



# Welcome to the Riot API Bootcamp!

*KNOW MORE, WIN MORE.*

# RIOT API BOOTCAMP SYLLABUS

## 1. Basics (Python, GitHub, Notepad++)

1. Resources to get started
2. Setting up an environment
3. Downloading GitHub repos
4. JSON explanation & Notepad++ example
5. *Project: read csv file, convert to data frame, create graphs*

## 2. Riot API introduction

1. What is an API?
2. Getting access & Registering your App
3. What end points are there/what data is available?
4. Explanation of puuid/account name
5. *Project: make an API call on the website & download the data*

## 3. Automating API interactions

1. Introduction to libraries (Cassiopeia, Riot Watcher)
2. Getting help (documentation, Discord)
3. *Project: automate an API call using a library*

## 4. Single Endpoint Data

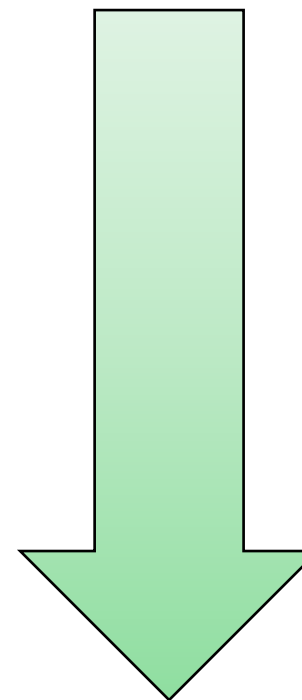
1. Use case explanation (e.g., in-depth match analysis, leaderboards)
2. Code example- getting challenger leaderboard
3. *Project: request last 25 games for an account and determine the most common champion(s)*

## 5. Large Scale Data Collection

1. Use case explanation (e.g., match history of top 50 players)
2. Setting up a process pipeline
3. Comparing 1 file approach vs. functions across files approach
4. *Project: determine number of roles (TOP, MID, etc) on the challenger ladder using the last 5 games*

**5 Modules** covering core topics

**Project** at the end of each



# Module 3: Automating API interactions

*RIOT API BOOTCAMP*

Slide Deck





# MODULE 3: AUTOMATION

## 3. Automating API interactions

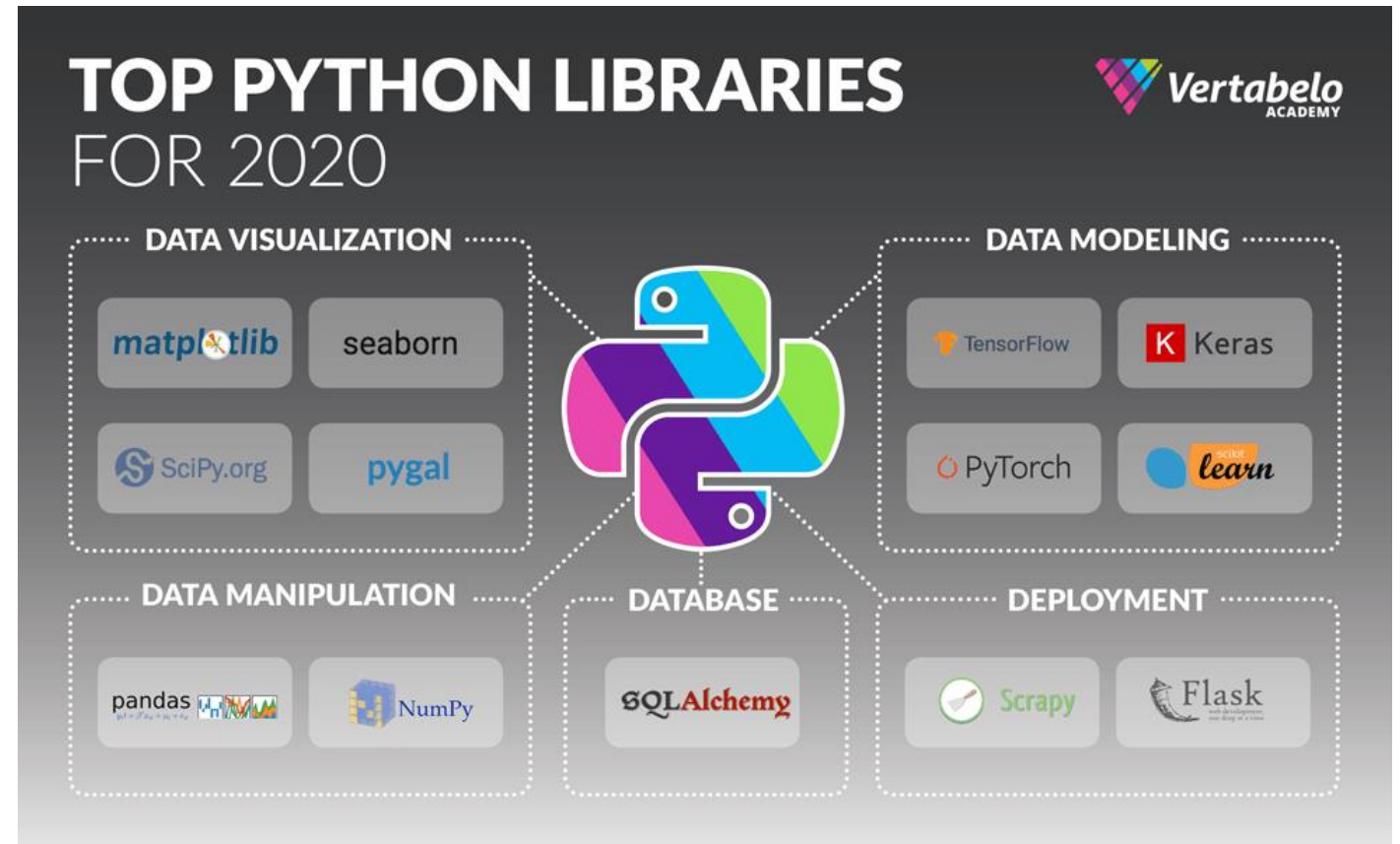
1. Introduction to libraries (Cassiopeia, Riot Watcher)
2. Getting help (documentation, Discord)
3. *Project: automate an API call using a library*

