

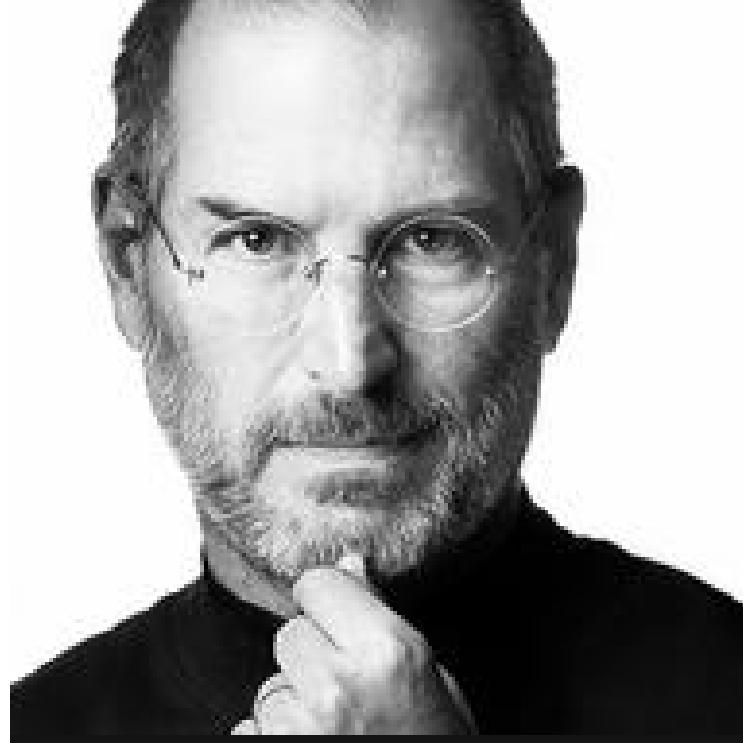
# Code Challenge

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 programming brings out in every one of us is logical/analytical thinking and the ability to solve problems in a creative manner.

- Hoe Yin



, ,

# So, Guess Who Wrote This?

Sudoku2.cpp

```
151
152     void Place(int S)
153 {
154     LevelCount[S]++;
155     Count++;
156
157     if (S >= 81) {
158         Succeed();
159         return;
160     }
161
162     int S2 = NextSeq(S);
163     SwapSeqEntries(S, S2);
164
165     int Square = Sequence[S];
166
167     int     BlockIndex = InBlock[Square],
168          RowIndex = InRow[Square],
169           ColIndex = InCol[Square];
170
171     int     Possibles = Block[BlockIndex] & Row[RowIndex] & Col[ColIndex];
172     while (Possibles) {
173         int valbit = Possibles & (-Possibles); // Lowest 1 bit in Possibles
174         Possibles &= ~valbit;
175         Entry[Square] = valbit;
176         Block[BlockIndex] &= ~valbit;
177         Row[RowIndex] &= ~valbit;
178         Col[ColIndex] &= ~valbit;
179
180         Place(S + 1);
181
182         Entry[Square] = BLANK; // Could be moved out of the Loop
183         Block[BlockIndex] |= valbit;
184         Row[RowIndex] |= valbit;
185         Col[ColIndex] |= valbit;
186     }
187
188     SwapSeqEntries(S, S2);
189 }
190 }
```

		3	8			5
8			6	9	7	
	9		2			4
6			3		2	
	8		4		1	
	3		5			6
4			7		5	
		7	5	1		8
2			9	4		

I used to enjoy this; it is a long time since I've done anything.  
The last program I wrote was a Sudoku solver in C++  
several years ago, so I'm out of date.

- Prime Minister Lee Hsien Loong



# Basic Introduction To Coding

- Basic Python
- Web scraping

## Aim

Most Importantly



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

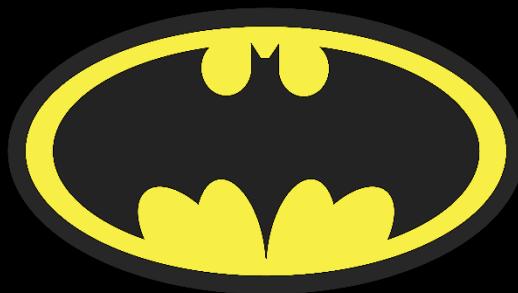
- 4:30-5:15 – Idea Generation and Pitch preparation
- 5:15-5:30 – Move to 38 Auditorium
- 5:30-6:15 – Pitch, Awards and Closing

# Get Ready!

# Why Code?

# COOLING

IT MAY BE THE CLOSEST THING  
WE HAVE TO A SUPERPOWER





Let's get a computer to do things.

But,

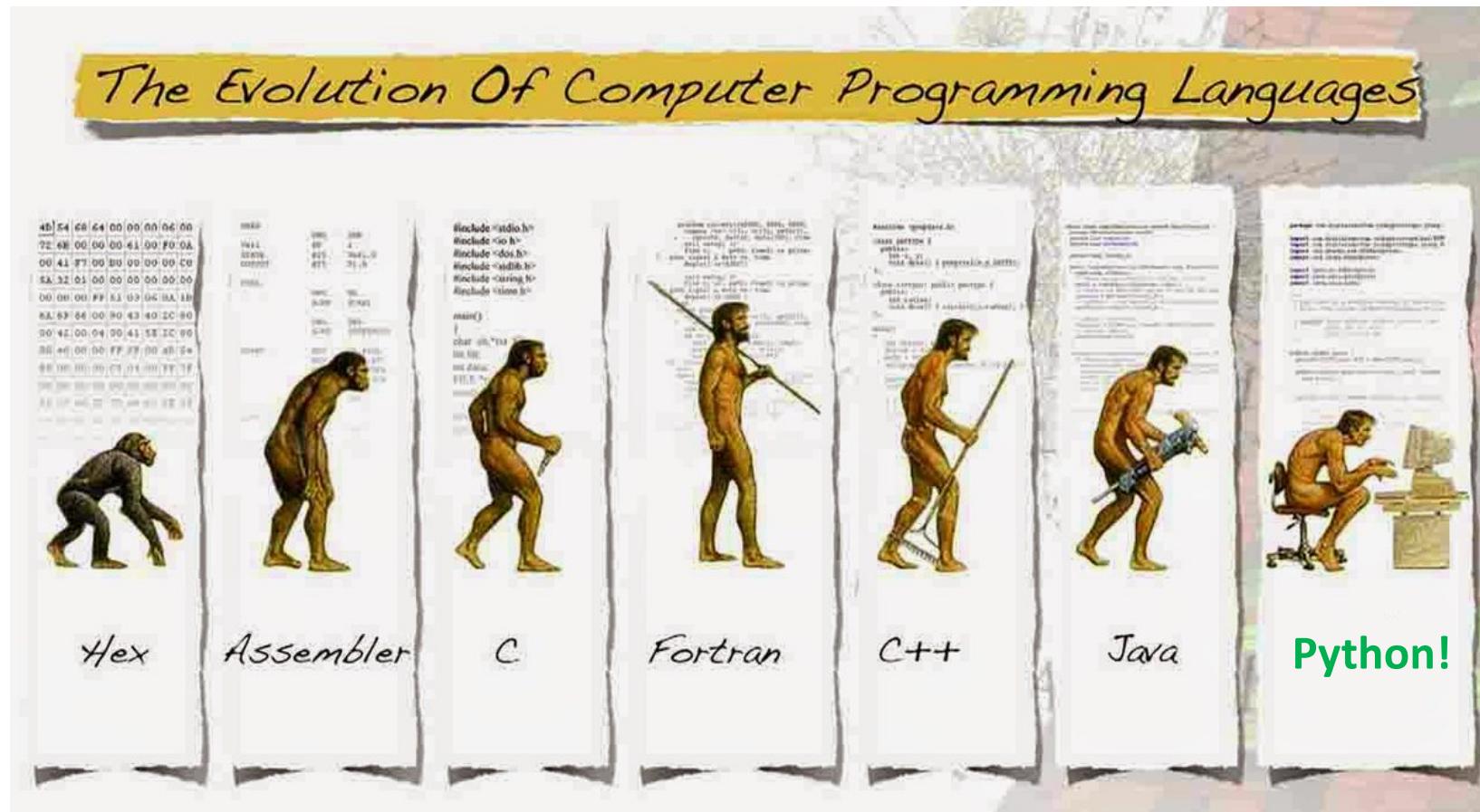
- The machine is a fast worker but very stupid
  - “Please, make me a sandwich!”
  - Will this work?
    - Step 1: Take a slice of bread, put it on the board
    - Step 2: Apply a 1mm layer of butter on the bread
    - Step 3: Take a slice of ham, put it on the bread
  - Probably closer to this,
    - 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?

