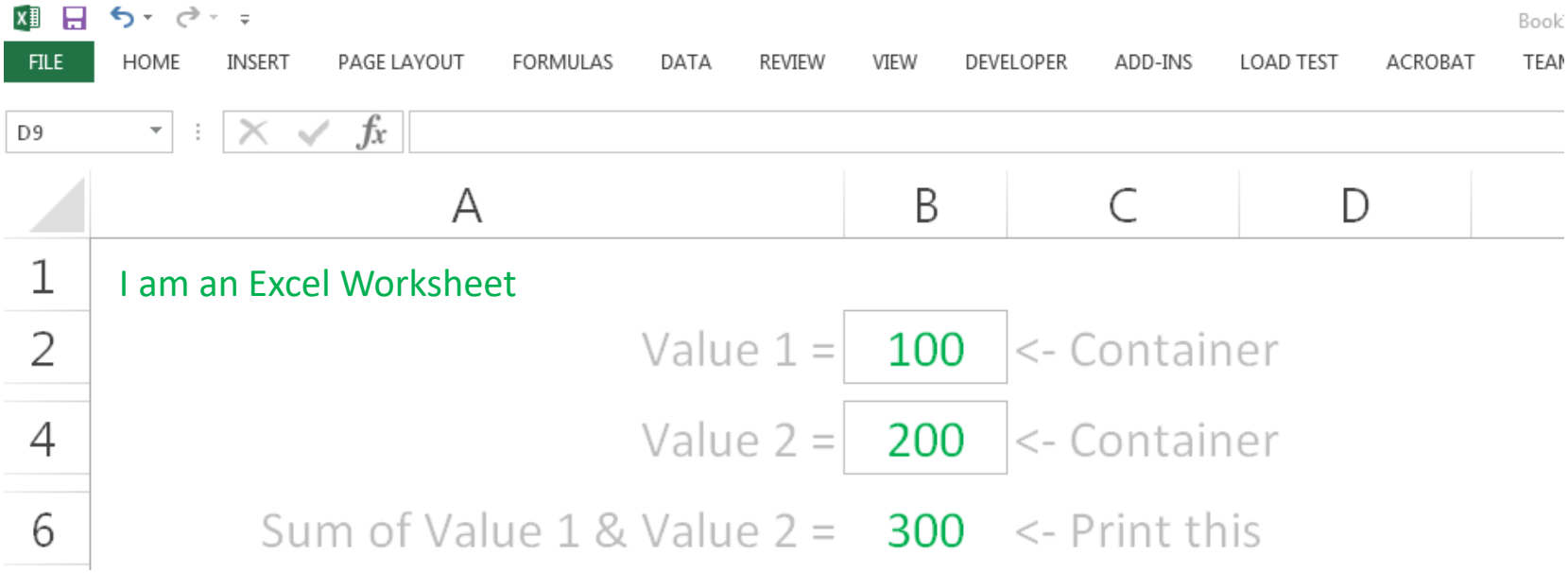
Real Code! Step-by-Step

- Appetizer
  - Print(“Hello World”)
  - Concept of variables (**containers**)
    - Create a string variable “name”
    - Print “Hello, my name is …”
- Main Course
  - Concept of variable types
    - String (text) – What is “1” + “1” ?
    - Integer (whole numbers) – What is 1 + 1 ?
    - Float (represented with decimals)
  - Concept of (IF .. THEN .. ELSE)
- Dessert
  - Recap + Free-to-play

Hey wait, isn't this familiar?

- Excel Formulas
  - Describe outcome with expressions
  - Cell = IF(SUM(A:A) > 1, SUM(A:A), "No")

- Excel Worksheet Cells
  - Think of "Cells" like "Containers"



Logical Design >> Code

- Coding easy to learn (And, can be fun ☺)
  - Following a recipe: EASY
  - Creating a recipe: HARD
- Given set of values: 5, 9, 10, 4, 11, 3, 6
  - Highest and lowest number?
  - What's your recipe?
- Don't just translate your work procedures
  - Simplify and refine it.

Golden Quotes

“The core skills that programing brings out in every one of us is logical/analytical thinking and the ability to solve problems in a creative manner”