LET'S DIVE IN



# WHAT ARE PYTHON LIBRARIES?

- A library is a collection of books or is a room or place where many books are stored to be used later.
- Similarly, in the programming world, a library is a collection of precompiled codes that can be used later in a program for some specific well-defined operations.
- **Example:** *pandas* let us preform a bunch of operations on a dataset



# LIBRARIES FOR THE RIOT API

---

- Thankfully, people have written libraries that allow us to automate Riot API interactions!
- Each programming language has its own version
- Simplifies complicated requests into short code bits

Getting champion mastery with the *Cassiopeia* library

```
kalturi = Summoner(name="Kalturi")
good_with = kalturi.champion_masteryes.filter(lambda cm: cm.level >= 6)
print([cm.champion.name for cm in good_with])

# At the time of writing this, this prints:
["Vel'Koz", 'Blitzcrank', 'Braum', 'Lulu', 'Sejuani']
```

Checking competitive match rank with *Riot Watch* library

```
me = lol_watcher.summoner.by_name(my_region, 'pseudonym117')
print(me)

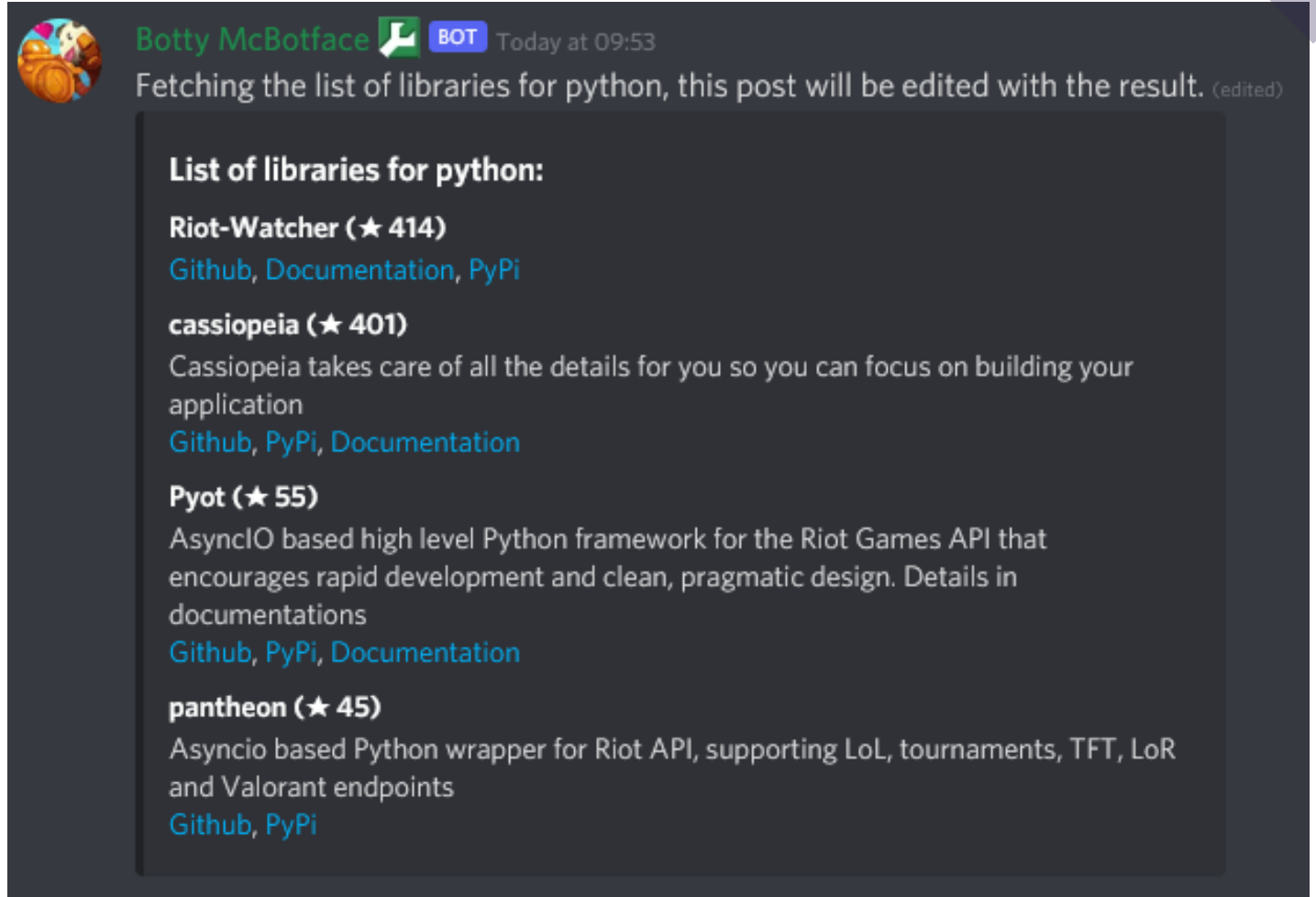
# all objects are returned (by default) as a dict
# lets see if i got diamond yet (i probably didnt)
my_ranked_stats = lol_watcher.league.by_summoner(my_region, me['id'])
print(my_ranked_stats)