# 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](http://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 03: Taming Python

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

[OK, LET'S GO!](#)

## 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 04: Rebot Challenge

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

[CHALLENGE ACCEPTED!](#)

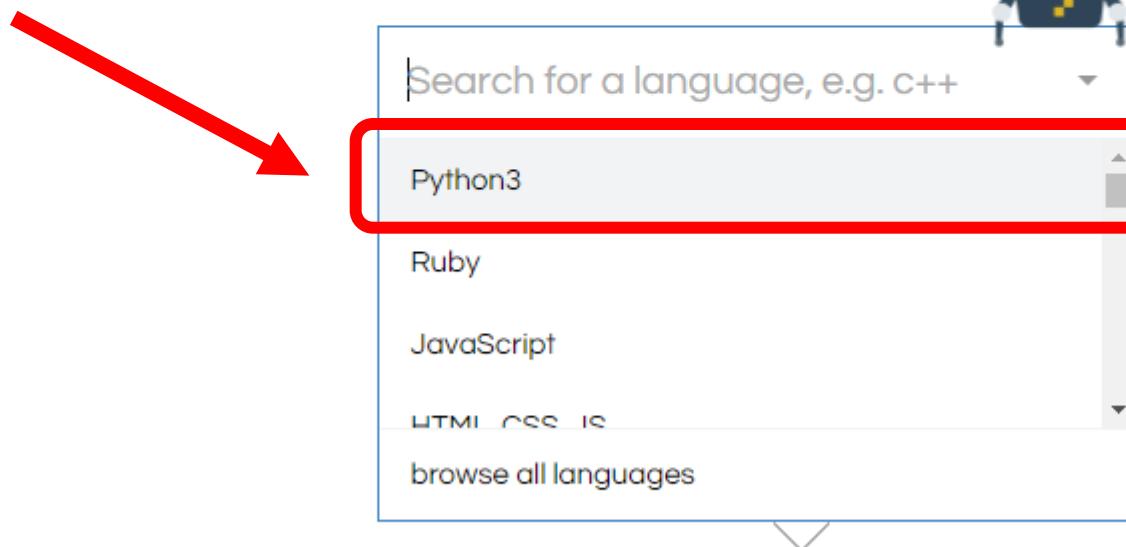
# REPL: Basic Online Python Editor

01 Python

replit is **a cloud coding environment** for ES6

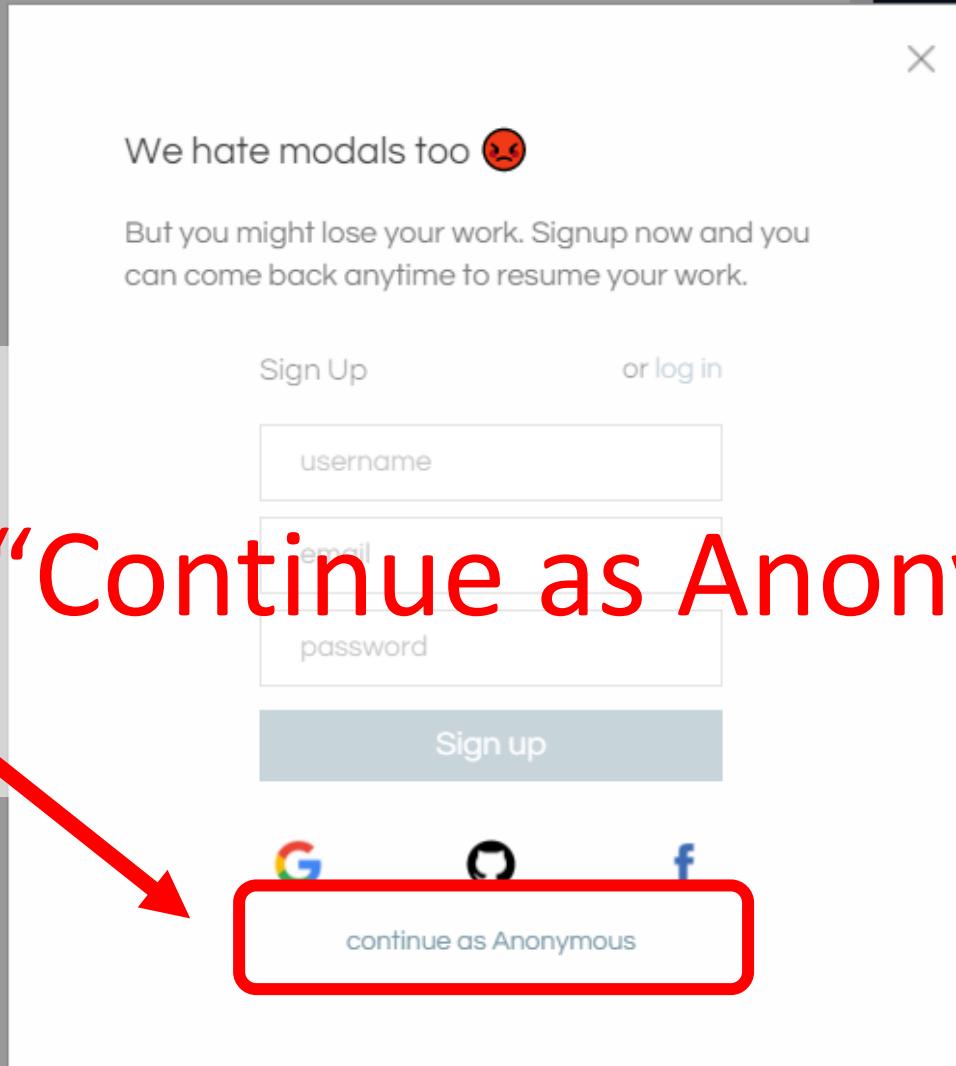
join a community of  
*engineers, teachers, and students*