- Hoe Yin

Lunch-Break

12.30pm – 1.00pm

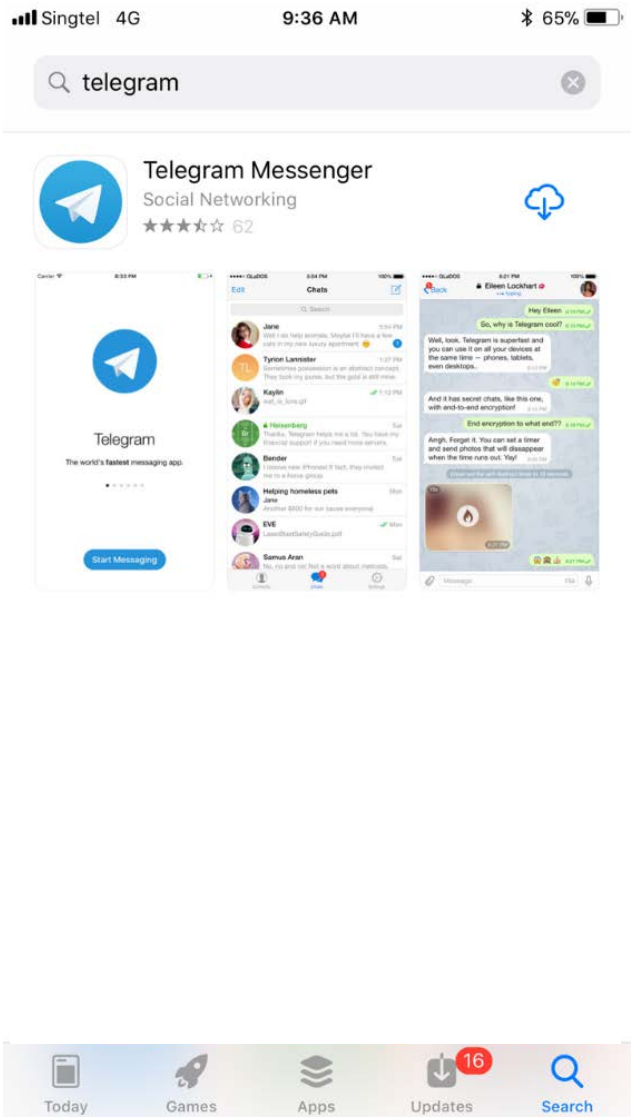
30 minutes

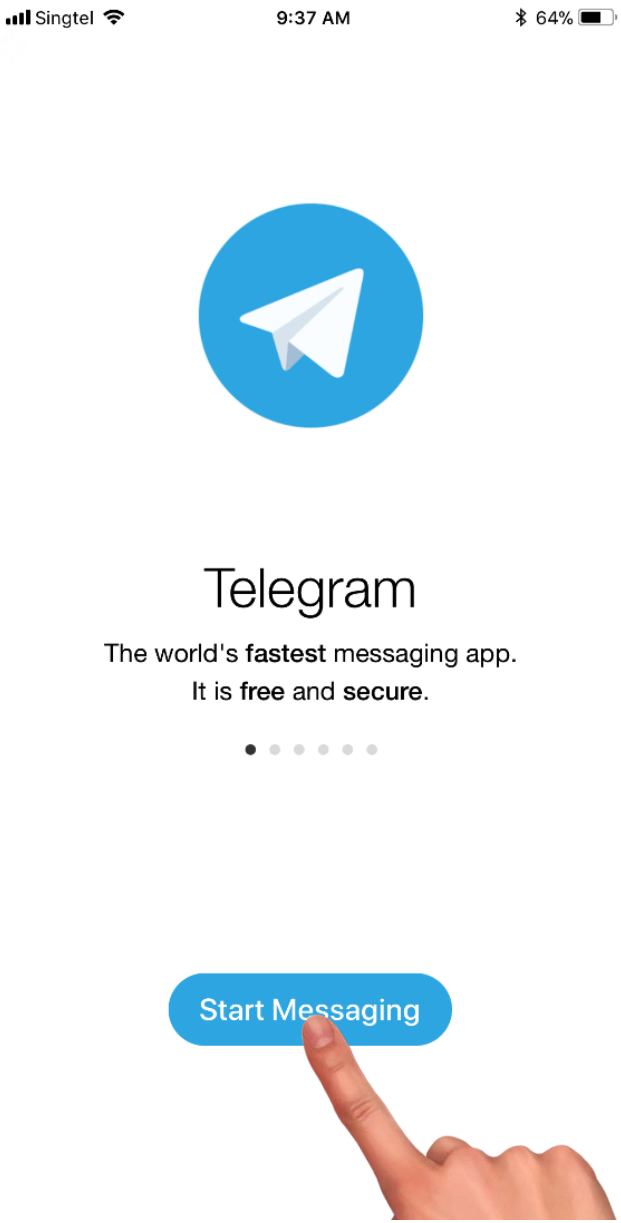
Step 1: Install Telegram

Telegram

Download on the App Store

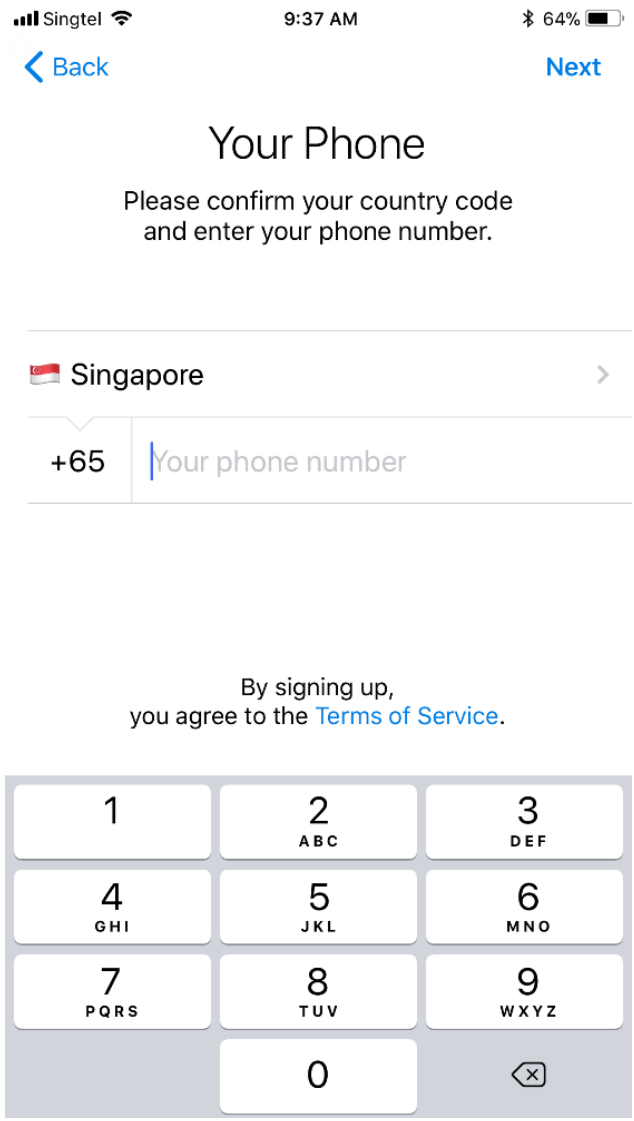
ANDROID APP ON Google Play

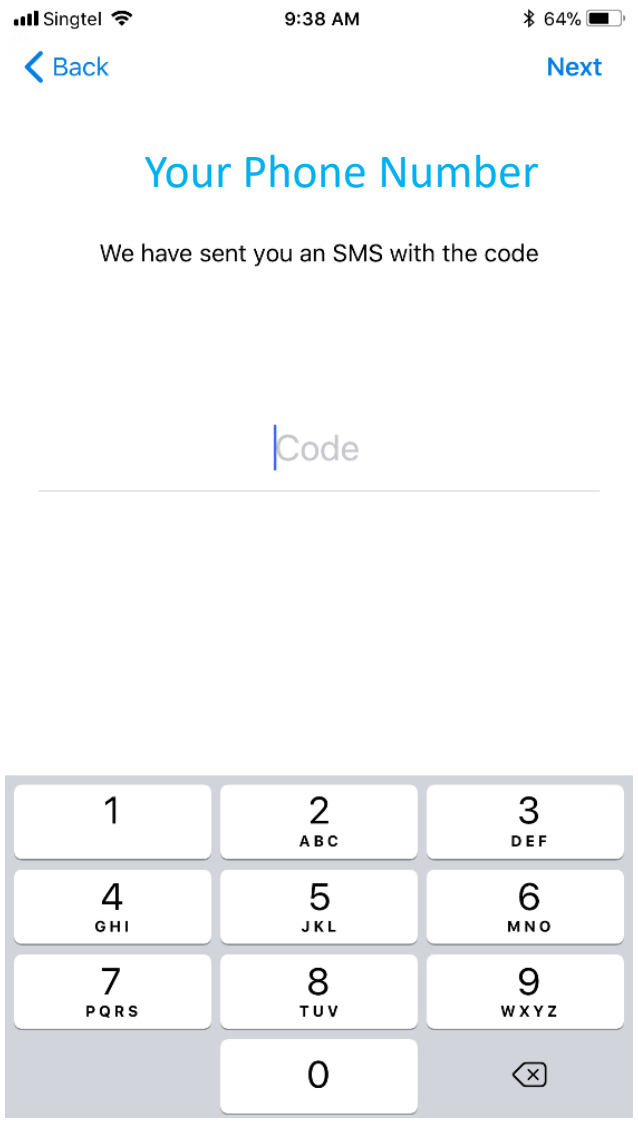




Step 2: Register Account

- Register for an account
- Wait for an SMS passcode
- Key in the passcode

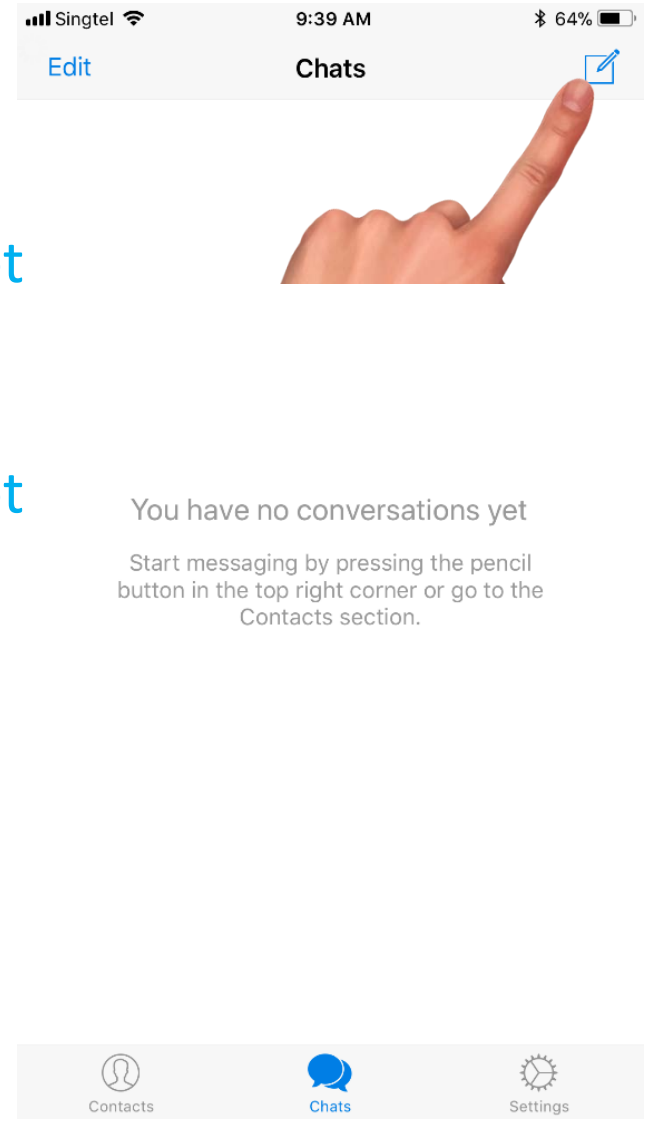


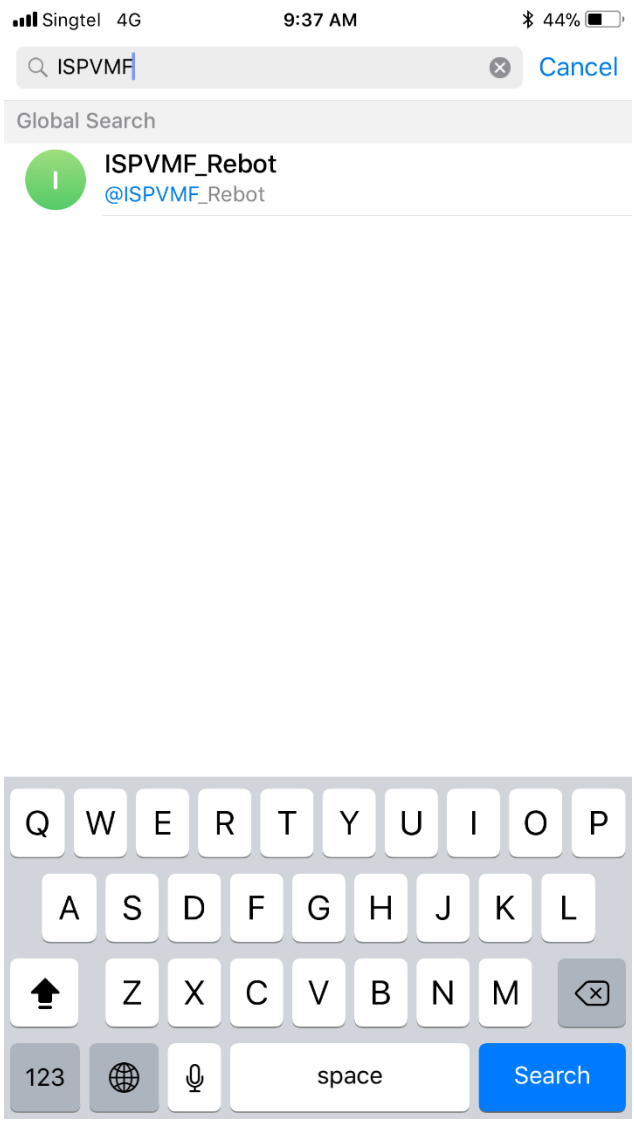


Step 3: Add @ISPVMF_ReBot

Install Telegram

- Add a Chat
- Search for @ISPVMF_Rebot
- Select @ISPVMF_Rebot





Rebot

Putting it Together For the First Time

30 minutes

Code the Future!

Lets get creative with Python. Python is a very simple programming language that can help you get started quickly!

Download the **training notes** here, and follow the lesson plan below.

Note: Are you still using IE? If so, please take 1 min to install Google Chrome to follow the lesson effectively.

Level 01: Baby Python

In this part of the course, you will learn to write and run basic python using a simple online python editor.

START PYTHON

Level 02: Hello Rebot

Here, you will write chatbot code using our integrated python editor which connects to our chatbot on Telegram.

TRY IT NOW!

Level 03: Taming Python

Next, let's learn basic HTML and webscrapping to extract data from websites, which are rich with information.

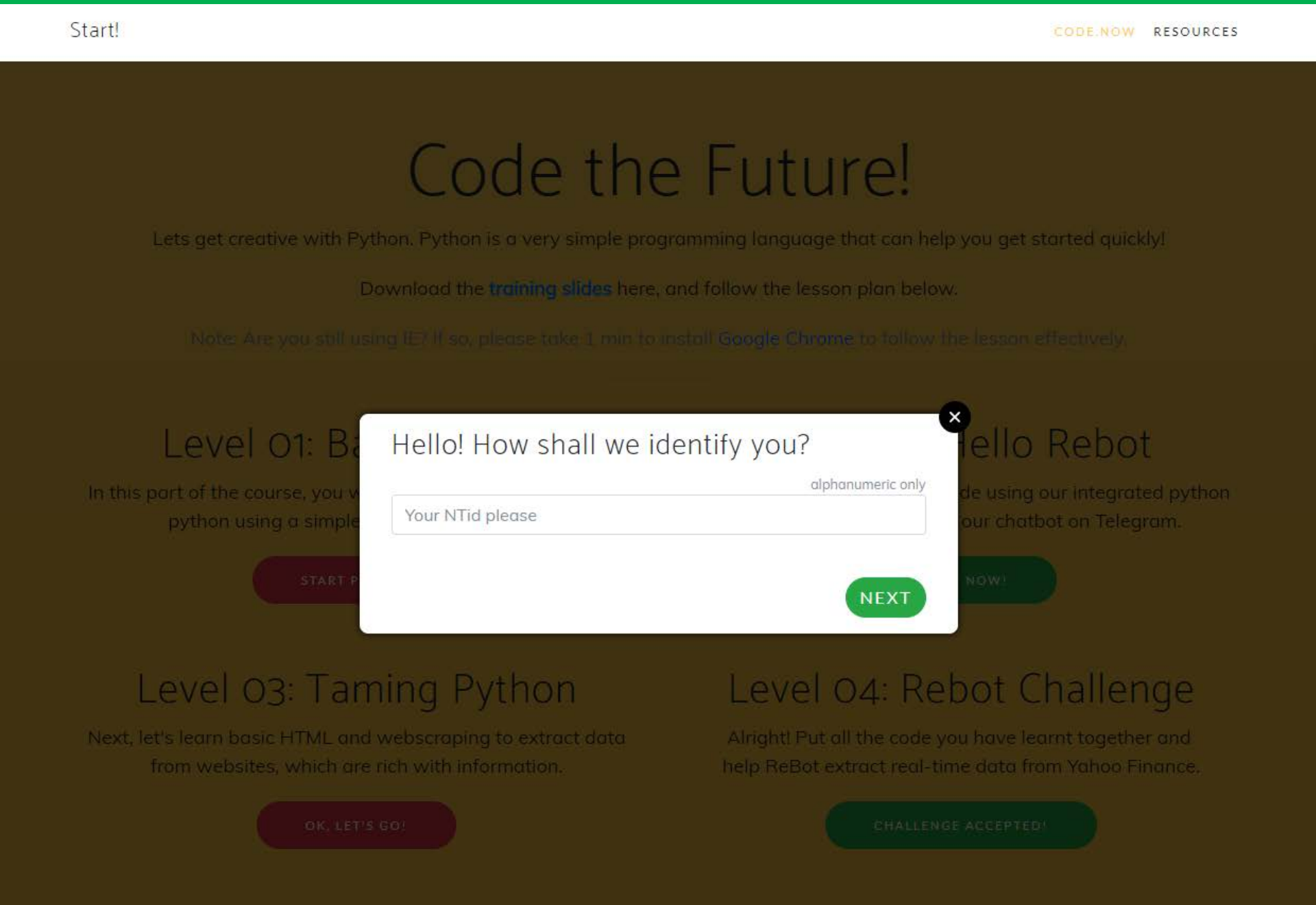
OK, LET'S GO!

