



# 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

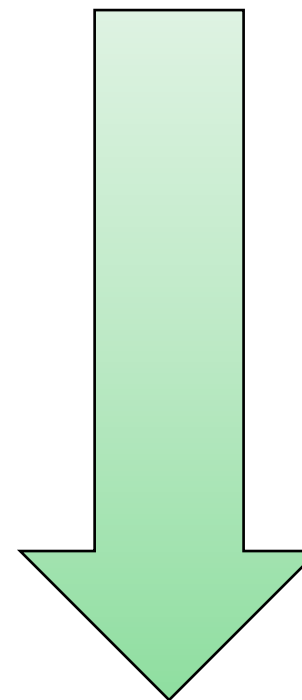
1. Use case explanation (e.g., in-depth match analysis, leaderboards)
2. Code example- getting challenger leaderboard
3. *Project: request last 25 games for an account and determine the most common champion(s)*

## 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 5: Large Scale Data Collection

*RIOT API BOOTCAMP*

Slide Deck





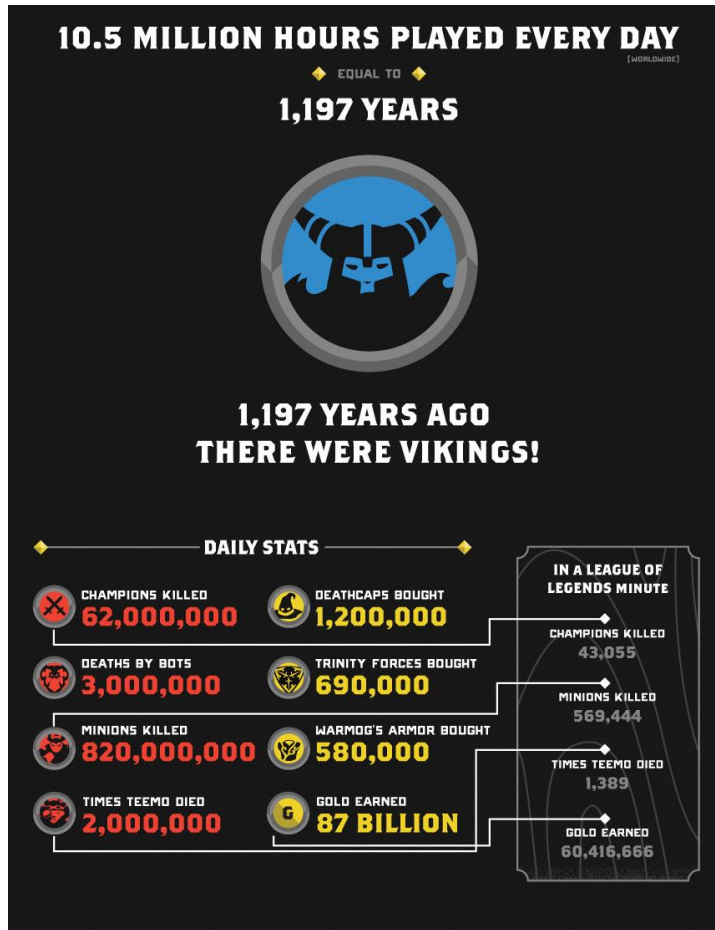
# MODULE 5: LARGE SCALE DATA COLLECTION

## 5. Single Endpoint Data

1. Use case explanation
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*

LET'S DIVE IN

# WHAT IS LARGE SCALE DATA COLLECTION?



This is what Riot released in 2016.  
League now has ~180 million monthly players,  
more than 10x the above info

## Previously we looked at small data

- 1 account
- 1-2 end points

## What if we wanted to get an idea of large trends?

- What champions are popular?
- How are different roles represented in different ranks?
- Is there a new build?

## This is a lot of data!

## Just how big is this?

- 2020 World Championship peaked at **46,067,896 viewers**
- Getting the last 20 games for these players is **921 million games**
- 1 match stat file takes up about 115 KB, storing the game data would take up **101,000 GB**
- If each request takes 0.5 s, that will take **14.6 years**

# PROCESS PIPELINE

*Because we have so much data, we need a clean process to automatically get the information!*

## **Define the relevant data:** *What data do I want to collect?*

- List of players
- List of matches
- Pro matches
- Timeframe

## **Identify API endpoints:** *Where is the data coming from?*

- Account-V1
- League-V4
- Match-V5
- Etc.

## **Determine the data flow:** *In what order am I collecting the data?*

- Account info
- Match history
- Match stats

## **Check for efficiency improvements:** *How can I minimize the amount of data requested?*

- Info by match instead of player
- Discarding useless info
- Preload static files

# MEMORY EFFICIENCY EXAMPLE

/lol/match/v5/matches/{matchId}

- From Project 4
- Returns MatchDTO object
- Looking at game time



# SETTING UP EXTERNAL FUNCTIONS

- *Request account info*
  - *Request match history*
  - *Request match info for each match*
- [/riot/account/v1/accounts/by-riot-id/{gameName}/{tagLine}](#)
  - [/lol/match/v5/matches/by-puuid/{puuid}/ids](#)
  - [/lol/match/v5/matches/{matchId}](#)

There are three routing values for account-v1; americas, asia, and europe. You can query for any account in any region. We recommend using the nearest cluster.

## ACCOUNT-V1

[Collapse Operations](#) | [Expand Operations](#)

GET	<a href="#">/riot/account/v1/accounts/by-puuid/{puuid}</a>	<a href="#">Get account by puuid</a>
GET	<a href="#">/riot/account/v1/accounts/by-riot-id/{gameName}/{tagLine}</a>	<a href="#">Get account by riot id</a>
GET	<a href="#">/riot/account/v1/active-shards/by-game/{game}/by-puuid/{puuid}</a>	<a href="#">Get active shard for a player</a>
GET	<a href="#">/riot/account/v1/accounts/me</a>	<a href="#">Get account by access token</a>

The AMERICAS routing value serves NA, BR, LAN, LAS, and OCE. The ASIA routing value serves KR and JP. The EUROPE routing value serves EUNE, EUW, TR, and RU.

## MATCH-V5

[Collapse Operations](#) | [Expand Operations](#)

GET	<a href="#">/lol/match/v5/matches/by-puuid/{puuid}/ids</a>	<a href="#">Get a list of match ids by puuid</a>
GET	<a href="#">/lol/match/v5/matches/{matchId}</a>	<a href="#">Get a match by match id</a>
GET	<a href="#">/lol/match/v5/matches/{matchId}/timeline</a>	<a href="#">Get a match timeline by match id</a>



# 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!***