Select “Python3”



# REPL: Basic Online Python Editor

01 Python



Select “Continue as Anonymous”

# REPL: Basic Online Python Editor

01 Python



Untitled

Log in



1

Not sure what to do? Run some [examples](#) (dismiss)

share ↗

save ↘

run ➔



input ↗

clear ↘

(Left-Hand Side)

**Write code here**

Python 3.5.2 (default, Dec 2015, 13:05:11)  
[GCC 4.8.2] on linux

>

(Right-Hand Side)

**See output here**

# REPL: Basic Online Python Editor

01 Python

The screenshot shows the repl.it interface. At the top left is the repl.it logo. The title bar says "Untitled". On the right is a "Log in" button. Below the title bar are three buttons: "share" with a link icon, "save" with a disk icon, and "run" with a play icon. A red arrow points from the text below to this row of buttons. To the left of the main area is a sidebar with icons for gear (settings), document (file), and user (profile). A message in the sidebar says "Not sure what to do? Run some examples (dismiss)". The main workspace is dark-themed and displays Python code. At the top of the code area, there are "input" and "clear" buttons. The code itself starts with "Python 3.5.2 (default, Dec 2015, 13:05:11) [GCC 4.8.2] on linux".

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

- 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?

01 Python

- Excel Formulas

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

- Excel Worksheet Cells

- Think of “Cells” like “Containers”

I am an Excel Worksheet

	A	B	C	D
1	I am an Excel Worksheet			
2		Value 1 = 100	<- Container	
4		Value 2 = 200	<- Container	
6		Sum of Value 1 & Value 2 = 300	<- Print this	

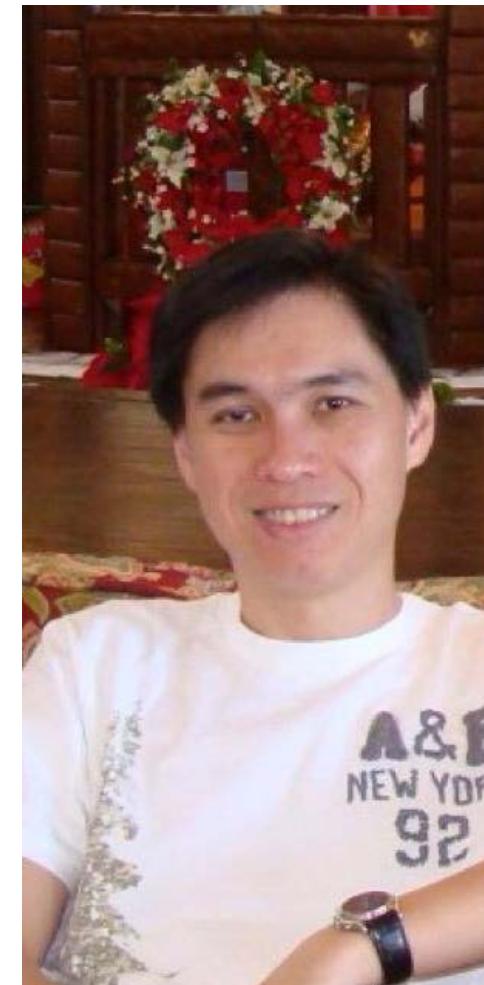
- Coding easy to learn (And, can be fun ☺)
  - Following a recipe: EASY
  - Creating a recipe: HARD
- Given set of values: 5, 9, 10, 4, 11, 3, 6
  - Highest and lowest number?
  - What's your recipe?
- Don't just translate your work procedures
  - Simplify and refine it.

“

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

Install Telegram



# Telegram

Singtel 4G      9:36 AM      65% Singtel WiFi      9:37 AM      64%

telegram

Telegram Messenger  
Social Networking  
★★★★★ 62

The world's fastest messaging app.

Start Messaging

A screenshot of the Telegram mobile application. It shows a search bar at the top with the word "telegram". Below it is the app's logo and name "Telegram Messenger" with the subtitle "Social Networking". A 5-star rating with 62 reviews is displayed. The main screen shows a list of contacts with small profile pictures and names like "John", "Tyron Lancaster", "Kapten", "Hanselberg", "Bender", and "Helping homeless pets". Below the contacts is a message feed with several messages from different users. One message from "Eileen Lockhart" asks "So why is Telegram cool?", and another from "Hey Eileen" replies "Well, look, Telegram is so fast, and you can use it on all your devices at the same time — phones, tablets, even desktops...". There are also mentions of "And it has secret chats, like this one, with end-to-end encryption!" and "End everyone to what???" followed by a fire emoji.



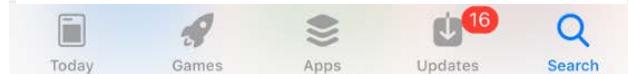
# Telegram

The world's **fastest** messaging app.  
It is **free** and **secure**.



Download on the  
App Store

ANDROID APP ON  
Google Play



Start Messaging



# Step 2: Register Account

Install Telegram

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

The image shows two screenshots of the Telegram registration process. The first screenshot, titled 'Your Phone', displays a dropdown menu for country codes with 'Singapore' selected. It also shows a field for entering a phone number starting with '+65'. The second screenshot, titled 'Your Phone Number', shows a message stating 'We have sent you an SMS with the code' followed by a blank input field labeled 'Code'.

9:37 AM      9:38 AM

Singtel      Singtel

64%      64%

Back      Back

Next      Next

Your Phone

Please confirm your country code and enter your phone number.

Singapore

+65 Your phone number

Code

By signing up,  
you agree to the [Terms of Service](#).

1	2 ABC	3 DEF
4 GHI	5 JKL	6 MNO
7 PQRS	8 TUV	9 WXYZ
0		✖

1	2 ABC	3 DEF
4 GHI	5 JKL	6 MNO
7 PQRS	8 TUV	9 WXYZ
0		✖

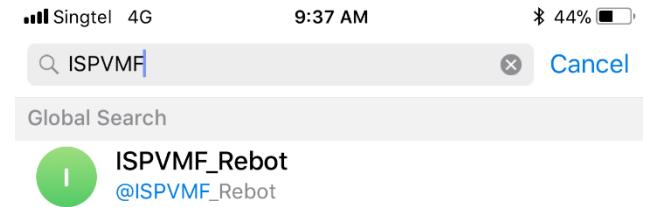
# Step 3: Add @ISPVMF\_ReBot

Install Telegram

- Add a Chat



- Search for  
@ISPVMF\_Rebot



- Select  
@ISPVMF\_Rebot

You have no conversations yet

Start messaging by pressing the pencil  
button in the top right corner or go to the  
Contacts section.