Level 04: Rebot Challenge

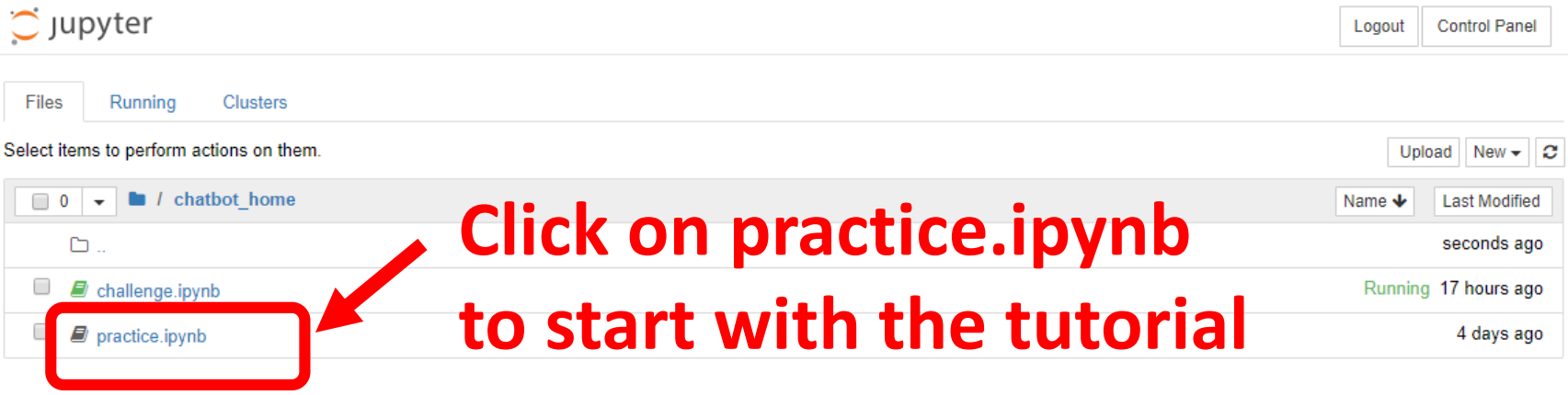
Alright! Put all the code you have learnt together and help ReBot extract real-time Yahoo Finance data.

CHALLENGE ACCEPTED!

Choose an identity (alphanumeric)



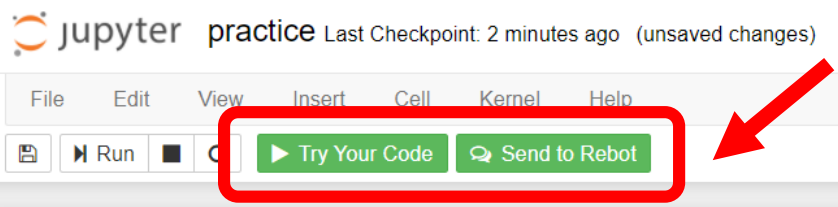
Jupyter: Integrated Python/Rebot



Jupyter has been integrated with **@ISPVMF_Rebot** on Telegram

Next, lets see how your code changes on Jupyter appear magically (hopefully) on your mobile phone

Jupyter: Integrated Python Editor



Two Buttons

- Try Your Code
- Send To Rebot

Practice

In this practice, you will learn to write python code to reply to your chatbot conversations on Telegram.

Use the green "Send to Rebot" button to see your changes on Rebot via Telegram.

Pay attention to the instructor.

Have fun! :)

```
import rebot

def send_to_rebot(ticker):

    message = rebot.createEmptyMessage()
    message.addText("Hello from my first chatbot code!")

    return message
```

Write your code in this grey box area

Try it Out!

Test your changes by pressing the green "Try Your Code" button above!

```
print(send_to_rebot("Hello")) #j_ignore_
```

Let the Magic Happen

- Appetizer
  - Return “Hello World”
  - Return “Your Name”
- Main Course
  - Lets make our chat bot smarter.
  - Return Sum (1 + 1)
  - Return (IF .. THEN .. ELSE)
- Dessert
  - Return random results.
  - What is the day today?
  - What is your name?
  - I am bored?

Web-scraping

Scrape Data From HTML

60 minutes

Code the Future!

Lets get creative with Python. Python is a very simple programming language that can help you get started quickly!

Download the training notes here, and follow the lesson plan below.

Note: Are you still using IE? If so, please take 1 min to install Google Chrome to follow the lesson effectively.

Level 01: Baby Python

In this part of the course, you will learn to write and run basic python using a simple online python editor.

START PYTHON

Level 02: Hello Rebot

Here, you will write chatbot code using our integrated python editor which connects to our chatbot on Telegram.

TRY IT NOW!

Level 03: Taming Python

Next, let's learn basic HTML and webscrapping to extract data from websites, which are rich with information.

OK, LET'S GO!

Level 04: Rebot Challenge

Alright! Put all the code you have learnt together and help ReBot extract real-time Yahoo Finance data.

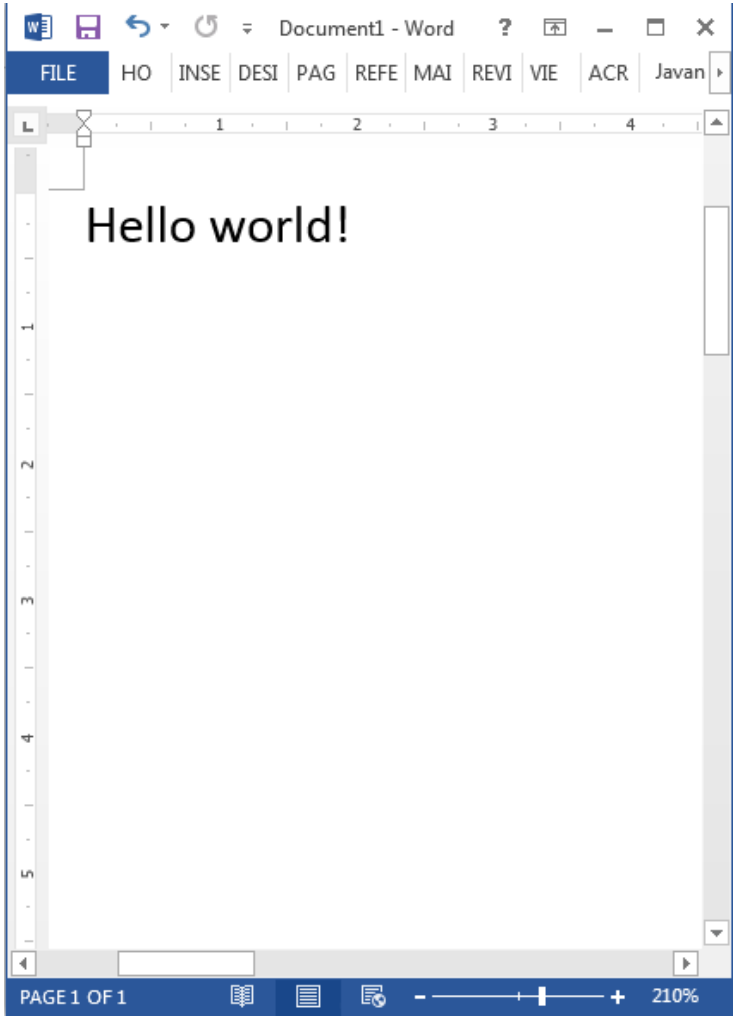
CHALLENGE ACCEPTED!

What is HTML?

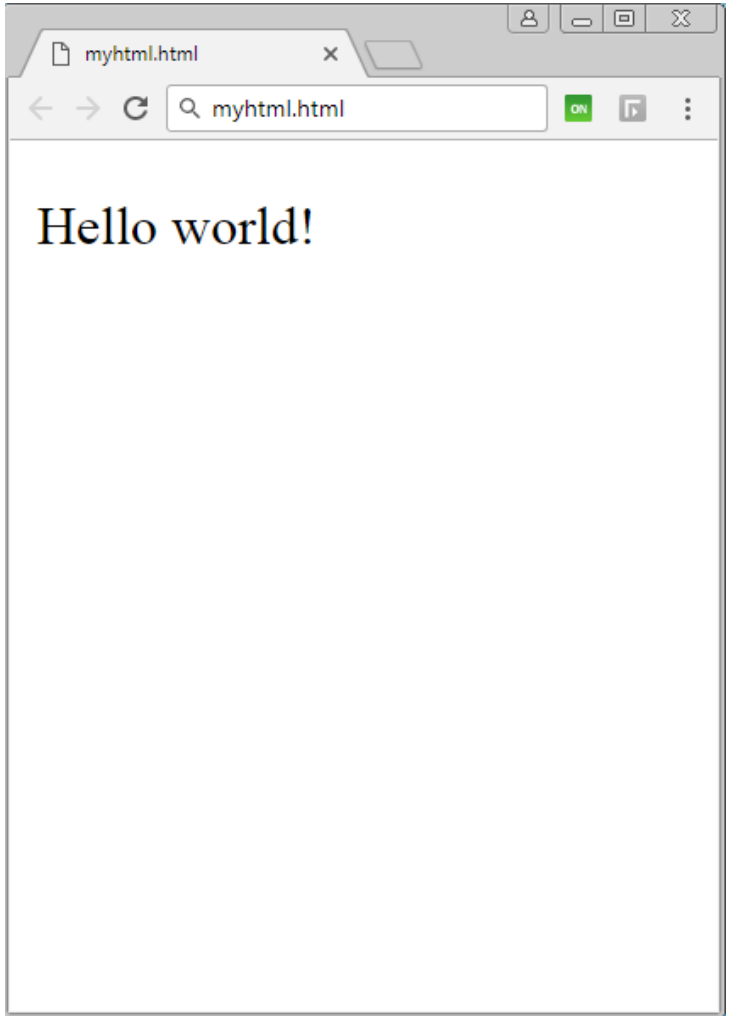
- HTML is
  - The building block of websites.
  - Source code of the Web
- Concept of HTML Tags
  - Open tags <HTML>
  - Close tags </HTML>
- Let’s see how to scrape data from HTML

Word Doc vs HTML Doc

Word Document

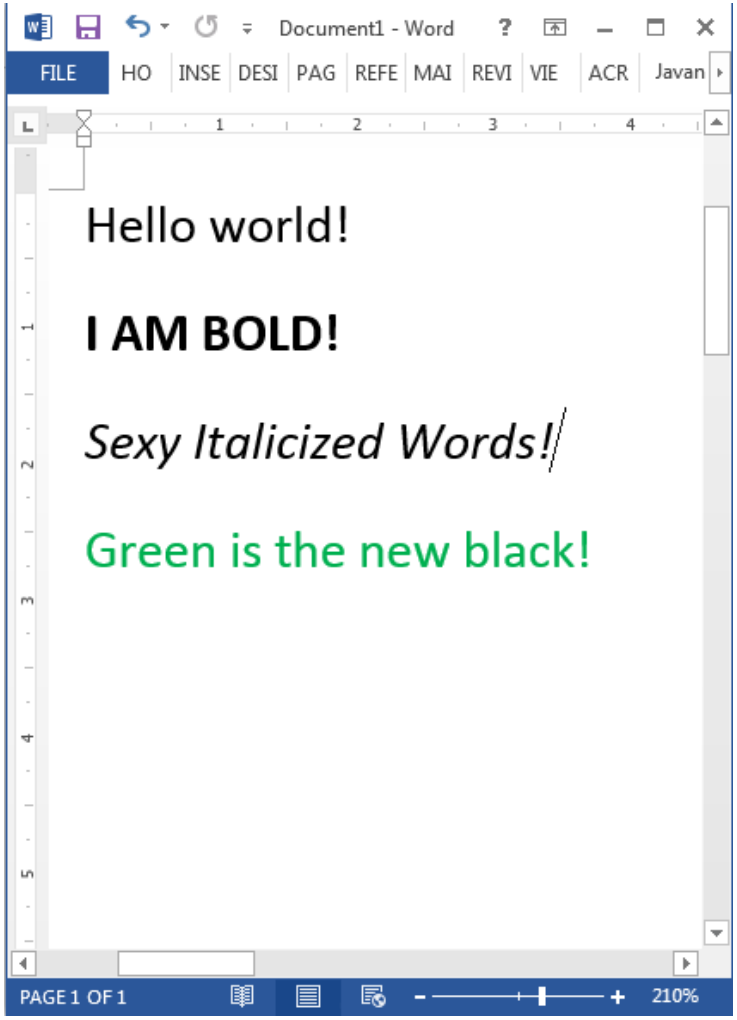


HTML Document



Word Doc vs HTML Doc

Word Document



HTML Document

HTML - Code Visualization

HTML - NESTED CONTAINERS

Container (HTML)