# First we get the latest version of the game from data dragon
versions = lol_watcher.data_dragon.versions_for_region(my_region)
champions_version = versions['n']['champion']

# Lets get some champions
current_champ_list = lol_watcher.data_dragon.champions(champions_version)
print(current_champ_list)
```

# POPULAR RIOT API PYTHON LIBRARIES

- *Riot Watcher*
  - Least buggy
  - Supports different Riot games
  - <https://github.com/pseudonym117/Riot-Watcher>
- *Cassiopeia*
  - Easiest to use
  - Web-based integrations
  - <https://github.com/meraki-analytics/cassiopeia>
- *Pyot*
  - AsyncIO based
  - Does not support DDragon
  - <https://github.com/paaksing/Pyot>
- *Pantheon*
  - Simpler Pyot
  - Supports most endpoints
  - <https://github.com/Canisback/pantheon>



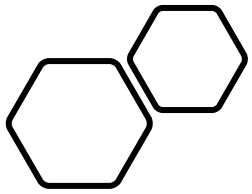
A screenshot of a Discord message from a user named 'Botty McBotface' with a bot icon. The message is timestamped 'Today at 09:53' and contains a status 'BOT'. The message text says 'Fetching the list of libraries for python, this post will be edited with the result. (edited)'. Below the text is a dark-themed box containing a list of Python libraries for the Riot API. The list includes 'Riot-Watcher' (414 stars), 'cassiopeia' (401 stars), 'Pyot' (55 stars), and 'pantheon' (45 stars). Each entry includes a brief description and links to its GitHub, PyPi, and Documentation pages.