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 03: Taming Python

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

[OK, LET'S GO!](#)

## 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 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)

02 Chatbot

Start!

CODE.NOW 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 slides 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: Basic Python

In this part of the course, you will learn how to code in python using a simple command line interface.

START NOW!

Hello! How shall we identify you?

alphanumeric only

Your NTid please

NEXT

### Hello Rebot

Rebot is a chatbot built using our integrated python library. You can interact with our chatbot on Telegram.

NOW!

### Level 03: Taming Python

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

OK, LET'S GO!

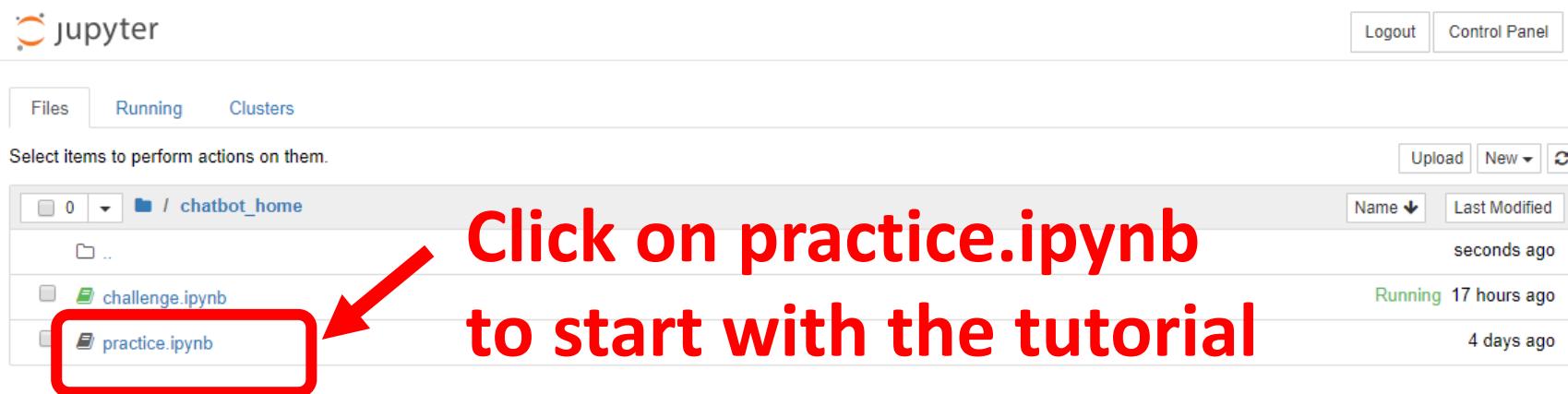
### Level 04: Rebot Challenge

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

CHALLENGE ACCEPTED!

# Jupyter: Integrated Python/Rebot

02 Chatbot



The screenshot shows the Jupyter Notebook interface. At the top, there's a navigation bar with 'Logout' and 'Control Panel'. Below it, a toolbar has 'Files' (selected), 'Running', and 'Clusters' tabs. A message says 'Select items to perform actions on them.' On the right, there are 'Upload', 'New', and a refresh icon. The main area shows a file list in 'chatbot\_home':

Name	Last Modified
challenge.ipynb	seconds ago
practice.ipynb	Running 17 hours ago
	4 days ago

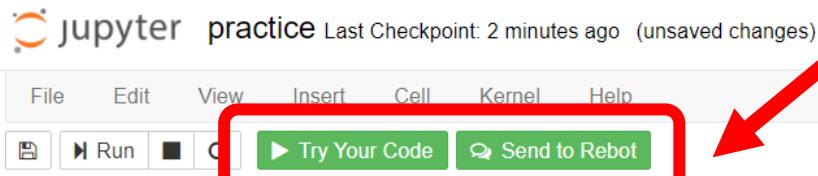
A red arrow points from the text 'Click on practice.ipynb to start with the tutorial' to the 'practice.ipynb' file in the list.

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

02 Chatbot



## 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

02 Chatbot

- 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

# 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 webscraping 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!](#)

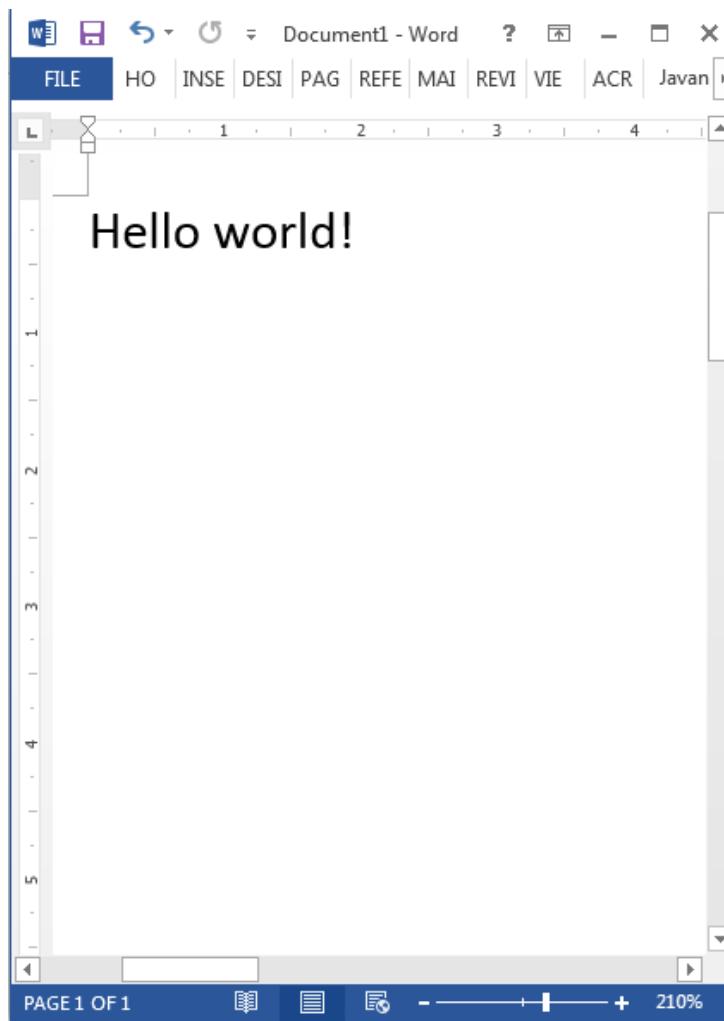
- HTML is
  - The building block of websites.
  - Source code of the Web
- Concept of HTML Tags
  - Open tags <HTML>
  - Close tags </HTML>
- Let's see how to scrape data from HTML