Container (BODY)

Container (P)

HTML - CODE BLOCKS

```
<HTML>

<BODY>

<P CLASS="Text">
Hello World!
</P>

</BODY>

</HTML>
```

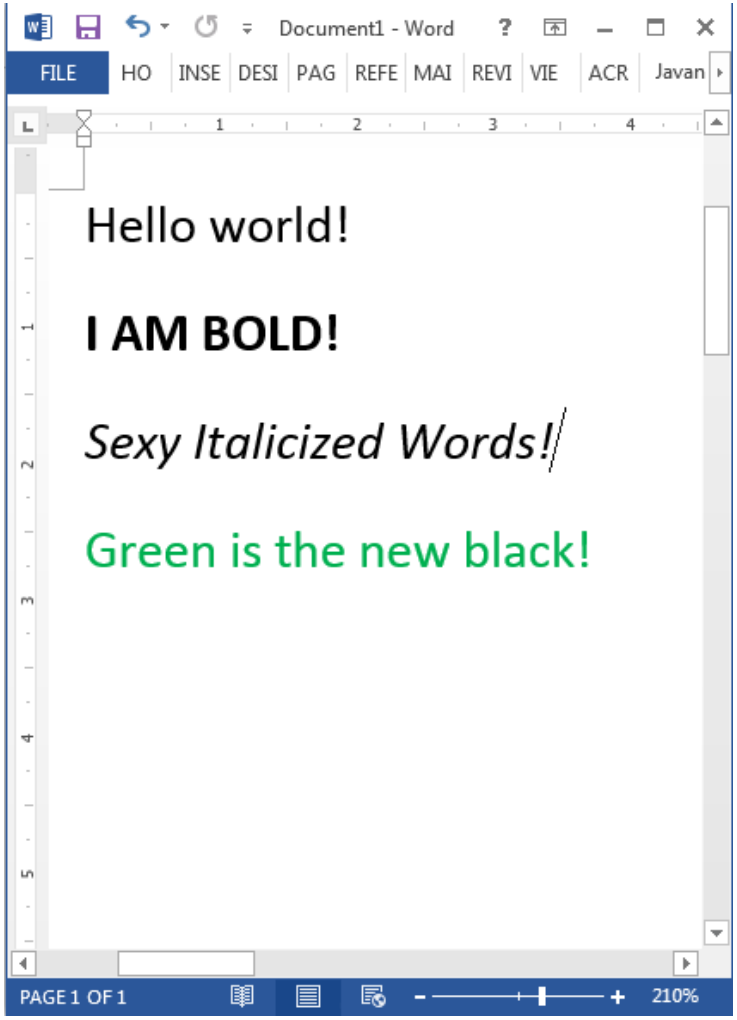
HTML - RAW CODE

```
<HTML>

<BODY>

<P CLASS="Text">
Hello World!
</P>

</BODY>

</HTML>
```

- These values are enclosed in tags (think of containers).
- They are the basic building blocks needed used to build basic to complex websites.

Word Doc vs HTML Doc

Word Document



HTML Document

Word Doc vs HTML Doc

Hey! HTML is just made up of TAGs

```
<html>
  <body>
    <p>                    Hello World!                    </p>
    <p>                  <b>I AM BOLD!</b>                 </p>
    <p>                <i>Sexy Italicized Words!</i>       </p>
    <p>        <font color="green">Green is the new black!</font>  </p>
  </body>
</html>
```

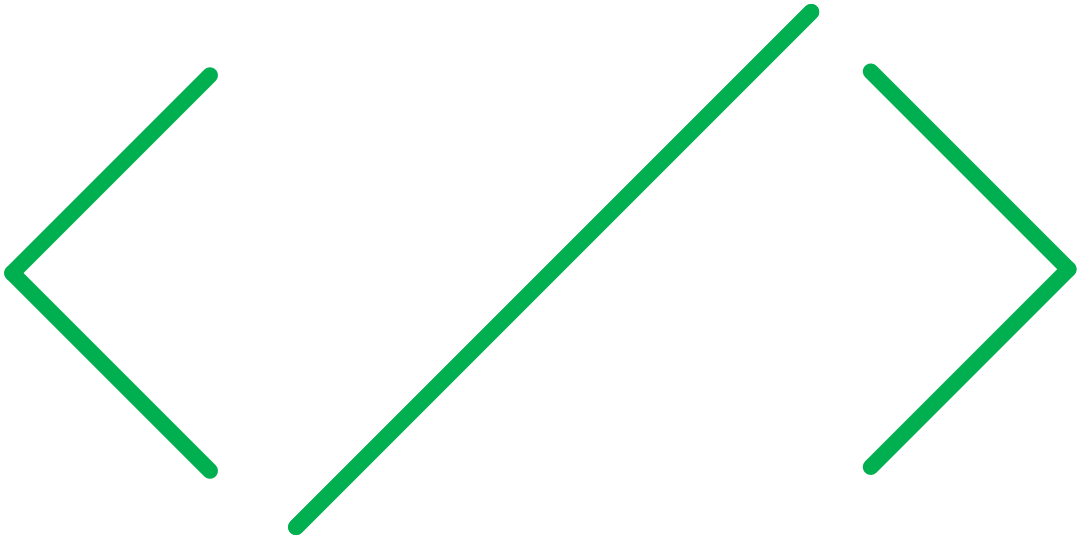
Think of Tags as Containers

Think of a

HTML TAG

As a

CONTAINER



Think of Tags as Containers

Think of a

HTML TAG

As a

CONTAINER

I AM A VALUE

So many types of HTML tags…

HTML 5 NEW TAG

TAG NOT SUPPORTED IN HTML 5

<!--...-->	Define a comment
<!DOCTYPE>	Defines the document type
<a>	Defines a hyperlink href, hreflang, media, ping , rel, target, type
<abbr>	Defines an abbreviation
<acronym>	Used to define an embedded acronyms
<address>	Defines an address element
<applet>	Used to define an embedded applet
<area>	Defines an area inside an image map alt, coords, href, hreflang, media, ping, rel, shape, target, type
<article>	Defines an article cite, pubdate
<aside>	Defines content aside from the page content
<audio>	Defines sound content autobuffer, autoplay, controls, src
<b>	Defines bold text
<base>	Defines a base URL for all the links in a page href, target
<basefont>	Used to define a default font-color, font-size, or font-family for all the document
<bdo>	Defines the direction of text display dir
<big>	Used to make text bigger
<blockquote>	Defines a long quotation cite
<body>	Defines the body element
 	Inserts a single line break
<button>	Defines a push button autofocus, disabled, form, formaction, formenctype, formmethod, formnovalidate, formtarget, name, type, value
<canvas>	Defines graphics height, width
<caption>	Defines a table caption
<cite>	Defines a citation
<code>	Defines computer code text autobuffer, autoplay, controls, src
<col>	Defines attributes for table columns
<colgroup>	Defines groups of table columns span
<command>	Defines a command button checked, disabled, icon, label, radiogroup, type
<datalist>	Defines a dropdown list
<dd>	Defines a definition description
<del>	Defines deleted text cite, datetime
<details>	Defines details of an element open
<dialog>	Defines a dialog (conversation)
<dfn>	Defines a definition term
<dir>	Used to define a directory list
	Defines a section in a document
<dl>	Defines a definition list
<dt>	Defines a definition term
	Defines emphasized text
<embed>	Defines external interactive content or plugin height, src, type, width
<fieldset>	Defines a fieldset disabled, form, name
<figure>	Defines a group of media content, and their caption
<font>	Used to define font face, font size, and font color of text
<footer>	Defines a footer for a section or page
<form>	Defines a form accept-charset, action, autocomplete, enctype, method, name, novalidate, target
<frame>	Used to define one particular window (frame) within a frameset
<frameset>	Used to define a frameset, which organized multiple windows (frames)
<h1> to <h6>	Defines header 1 to header 6
<head>	Defines information about the document
<header>	Defines a header for a section or page
<hgroup>	Defines information about a section in a document
<hr>	Defines a horizontal rule
<html>	Defines an html document manifest, xmlns
<i>	Defines italic text
<iframe>	Defines an inline sub window height, name, sandbox, seamless, src, width
<img>	Defines an image alt, src, height, ismap, usemap, width
<input>	Defines an input field accept, alt, autocomplete, autofocus, checked, disabled, form, formaction, formenctype, formmethod, formnovalidate, formtarget, height, list, max, maxlength, min, multiple, name, pattern, placeholder, readonly, required, size, src, step, type, value, width
<ins>	Defines inserted text cite, datetime
<keygen>	Defines a generated key in a form autofocus, challenge, disabled, form, keytype, name
<kbd>	Defines keyboard text
<label>	Defines an inline sub window for, form
<legend>	Defines a title in a fieldset
	Defines a list item value
<link>	Defines a resource reference href, hreflang, media, rel, sizes, type
<map>	Defines an image map name
<mark>	Defines marked text
<menu>	Defines a menu list label, type
<meta>	Defines meta information charset, content, http-equiv, name
<meter>	Defines measurement within a predefined range high, low, max, min, optimum, value
<nav>	Defines navigation links
<noframes>	Used to display text for browsers that do not handle frames
<noscript>	Defines a noscript section
<object>	Defines an embedded object data, form, height, name, type, usemap, width
	Defines an ordered list reversed, start
<optgroup>	Defines an option group label, disabled
<option>	Defines an option in a drop-down list disabled, label, selected, value
<output>	Defines some types of output for, form, name
<p>	Defines a paragraph
<param>	Defines a parameter for an object name, value
<pre>	Defines preformatted text
<progress>	Defines progress of a task of any kind max, value
<q>	Defines a short quotation cite
<rp>	Used in ruby annotations to define what to show browsers that to not support the ruby element
<rt>	Defines explanation to ruby annotations
<ruby>	Defines ruby annotations
<s>, <strike>	Used to define strikethrough text
<samp>	Defines sample computer code
<script>	Defines a definition list async, type charset defer, src
<section>	Defines a section cite
<select>	Defines a selectable list autofocus, disabled, form, multiple, name, size
<small>	Defines small text
<source>	Defines media resources media, src, type
	Defines a section in a document
	Defines strong text
<style>	Defines a style definition type, media, scoped
<sub>, <sup>	Defines sub/super-scripted text
<table>	Defines a table summary
<tbody>	Defines a table body summary
<td>	Defines a table cell colspan, headers, rowspan
<textarea>	Defines a text area autofocus, cols, disabled, form, maxlength, name, placeholder, readonly, readonly, required, rows, wrap
<tfoot>, <thead>	Defines a table footer / head
<th>	Defines a table header colspan, headers, rowspan, scope
<time>	Defines a date/tim datetime
<title>	Defines the document title
<tr>	Defines a table row datetime
<tt>	Used to define teletype text
<u>	Used to define underlined text
	Defines an unordered list
<var>	Defines a variable
<video>	Defines a video autobuffer, autoplay, controls, height, loop, src, width

HTML5 TAG CHEAT SHEET

Created by WebsiteSetup.org

For Illustration Purposes Only

Useful Tags for Today

Never fear! Only need these today.

`<html>`	`</html>`
`<body>`	`</body>`
`<p>`	`</p>`
`<span>`	`</span>`
`<td>`	`</td>`
`<div>`	`</div>`

Code the Future!

Lets get creative with Python. Python is a very simple programming language that can help you get started quickly!

Download the training notes here, and follow the lesson plan below.