**Botty McBotface** BOT Today at 09:53  
Fetching the list of libraries for python, this post will be edited with the result. (edited)

**List of libraries for python:**

- Riot-Watcher (★ 414)**  
[Github](#), [Documentation](#), [PyPi](#)
- cassiopeia (★ 401)**  
Cassiopeia takes care of all the details for you so you can focus on building your application  
[Github](#), [PyPi](#), [Documentation](#)
- Pyot (★ 55)**  
AsyncIO based high level Python framework for the Riot Games API that encourages rapid development and clean, pragmatic design. Details in documentations  
[Github](#), [PyPi](#), [Documentation](#)
- pantheon (★ 45)**  
Asyncio based Python wrapper for Riot API, supporting LoL, tournaments, TFT, LoR and Valorant endpoints  
[Github](#), [PyPi](#)





# WHERE TO GET HELP

- Each library has its own documentation explaining what different functions do
- The code documentation also has examples!
- Because the Riot API community is relatively small, searching the internet if you have a problem doesn't really help.
- Check out the Riot Games Developer discord server: <https://discord.gg/riotgamesdevrel>
- You can also reach out to me or others in the community if you have questions!

## RiotWatcher

### Navigation

[League of Legends  
Watcher](#)  
[Legends Of Runeterra  
Watcher](#)  
[Riot Watcher](#)  
[Team Fight Tactics  
Watcher](#)  
[Valorant Watcher](#)  
[Handlers](#)  
[Testing](#)

### Quick search

## Welcome to RiotWatcher's documentation!

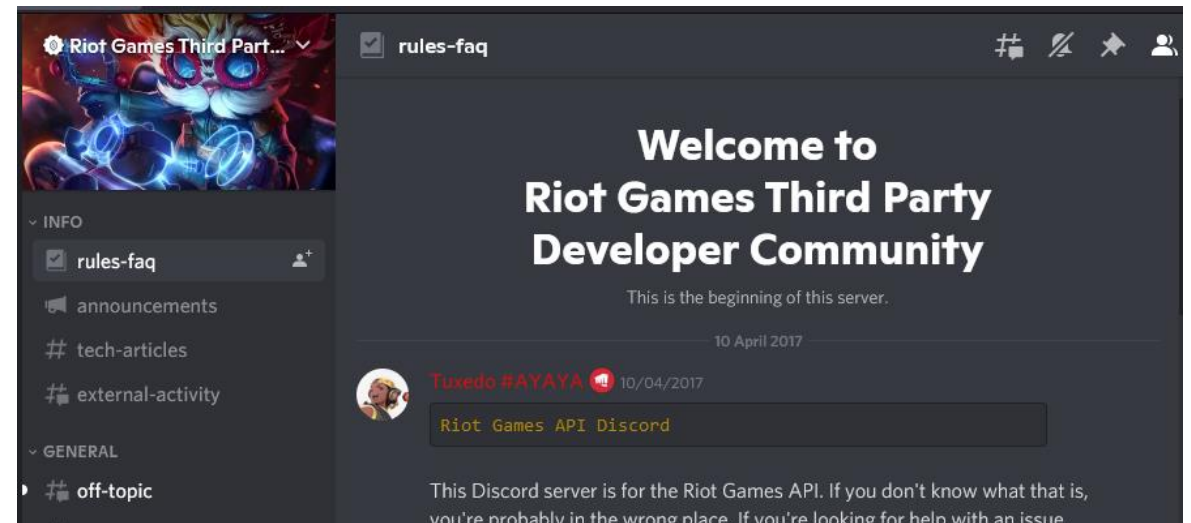
RiotWatcher is a thin wrapper on top of the [Riot Games API for League of Legends](#). All public methods as of 7/4/2021 are supported in full.

RiotWatcher by default supports a naive rate limiter. This rate limiter will try to stop you from making too many requests, and in a single threaded test environment does this rather well. In a multithreaded environment, you may still get some 429 errors. 429 errors are currently NOT retried for you.

## To Start...

To install RiotWatcher:

```
pip install riotwatcher
```





# AUTOMATING AN API REQUEST

## DEMONSTRATION with

/riot/account/v1/accounts/by-riot-id/{gameName}/{tagLine}

*Same as Module 2, but now with code*

GET

/riot/account/v1/accounts/by-riot-id/{gameName}/{tagLine}

Get account by riot id

Jump to Inputs

RESPONSE CLASSES

Return value: AccountDto

AccountDto

NAME	DATA TYPE	DESCRIPTION
puuid	string	
gameName	string	This field may be excluded from the response if the account doesn't have a gameName.
tagLine	string	This field may be excluded from the response if the account doesn't have a tagLine.

# QUESTIONS?

Contact me



RebirthNA#2359



@LoL-Genius



417devops@gmail.com

It is my hope that this course is easy to understand and follow

Have a question or want additional details?  
Just reach out!

***If you want to know more about my work (LoL Genius) or have questions about something you're building, LMK!***