# Word Doc vs HTML Doc

03 Web-scraping

Word Document



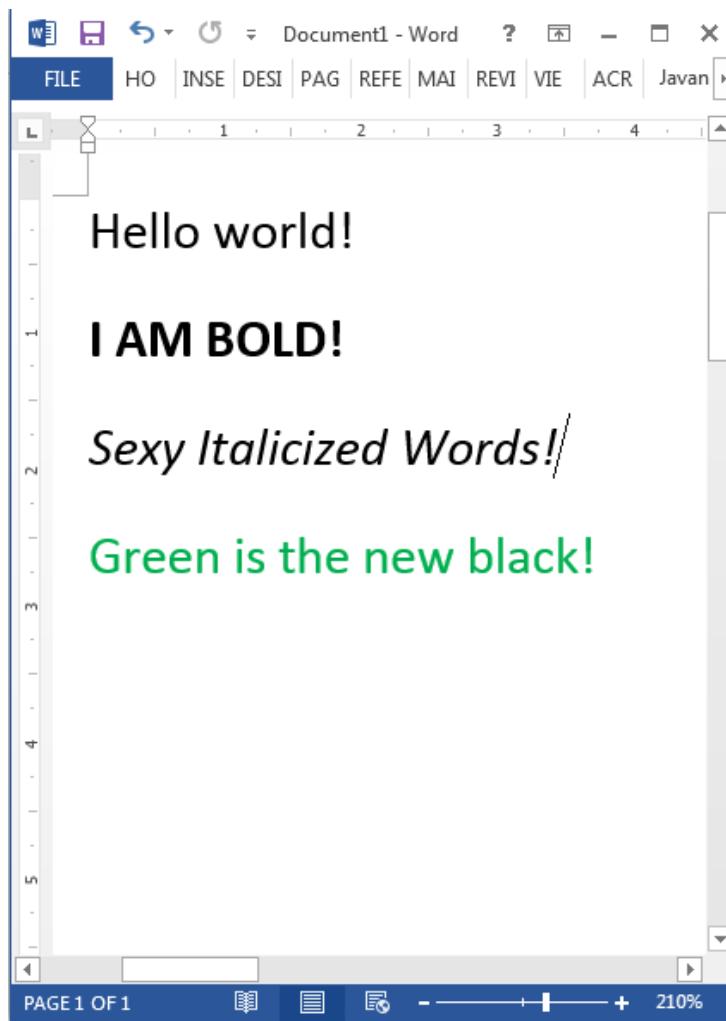
HTML Document



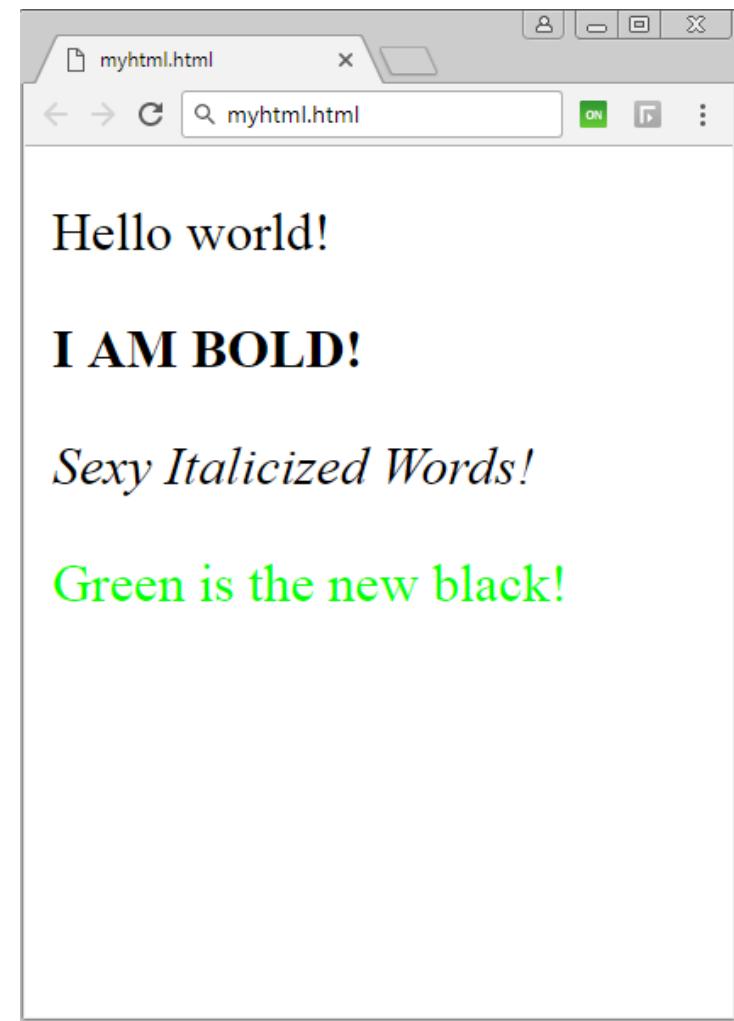
# Word Doc vs HTML Doc

03 Web-scraping

Word Document



HTML Document



# HTML - Code Visualization

03 Web-scraping

## 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>
```

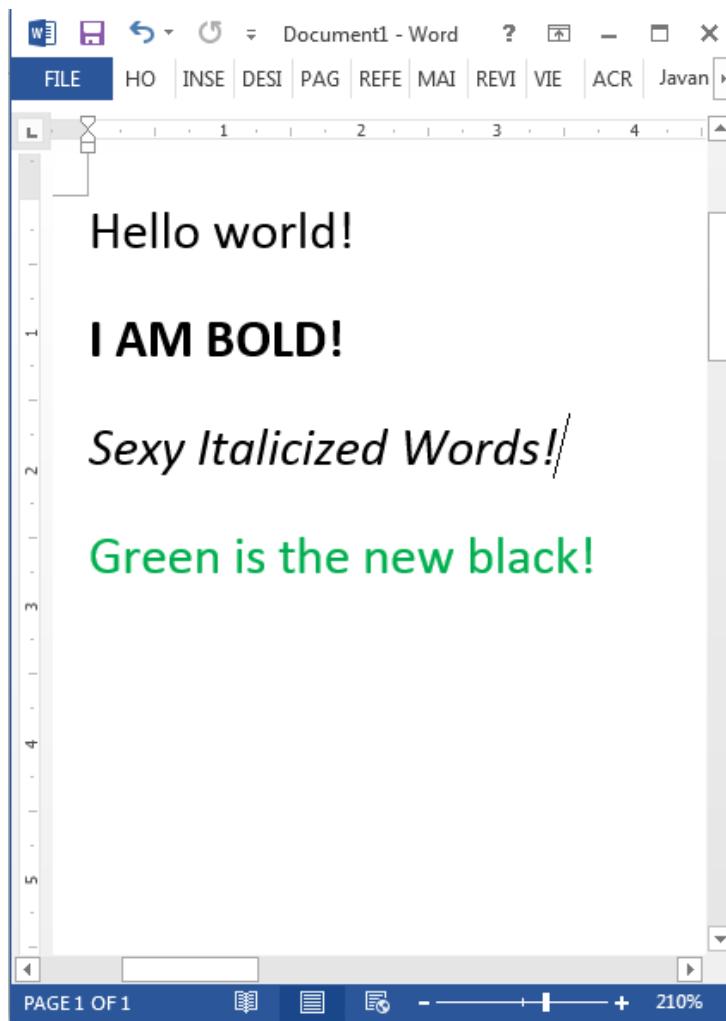


- These values are enclosed in tags (think of containers).
- They are the basic building blocks needed used to build basic to complex websites.

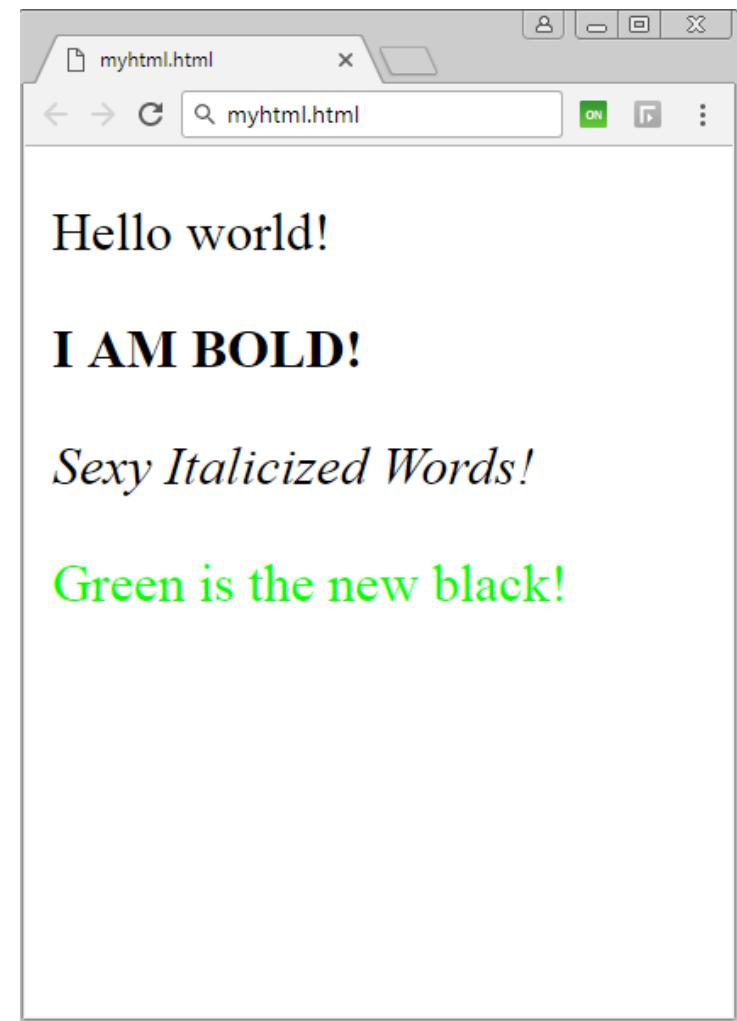
# Word Doc vs HTML Doc

03 Web-scraping

Word Document



HTML Document



## 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

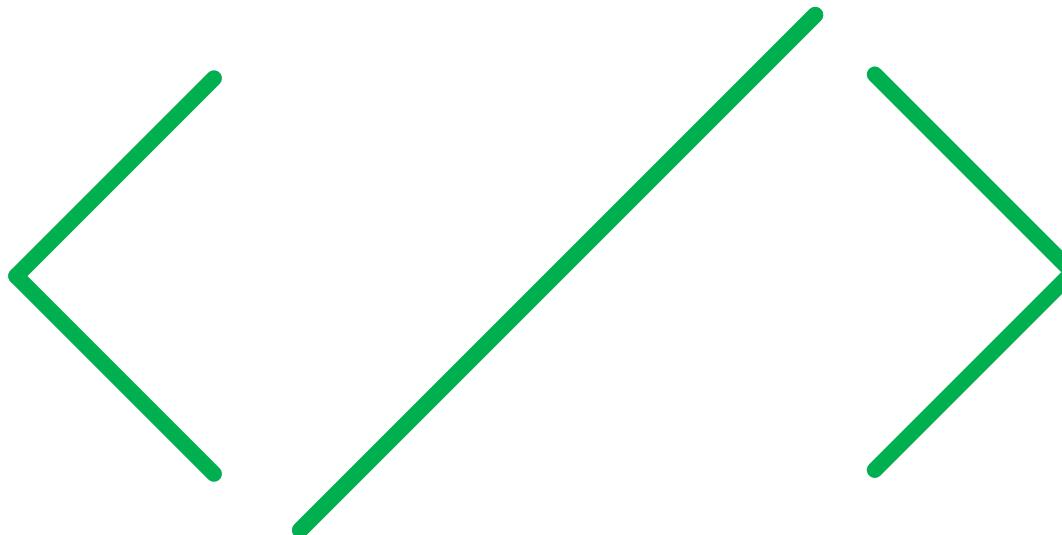
03 Web-scraping

Think of a

**HTML TAG**

As a

**CONTAINER**



# Think of Tags as Containers

03 Web-scraping

Think of a

**HTML TAG**

As a

**CONTAINER**

I AM A VALUE

# So many types of HTML tags...

03 Web-scraping

## HTML 5 NEW TAG

### TAG NOT SUPPORTED IN HTML 5

<!--><!--&gt;</td> <td>Define a comment</td>	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 acronym
<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
  	Defines a long quotation elite
<body>	Defines the body element
 	Inserts a single line break
<button>	Defines a push button autofocus, disabled, form, formaction, formmethod, formnovalidate, formtarget, height, type, value
<canvas>	Defines graphics height, width
<caption>	Defines a table caption
<center>	Used to center text and tables
<ci>	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

For  
Illustration  
Purposes  
Only

<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
<div>	Defines a section in a document
<dl>	Defines a definition list
<dt>	Defines a definition term
<em>	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 organizes 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, 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
<li>	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