Note: Are you still using IE? If so, please take 1 min to install Google Chrome to follow the lesson effectively.

Level 01: Baby Python

In this part of the course, you will learn to write and run basic python using a simple online python editor.

START PYTHON

Level 02: Hello Rebot

Here, you will write chatbot code using our integrated python editor which connects to our chatbot on Telegram.

TRY IT NOW!

Level 03: Taming Python

Next, let's learn basic HTML and webscrapping to extract data from websites, which are rich with information.

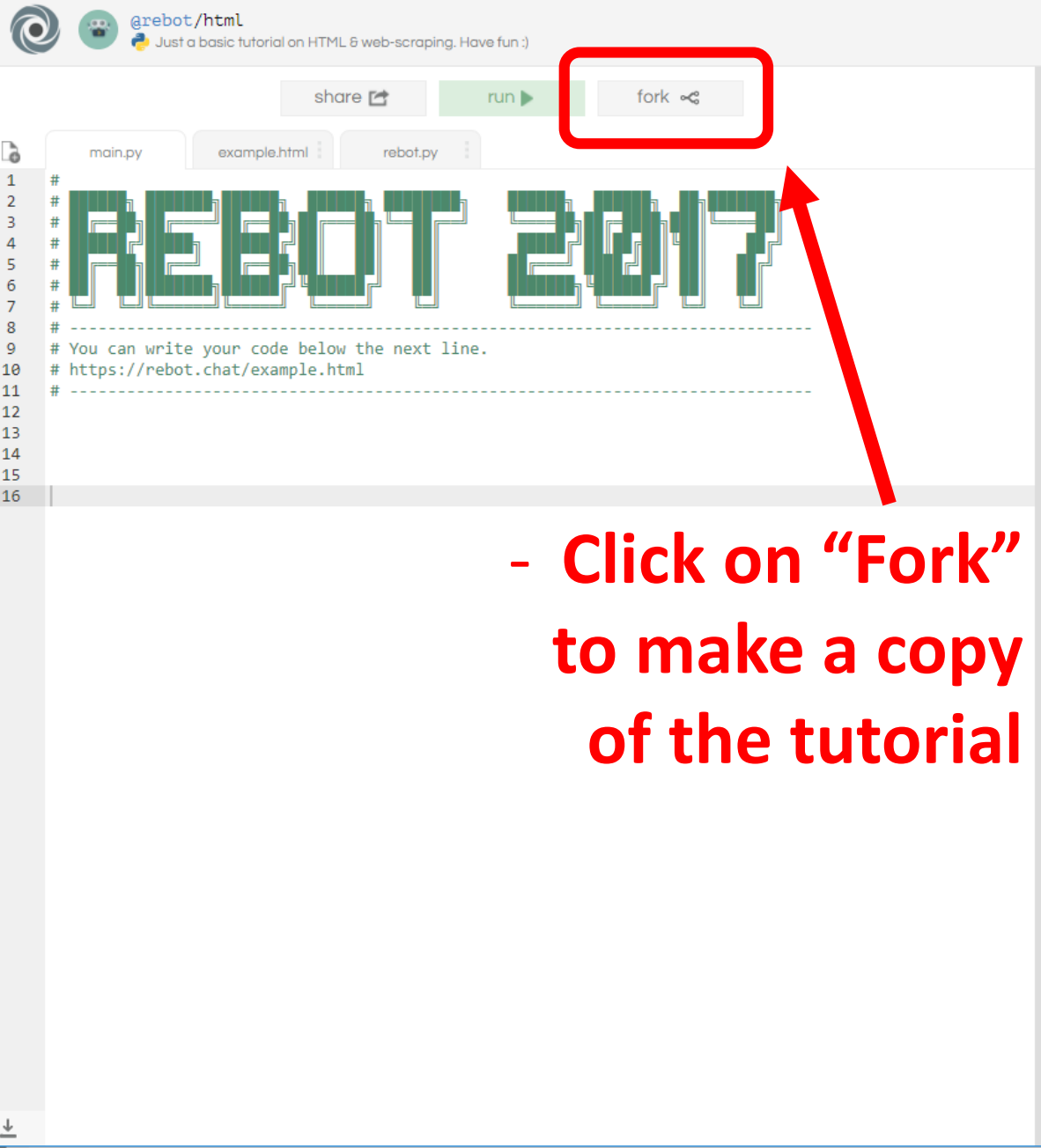
OK, LET'S GO!

Level 04: Rebot Challenge

Alright! Put all the code you have learnt together and help ReBot extract real-time Yahoo Finance data.

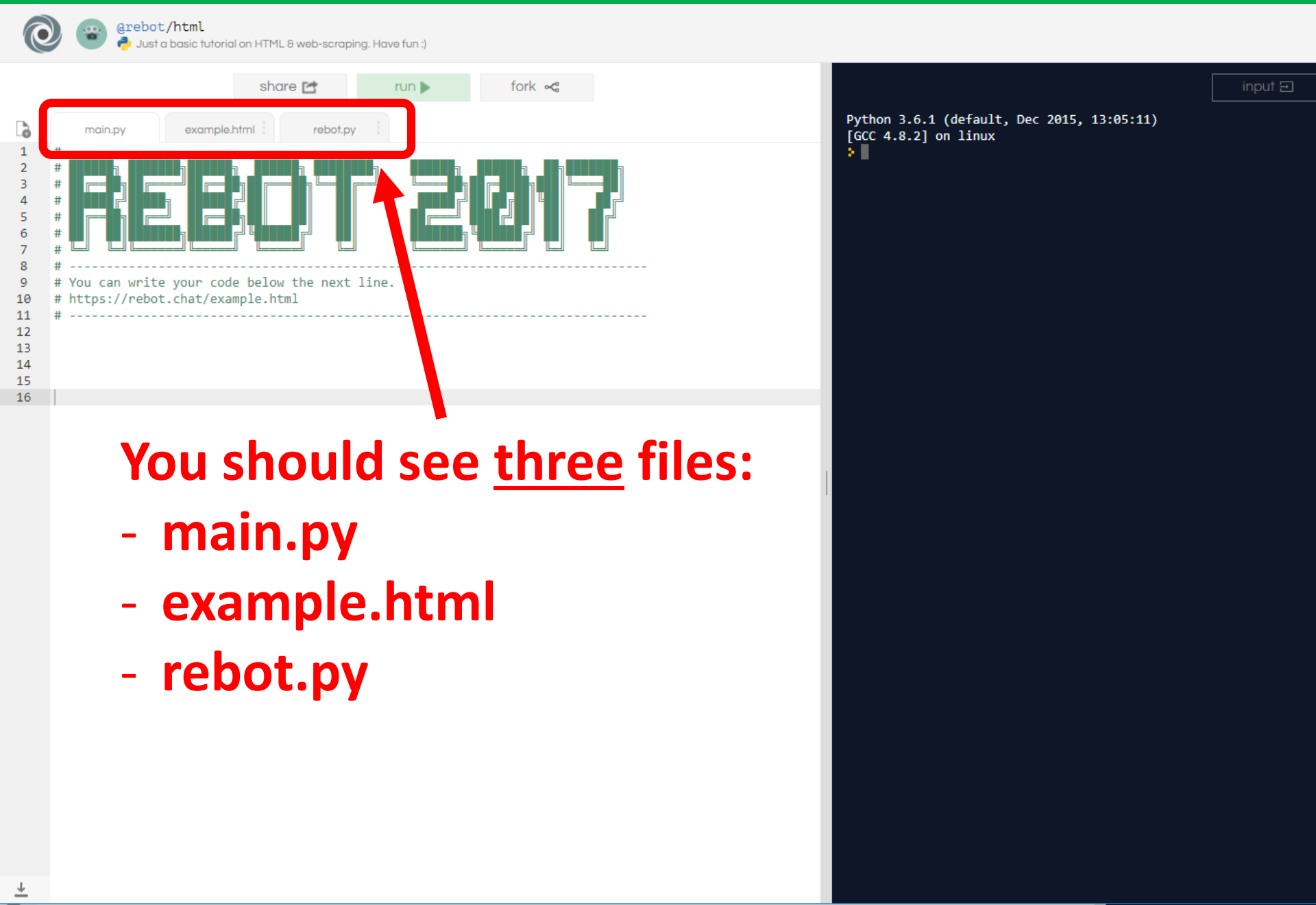
CHALLENGE ACCEPTED!

Let's Give it a Try



- **Click on "Fork" to make a copy of the tutorial**

Let's Give it a Try



Let’s Give it a Try

Step 1: Import libraries

Coders in the community can create libraries of commonly used features. These templates are pre-defined and can help us simplify and reuse code.

Step 2: What is the document filename?

For example. “Quarterly reporting.doc”, “Mydocument.html”, http://rebot.chat/index.html.

Step 3: Get the content within the document

Extract the file content as bytecode literals directly from the document.

Step 4: Format the content into a proper HTML format

Creates a clean HTML format of the content, which is then readable by the extraction logic.

Step 5: Find the data-point we need

Use the rebot library to search for the container that we need, including any specific class properties into the search criteria

Let's Give it a Try


```
# Step 1: Tell python that you want to use rebot library.
import rebot

# Step 2: Keep the filename in a variable (ie. container)
filename = "example.html"

# Step 3: Ask rebot to give you the contents in the file
content = rebot.getContent(filename)

# Step 4: Ask rebot to convert the content to proper HTML
html = rebot.getHtml(content)

# Step 5: Search for the container in the HTML
data = html.search("span", "class", "target")

# Finally! Show the data
print(data)
```

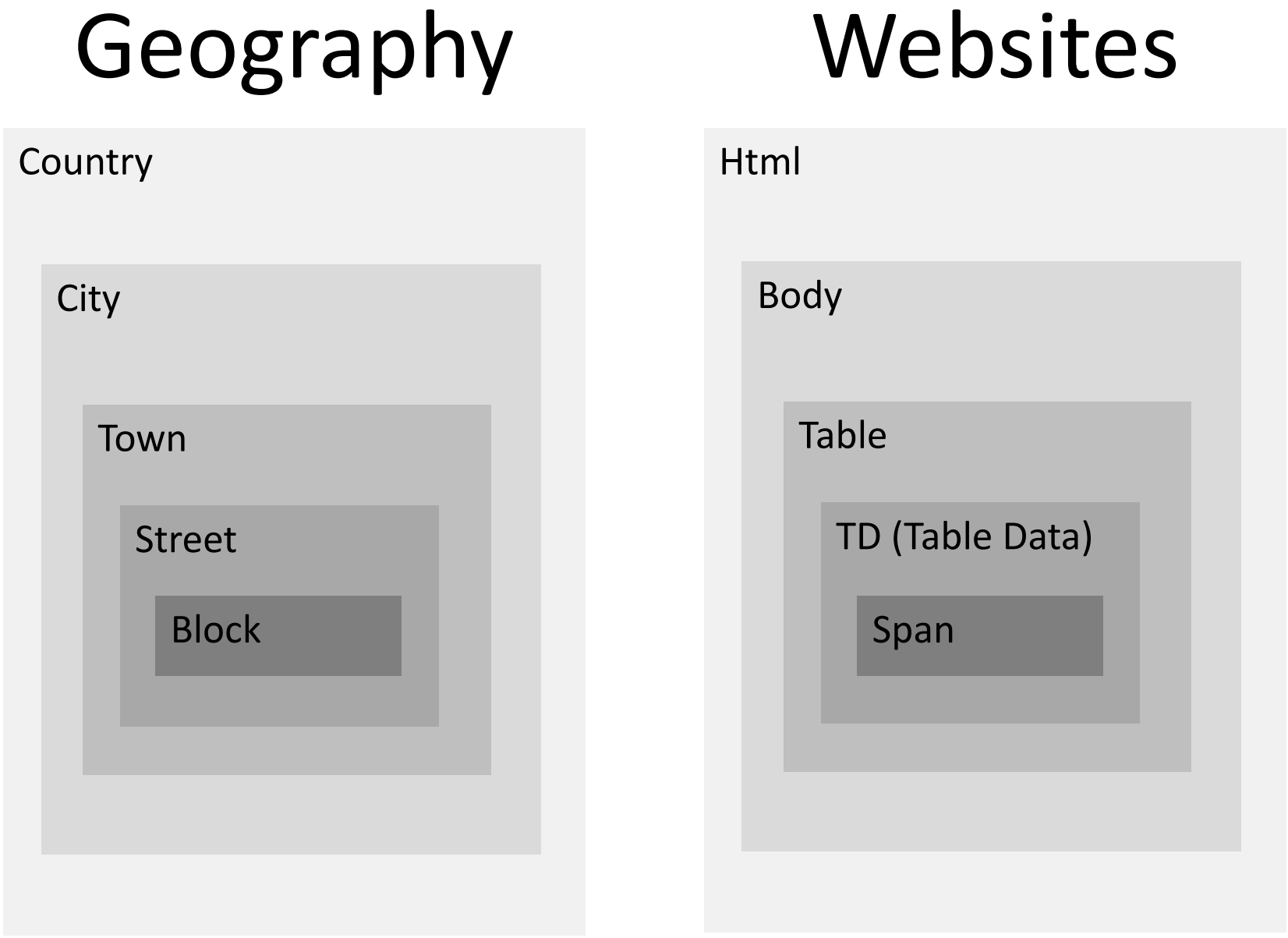
Often, its not so easy..

Example: HTML code behind Yahoo.com

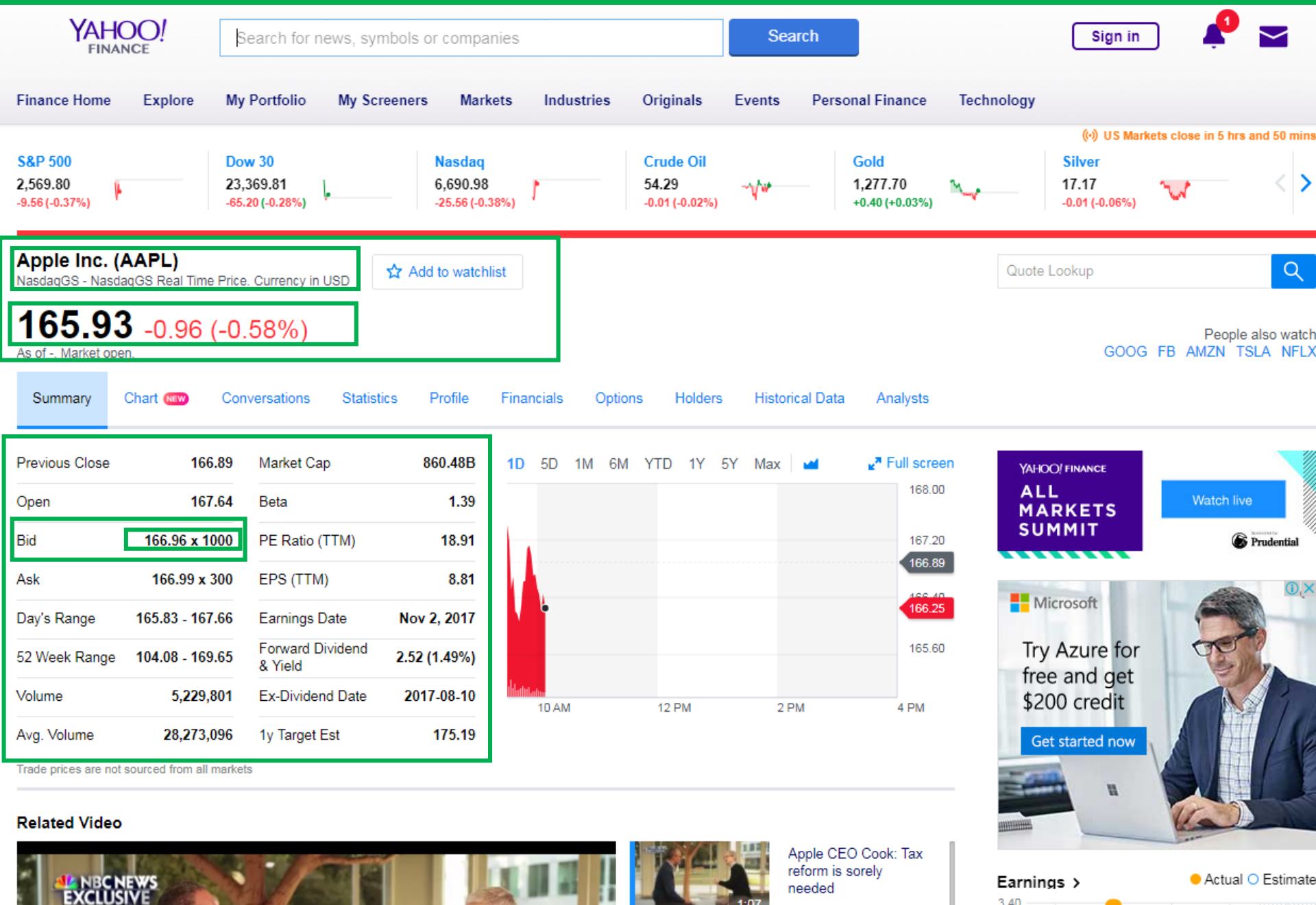
```
<!DOCTYPE html><html id="atomic" class="NoJs chrome desktop" lang="en-US"><head prefix="og: http://ogp.me/ns#"><s
for Apple Inc. - Yahoo Finance</title><meta name="keywords" content="AAPL, Apple Inc., AAPL stock chart, Apple In
content="on"/><meta property="twitter:site" content="@YahooFinance"/><meta property="fb:app_id" content="90376669
lang="en-US" content="View the basic AAPL stock chart on Yahoo Finance. Change the date range, chart type and com
rel="dns-prefetch" href="//csc.beap.bc.yahoo.com"/><link rel="dns-prefetch" href="//geo.query.yahoo.com"/><link r
href="//iquery.finance.yahoo.com"/><link rel="dns-prefetch" href="//fc.yahoo.com"/><link rel="dns-prefetch" href=
rel="preconnect" href="//l.yimg.com"/><link rel="preconnect" href="//s.yimg.com"/><link rel="preconnect" href="//
rel="preconnect" href="//b.scorecardresearch.com"/><link rel="preconnect" href="//iquery.finance.yahoo.com"/><lin
href="//yrtas.btrll.com"/><link rel="preconnect" href="//shim.btrll.com"/><link rel="icon" href="https://s.yimg.c
bypass=true" type="image/x-icon"/><link rel="canonical" href="https://finance.yahoo.com/quote/AAPL/"/><meta prope
date range, chart type and compare Apple Inc. against other companies."/><meta property="og:title" content="AAPL
the date range, chart type and compare Apple Inc. against other companies."/><meta property="twitter:title" conte
property="al:ios:app_store_id" content="328412701"/><meta property="al:ios:app_name" content="Yahoo Finance"/><me
content="intent://quote/AAPL/#Intent;scheme=yfinance;action=android.intent.action.VIEW;package=com.yahoo.mobile.c
end"/><meta property="al:android:app_name" content="Yahoo Finance"/><meta property="al:android:package" content="
banner&amp;pt=9029, app-argument=yfinance://quote/AAPL"/><link rel="stylesheet" href="https://s.yimg.com/zz/combo
site/css/atomic.17e8bca3.css&amp;os/finance/dd-site/css/app.8e30113a.css"/><script src="https://www.yahoo.com/pol
us%2Cpromise%2Carray.isarray%2Carray.prototype.every%2Carray.prototype.foreach%2Carray.prototype.indexof%2Carray.
.create%2Cobject.freeze%2Carray.prototype.filter%2Carray.prototype.reduce%2Cobject.assign%2Crequestanimationframe
ss/rapid-3.42.2.js"></script><script src="https://s.yimg.com/os/finance/dd-site/js/vendor.336c13fb391e6a2d642a.js
src="https://yep.video.yahoo.com/js/3/videoplayer-min.js?r=nextgen-desktop&amp;lang=en-US&amp;ypv=prod"></script>
c.replace("NoJs","");html.className = c;})(document.documentElement);</script><script>window.Modernizr=function(a
e=a[d];if(!f(e,"-")&&q[e]!==c)return"pfx"!=b||e}return!1}function h(a,b,d){for(var f in a){var g=b[a[f]];if(g!==c
")+d).split(" ");return e(b,"string")||e(b,"undefined")?g(f,b):(f=(a+" "+u.join(d+" ")+d).split(" "),h(f,b,c))}va
ms",t=s.split(" "),u=s.toLowerCase().split(" "),v={svg:"http://www.w3.org/2000/svg"},w={},x=[],y=x.slice,z=functi
e[d]:o+(d+1),j.appendChild(h);return f=["&#173;",'<style id="s',o,'">',a,"</style>"].join(""),j.id=o,(k?j:l).inne
(l.style.background="",l.style.overflow="hidden",i=n.style.overflow,n.style.overflow="hidden",n.appendChild(l)),g
{}.hasOwnProperty;k=e(A,"undefined")||e(A.call,"undefined")?function(a,b){return b in a&&e(a.constructor.prototyp
b)throw new TypeError;var c=y.call(arguments,1),d=function(){if(this instanceof d){var e=function(){};e.prototype
d}),w.canvas=function(){var a=b.createElement("canvas");return!(!a.getContext||!a.getContext("2d"))},w.history=fu
n.style&&z("@media (transform-3d),(-webkit-transform-3d){#modernizr{left:9px;position:absolute;height:3px;}}",fun
a=b.createElement("video"),c=!1;try{(c=!!a.canPlayType)&&(c=new Boolean(c),c.ogg=a.canPlayType('video/ogg; codecs
```

Often, its not so easy..