<ol>	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 do 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 elite
<select>	Defines a selectable list autofocus, disabled, form, multiple, name, size
<small>	Defines small text
<source>	Defines media resources media, src, type
<span>	Defines a section in a document
<strong>	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, maxlen, 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/time datetime
<title>	Defines the document title
<tr>	Defines a table row datetime
<tt>	Used to define teletype text
<u>	Used to define underlined text
<ul>	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

# Useful Tags for Today

03 Web-scraping

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 webscraping 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

03 Web-scraping

The screenshot shows a web-based code editor interface. At the top, there are user icons and the handle '@rebot/html'. Below that, a message reads 'Just a basic tutorial on HTML & web-scraping. Have fun :)'.

At the top right, there are buttons for 'share' and 'run'. A red box highlights the 'fork' button, which has a symbol of two people. An arrow points from the text below to this button.

The code editor displays a Python script named 'main.py' with the following content:

```
1 # REBOT 2017
2 # You can write your code below the next line.
3 # https://rebot.chat/example.html
4 # --
```

On the right side of the screen, there is a terminal window showing the command line environment:

```
Python 3.6.1 (default, Dec 2015, 13:05:11)
[GCC 4.8.2] on linux
>
```

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

# Let's Give it a Try

03 Web-scraping

The screenshot shows a code editor interface with three files listed in the tabs: `main.py`, `example.html`, and `rebot.py`. A red box highlights the first three tabs. A red arrow points from the text "You should see three files:" down to the highlighted area.

share run fork

main.py example.html rebot.py

```
1 # REBOT 2017
2 # You can write your code below the next line.
3 # https://rebot.chat/example.html
4 # --
```

Python 3.6.1 (default, Dec 2015, 13:05:11)
[GCC 4.8.2] on linux
>

input

You should see three files:

- **main.py**
- **example.html**
- **rebot.py**

# Let's Give it a Try

03 Web-scraping

## 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

## 03 Web-scraping

```
# 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)
```

# 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.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&pt=9029, app-argument=yfinance://quote/AAPL"/><link rel="stylesheet" href="https://s.yimg.com/zz/combo  
site/css/atomic.17e8bca3.css&os/finance/dd-site/css/app.e30113a.css"/><script src="https://www.yahoo.com/po  
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/videooplayer-min.js?r=nextgen-desktop&lang=en-US&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:1).inne  
(l.style.backgroundColor="",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.prototype  
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..

03 Web-scraping

## HOW TO FIND A CONTAINER LIKE DAT?



## Geography

Country

City

Town

Street

Block

## Websites

Html

Body

Table

TD (Table Data)

Span

# Nesting of Data: Example Yahoo

03 Web-scraping

**YAHOO! FINANCE**

Search for news, symbols or companies  Search Sign in 1 ✉

Finance Home Explore My Portfolio My Screeners Markets Industries Originals Events Personal Finance Technology (⌚) US Markets close in 5 hrs and 50 mins

S&P 500 2,569.80 -9.56 (-0.37%) Dow 30 23,369.81 -65.20 (-0.28%) Nasdaq 6,690.98 -25.56 (-0.38%) Crude Oil 54.29 -0.01 (-0.02%) Gold 1,277.70 +0.40 (+0.03%) Silver 17.17 -0.01 (-0.06%) < >

**Apple Inc. (AAPL)** NasdaqGS - NasdaqGS Real Time Price. Currency in USD Add to watchlist

**165.93 -0.96 (-0.58%)** As of - Market open.

Quote Lookup 🔍

People also watch: GOOG FB AMZN TSLA NFLX

Summary Chart NEW Conversations Statistics Profile Financials Options Holders Historical Data Analysts

Previous Close 166.89 Market Cap 860.48B  
Open 167.64 Beta 1.39  
**Bid** 166.96 x 1000 PE Ratio (TTM) 18.91  
Ask 166.99 x 300 EPS (TTM) 8.81  
Day's Range 165.83 - 167.66 Earnings Date Nov 2, 2017  
52 Week Range 104.08 - 169.65 Forward Dividend & Yield 2.52 (1.49%)  
Volume 5,229,801 Ex-Dividend Date 2017-08-10  
Avg. Volume 28,273,096 1y Target Est 175.19

Trade prices are not sourced from all markets

1D 5D 1M 6M YTD 1Y 5Y Max Full screen

168.00  
167.20  
166.89  
166.40  
166.25  
165.60

10 AM 12 PM 2 PM 4 PM

**YAHOO! FINANCE ALL MARKETS SUMMIT Watch live**

**Microsoft** Try Azure for free and get \$200 credit Get started now

**Related Video**

**NBC NEWS EXCLUSIVE**

**Apple CEO Cook: Tax reform is sorely needed**

Earnings > Actual Estimate

# A Big Thank You to Community

03 Web-scraping

## Using Google Chrome: Developer Tools

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

The screenshot shows a Yahoo Finance page for Apple Inc. (AAPL) with various financial metrics and a chart. A context menu is open over the company name, with the 'Inspect' option highlighted. A green arrow points from this highlighted option to the corresponding entry in the developer tools sidebar. The sidebar displays the full HTML DOM structure of the page, with the selected element (the company name) highlighted in red.

YAHOO! FINANCE

Finance Home Explore My Portfolio My Screeners Markets Industries Originals Events ...

S&P 500 2,582.30 -2.32 (-0.09%) Dow 30 23,422.21 -39.73 (-0.17%) Nasdaq 6,750.94 +0.89 (+0.01%) Crude Oil 56.90 -0.27 (-0.47%) Gold 1,275.60 -11.90 (-0.92%)

Apple Inc. (AAPL)  
NasdaqGS - NasdaqGS D

174.67 -0 At close: November 10 4:00 PM

Summary Chart Conversations Statistics Profile Financials Options Historical Data Analysts

1D 5D 6M YTD 1Y 5Y Max

10 AM 12 PM 2 PM 4 PM

Previous Close 175.25 Market Cap 896.811

Elements Console Sources Network Performance Memory Application

```
<!DOCTYPE html>
<html id="atomic" class="chrome desktop JsEnabled TwoColumnLayout CollapsibleUI">
  <head prefix="og: http://ogp.me/ns#">...</head>
  <body>
    <div id="app">
      <div data-reactid="1" data-react-checksum="1779890518">
        <div data-reactid="2">
          <div class="render-target-active render-target-default Pos(a) W(100%)" reactid="3">
            <div class="Bgc($bg-body) Mih(100%) W(100%) US" style="margin-top:13px;">
              <div class="YDC-Header Z(10) End(0) Start(0) T(0) H(0) Panel-open_L hasScrolled_Translate3d(0,-19px,0) Translate3d(0,0,0) Pos(f) Trsp(a)">
                <div id="YDC-Lead" class="YDC-Lead" data-reactid="20">
                  <div id="YDC-Lead-Stack" class="YDC-Lead-Stack" data-reactid="21">
                    <div data-reactid="22">
                      <div data-reactid="23">...</div>
                      <div data-reactid="24">...</div>
                    <div data-reactid="25">
                      <div id="mrt-node-Lead-2-QuoteHeader" data-locator="subtree">
                        <div id="Lead-2-QuoteHeader-Proxy" data-reactroot data-reactid="4">
                          <div id="quote-header-info" class="quote-header-section Miw($minGridWidth) smartphone_Miw(ini) Miw(ini)!--tab768 !test='quote-header'" data-reactid="2">
                            ::before
                            <div class="W(100%) Bdt(s) Bdtw(7px) Bdtc($dataRed)">
                              <div class="Mt(15px)" data-reactid="4">
                                <div class="D(ib) Mt(-5px) Mend(20px) Maw(56%)--tab768 smartphone_Maw(18px)" data-reactid="5">
                                  <div class="D(ib) " data-reactid="6">
                                    <h1 class="D(ib) Fz(18px)" data-reactid="7">App...
                                  </div>
                                </div>
                              </div>
                            </div>
                          <div class="D(ib) Va(t) Mend(15px) smartphone_Mend(0)" data-reactid="6">
                            <div class="D(ib) F1(end) W(300px) W(250px)--tab768" data-reactid="7">
                              ...
                            </div>
                          </div>
                        </div>
                      </div>
                    </div>
                  </div>
                </div>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </body>
</html>
```

Styles Event Listeners DOM Breakpoints Properties

Filter :hover .cls +

element.style { }

#atomic .Fz\18px\), #atomic .Fz\ combo?os=vc/css\_2194cf8c.css:1 (1\){ font-size: 18px;

# Team Challenge

Scraping Real-time Data from Yahoo Finance

90 minutes

# Team Challenge Rules!

Earn as many gold coins as possible by completing challenges!

Spend your gold coins effectively to finish your challenges quickly!

\*Note: Challenges must be completed in sequence



x 2 : Challenge 1



x 3 : Challenge 2



x 4 : Challenge 3



x 5 : Challenge 4

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 webscraping 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

04 Challenge

The screenshot shows the Jupyter Notebook interface with the following details:

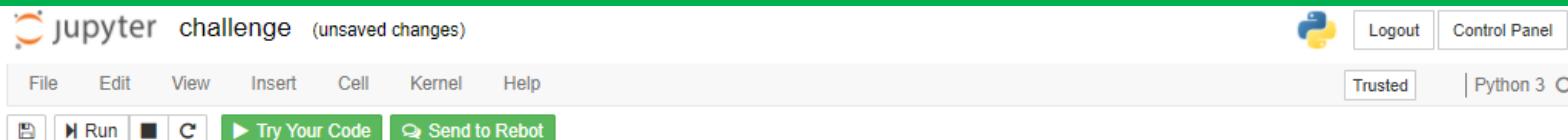
- Header:** jupyter, Logout, Control Panel
- Navigation:** Files (selected), Running, Clusters
- Message Bar:** Select items to perform actions on them.
- Toolbar:** Upload, New, Refresh
- File List:** chatbot\_home/
  - challenge.ipynb (highlighted with a red box and arrow)
  - practice.ipynb
- Sort Options:** Name (down), Last Modified
- File Details:**
  - challenge.ipynb: seconds ago, Running 17 hours ago
  - practice.ipynb: 4 days ago

**Text Overlay:** Click on challenge.ipynb to start the challenge!

- Teams of 3
- Complete challenges in sequence

# Instructions

04 Challenge



## Rebot Challenge: Webscraping Real-time Data from Yahoo Finance

For this challenge, your team will be web-scraping dynamically 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](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

04 Challenge

## 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>
```