Nesting of Data



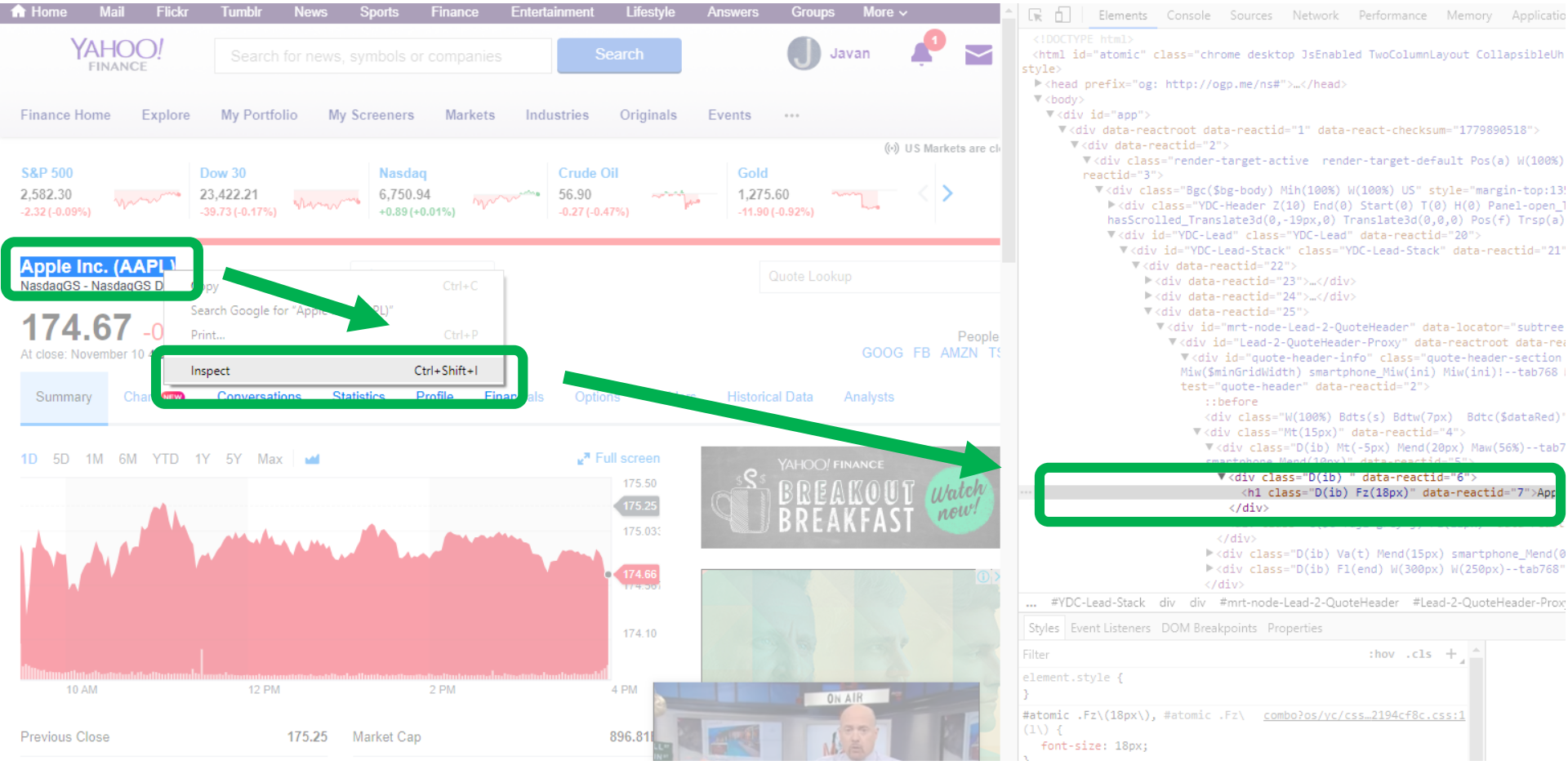
Nesting of Data: Example Yahoo



A Big Thank You to Community

Using Google Chrome: Developer Tools

- Yahoo Finance – AAPL (Apple Inc)
- Select Company Name
- Right-click > Inspect



Team Challenge

Scraping Real-time Data from Yahoo Finance

90 minutes

Team Challenge Rules!

Earn as many gold coins as possible by completing challenges!

*Note: Challenges must be completed in sequence

 x 2 : Challenge 1

 x 3 : Challenge 2

 x 4 : Challenge 3

 x 5 : Challenge 4

Spend your gold coins effectively to finish your challenges quickly!

Free : Use training materials

 x 1 : Seeking help from another sub-team

 x 2 : Ask facilitators

Prizes

Overall Best Pitch

Best Salesman

Group with the Most Outstanding Pitch

(By Popularity)

Overall Best Group

Techie Masters

Group with the Most Number of Gold coins

Best Team in the Group

Techie Buddies

Team with the Most Number of Gold coins

Best Participant in the Group

Tech Guru

Most Outstanding Participant in the Group

(Decided by TD Emcees and Game Masters)

Code the Future!

Lets get creative with Python. Python is a very simple programming language that can help you get started quickly!

Download the training notes here, and follow the lesson plan below.

Note: Are you still using IE? If so, please take 1 min to install Google Chrome to follow the lesson effectively.

Level 01: Baby Python

In this part of the course, you will learn to write and run basic python using a simple online python editor.

START PYTHON

Level 02: Hello Rebot

Here, you will write chatbot code using our integrated python editor which connects to our chatbot on Telegram.

TRY IT NOW!

Level 03: Taming Python

Next, let's learn basic HTML and webscrapping to extract data from websites, which are rich with information.

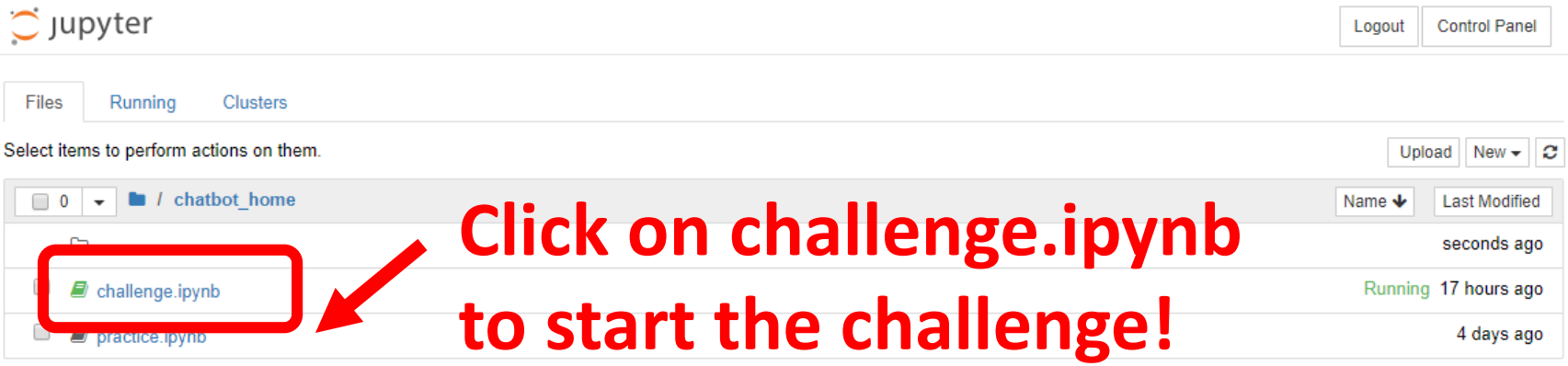
OK, LET'S GO!

Level 04: Rebot Challenge

Alright! Put all the code you have learnt together and help ReBot extract real-time Yahoo Finance data.

CHALLENGE ACCEPTED!

Jupyter: Integrated Python/Rebot



- Teams of 3
- Complete challenges in sequence

Instructions

jupyter challenge (unsaved changes)

Logout | Control Panel

File Edit View Insert Cell Kernel Help | Trusted | Python 3

Run | Try Your Code | Send to Rebot

Rebot Challenge: Webscraping Real-time Data from Yahoo Finance

For this challenge, your team will be web-scraping dyanmically from Yahoo Finance stock page.

Take a look at the stock page for Google Inc: Google - GOOG : https://finance.yahoo.com/quote/GOOG?p=GOOG

Hints: Remember the general steps for scraping HTML:

1. Get the Yahoo Finance webpage content of the ticker provided.
2. Format the content into proper HTML that is searchable.
3. Search for the piece information that your team wants.
4. Organise the information into the message.

And of course, remember to use the green "Send to Rebot" button to see your changes on Rebot via Telegram.

Good luck! :)

```
import rebot

def send_to_rebot(ticker):

    content = rebot.getContent(ticker)
    html = rebot.getHtml(content)

    stockName       = html.search("h1")
    beta            = html.search("td", "data-test", "BETA-value")

    message = rebot.createEmptyMessage()
    message.addText("Here are the results from Yahoo Finance!")
    message.addData("Stock Name", stockName)
    message.addData("Beta", beta)

    return message
```

Try it Out!

Test your changes by pressing the green "Try Your Code" button above!

```
print(send_to_rebot("AAPL")) #j_ignore_
```

Hints & Recap

HTML - NESTED CONTAINERS

Container (HTML)

Container (BODY)

Container (P)

HTML - CODE BLOCKS

```
<HTML>

    <BODY>

        <P CLASS="Text">
        Hello World!
        </P>

    </BODY>

</HTML>
```

HTML - RAW CODE

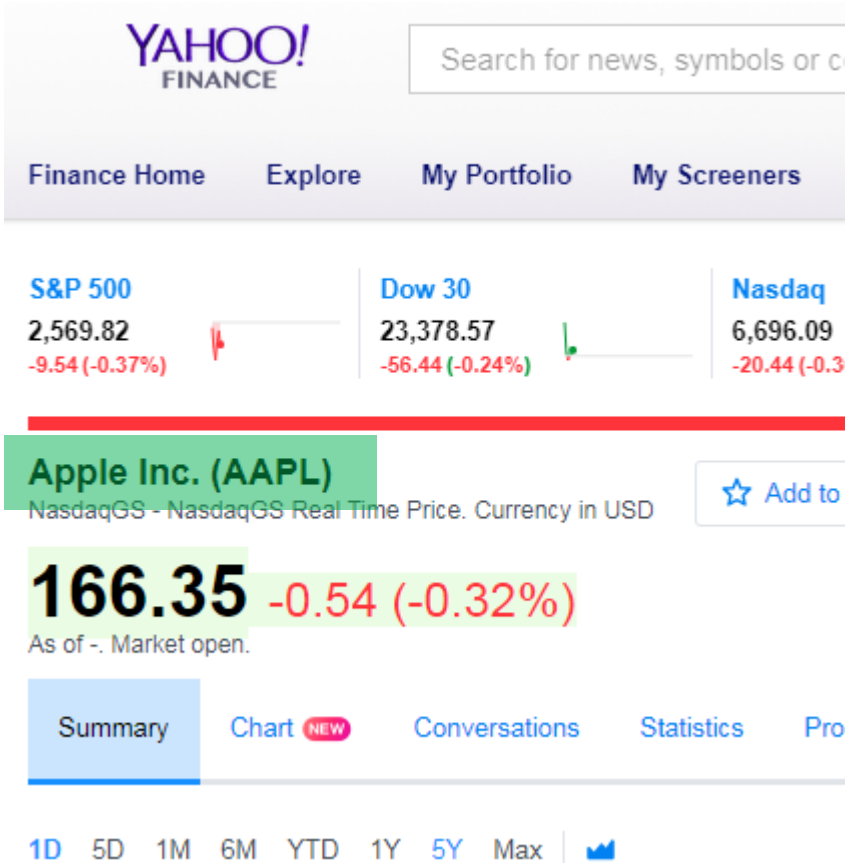
```
<HTML>

<BODY>

<P CLASS="Text">
Hello World!
</P>

</BODY>

</HTML>
```

- These values are enclosed in tags (think of containers).
- They are the basic building blocks needed used to build basic to complex websites.

```
data = html.search("p", "class", "text")
```

Hint A


```
<h1
class="D(ib) Fz(18px)"
data-reactid="7">
Apple Inc. (AAPL)
</h1>
```

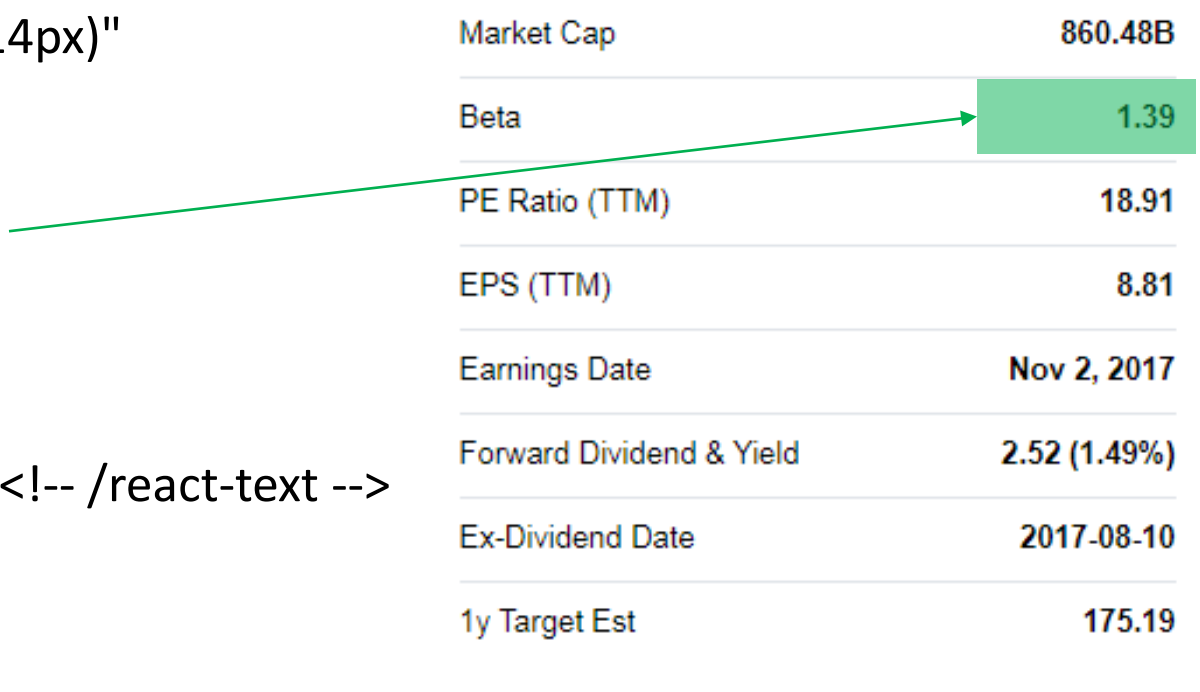


Solution:

data = html.search("h1", "class", "D(ib) Fz(18px)")