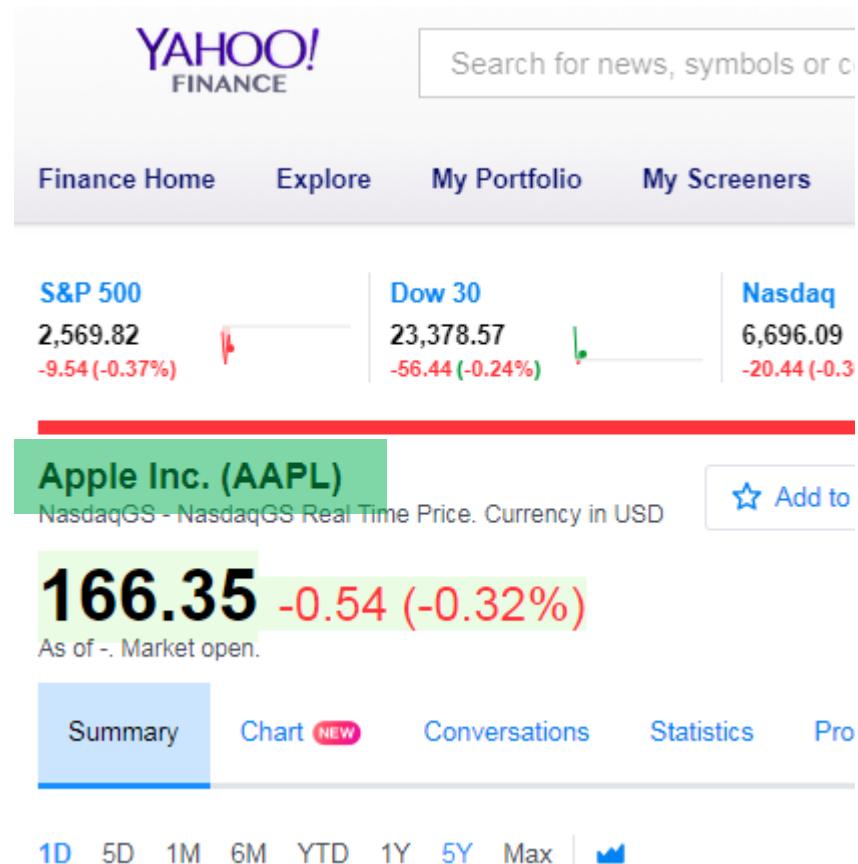
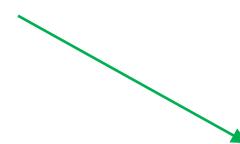
- These values are enclosed in tags (think of containers).
- They are the basic building blocks needed used to build basic to complex websites.

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

# Hint A

04 Challenge

```
<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

## 04 Challenge

```
<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("td", "data-test", " BETA-value")
```

# Good Luck!



**HAVE  
FUN  
AND  
KEEP  
SMILING**

# Team Challenge Round

04 Challenge

\*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  $\geq 1.5$  – “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 03: Taming Python

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

[OK, LET'S GO!](#)

## 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 04: Rebot Challenge

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

[CHALLENGE ACCEPTED!](#)

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.



```
1 # We are field testing a new battle unit: the decoy.
2 # Build 4 decoys, then report the total to Naria.
3 decoysBuilt = 0
4 loop:
5     coin = self.findNearest(self.findItems())
6     if coin:
7         # Loot the coin!
8         self.moveXY(coin.pos.x, coin.pos.y)
9
10    # Each decoy costs 25 gold.
11    # Know when you have more than 25 gold with self.gold
12    if self.gold >= 25:
13        self.buildXY("decoy", self.pos.x, self.pos.y)
14        # Keep a count of decoys you built as you go along.
15        decoysBuilt += 1
16
17    if decoysBuilt == 4:
18        # Break out of the loop when you have built 4.
19        break
20
21    self.say("Done building decoys!")
22    self.moveXY(14, 36)
23    # Go to Naria and say how many decoys you built.
24    self.say("I built " + decoysBuilt + " decoys!")
25
26
```

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!

The image shows the Scratch 3.0 interface with a project titled "Full 16 Frame Scratch Cat Walk Cycle" by griftpatch. The stage features a smiling orange cat sprite walking through a grassy field with tall green reeds and a small tree. The script editor on the right displays two scripts:

- Script 1 (Top):** A forever loop that waits 0.02 seconds and then changes the costume.
- Script 2 (Bottom):** A script triggered by the cat's own click. It sets size to 50%, goes to coordinates (-195, -123), clears the stage, switches to costume "Walk1", and repeats 8 times to stamp 8 copies of itself, changing x by 55 each time. After the repeat, it sets size to 120% and goes to (0, 55).

The bottom left corner shows the sprite library with the "Scratch Cat" sprite selected. The stage backdrop is "Stage 1 backdrop".

# 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 38<sup>th</sup> Auditorium.



**IMAGINE**



**PITCH**