Hint B


```
<td
class="Ta(end) Fw(b) Lh(14px)"
data-test="BETA-value"
data-reactid="89">

<span
class="Trsdu(0.3s) "
data-reactid="90">
<!-- react-text: 91 -->1.39<!-- /react-text -->
</span>

</td>
```



Market Cap	860.48B
Beta	1.39
PE Ratio (TTM)	18.91
EPS (TTM)	8.81
Earnings Date	Nov 2, 2017
Forward Dividend & Yield	2.52 (1.49%)
Ex-Dividend Date	2017-08-10
1y Target Est	175.19

Solution:

```
data = html.search("h1", "data-test", " BETA-value")
```

Good Luck!

HAVE
FUN
AND
KEEP
SMILING

Team Challenge Round

*Your team must complete challenges in sequence

Challenge 1

Get the following information

- Company Name
- Ticker Price

Challenge 2

Get the following stock indicators

- Beta
- PE Ratio (TTM)
- EPS (TTM)

Challenge 3

Give a recommendation if the following conditions are satisfied:

- Beta >= 1.5 – “RISKY”
- Beta < 1.5 – “SAFE”

Challenge 4

Get the latest news headline

Resources

Code the Future!

Lets get creative with Python. Python is a very simple programming language that can help you get started quickly!

Download the training notes here, and follow the lesson plan below.

Note: Are you still using IE? If so, please take 1 min to install Google Chrome to follow the lesson effectively.

Level 01: Baby Python

In this part of the course, you will learn to write and run basic python using a simple online python editor.

START PYTHON

Level 02: Hello Rebot

Here, you will write chatbot code using our integrated python editor which connects to our chatbot on Telegram.

TRY IT NOW!

Level 03: Taming Python

Next, let's learn basic HTML and webscrapping to extract data from websites, which are rich with information.

OK, LET'S GO!

Level 04: Rebot Challenge

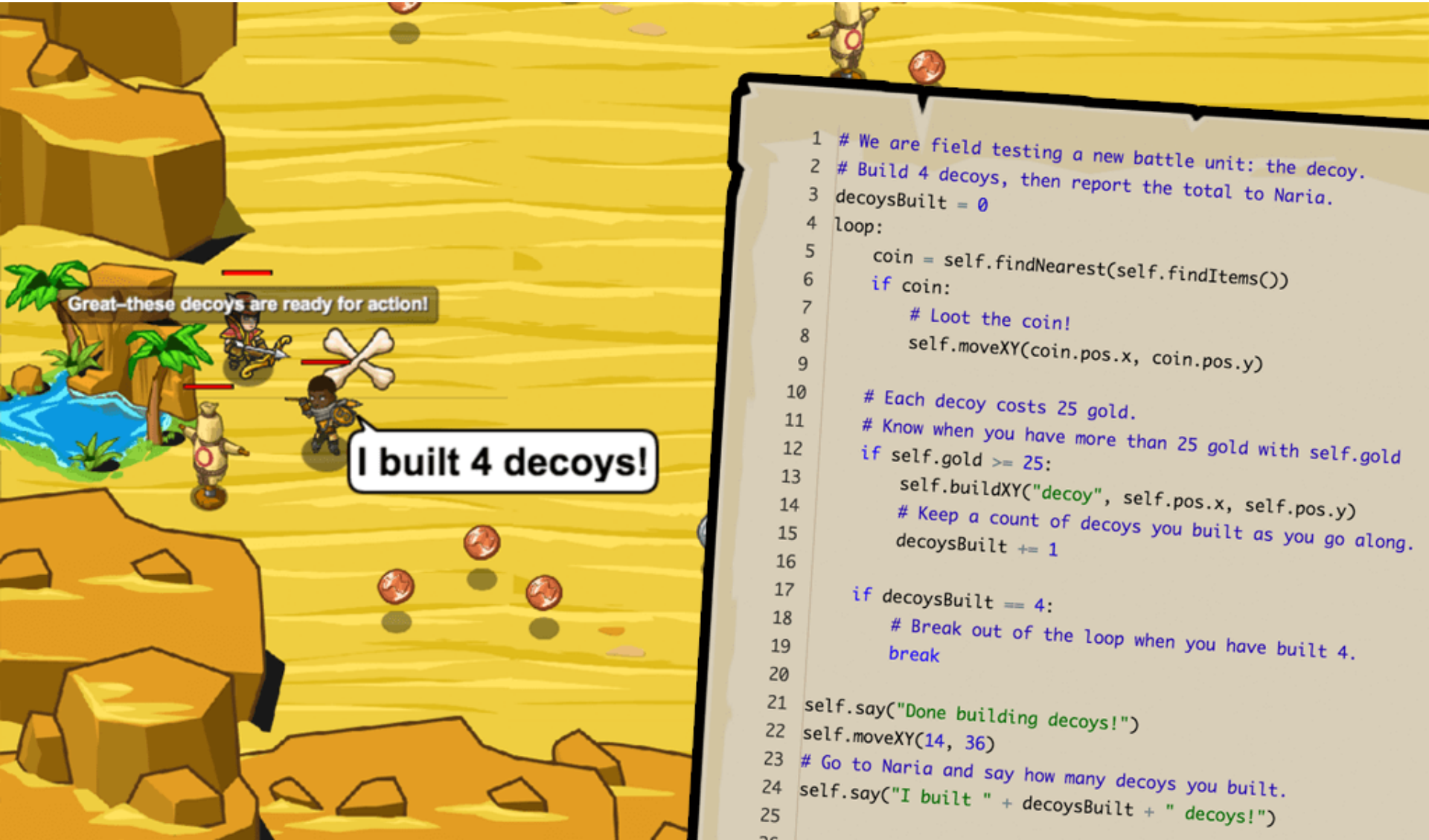
Alright! Put all the code you have learnt together and help ReBot extract real-time Yahoo Finance data.

CHALLENGE ACCEPTED!

CodeCombat.com

Resources

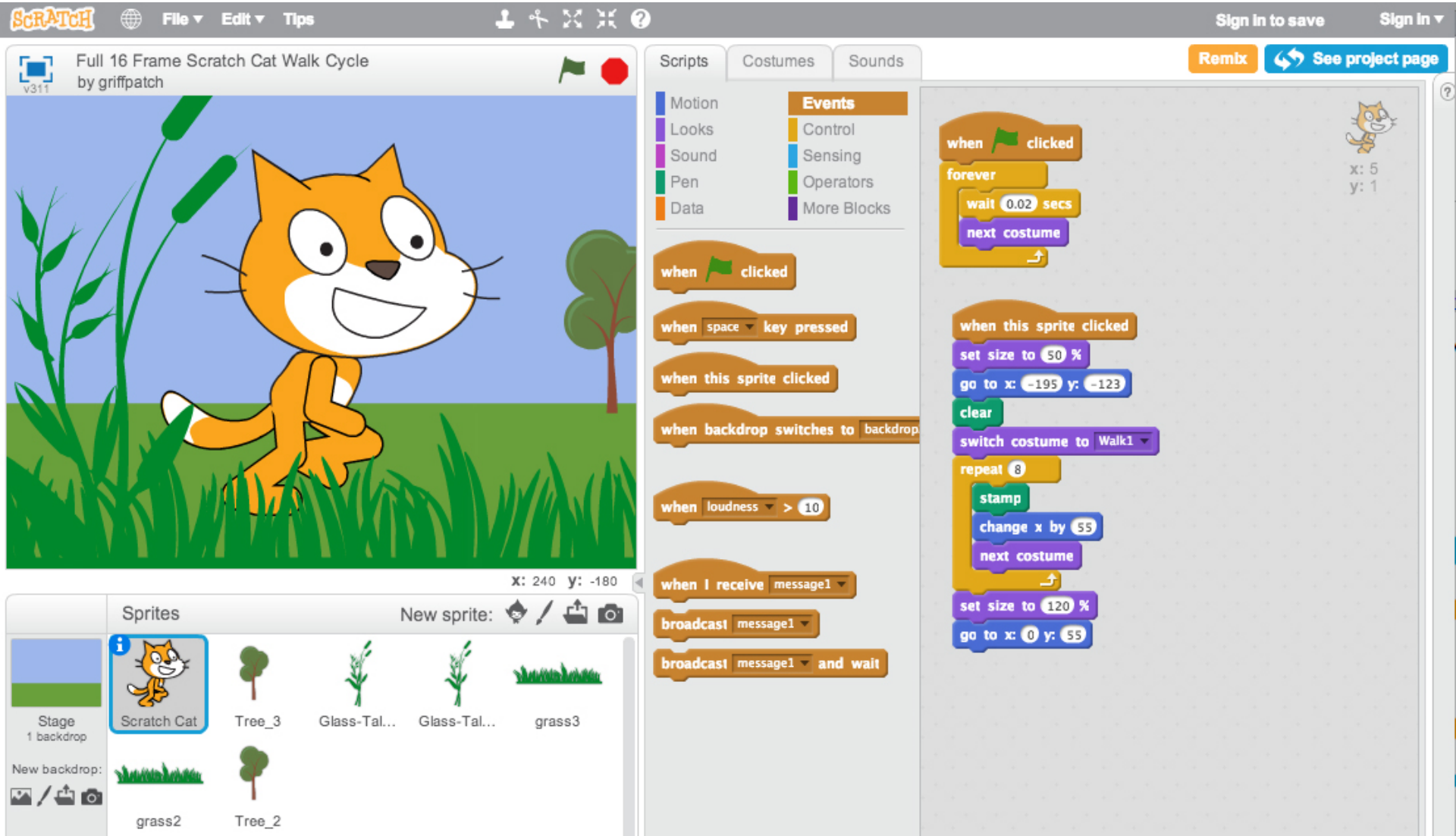
CodeCombat is a fun browser-based coding game that teaches programming. In order to advance through the game's levels, players must prove their knowledge by writing code.



Scratch.mit.edu

Resources

Scratch was created by MIT to help people learn to think creatively, reason systematically and work collaboratively. You can use it to create and publish interactive stories, animations, games!



Pitch Perfect

45 minutes

Sell Your Idea !

- 2 Teams merge into 1 Big Team (6 people)

- In your big teams, in the next 10 mins, think of **any** data & tech idea that you want to sell to the rest.
  - It may or may not be directly related to your work
  - It could be an idea to automate or improve efficiency
  - It could be an enhancement to an existing product (eg: Tableau dashboard, REKI)
  - Free format – Do a Skit? Flip Charts? Just Talk?

- Each team will be given 3 minutes to present

Pick 2 teams

- Finally, pick 2 teams to represent your group in the final round in 38th Auditorium.

IMAGINE

PITCH