RiotWatcher Documentation

Release 3.1.0

pseudonym117

Sep 01, 2020

Contents

RiotWatcher is a thin wrapper on top of the Riot Games API for League of Legends. All public methods as of 9/1/2020 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
```

OR:

```
python setup.py install
```

You also need to have an API key from Riot. Get that from here.

Using it...

All methods return dictionaries representing the json objects described by the official Riot API. Any HTTP errors that are returned by the API are raised as HTTPError exceptions from the Requests library.

```python
from riotwatcher import LolWatcher, ApiError

lol_watcher = LolWatcher('<your-api-key>')

my_region = 'na1'

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)

# For Riot's API, the 404 status code indicates that the requested data wasn't found
# and
# should be expected to occur in normal operation, as in the case of a an
# invalid summoner name, match ID, etc.
#
# The 429 status code indicates that the user has sent too many requests
# in a given amount of time ("rate limiting").

try:
```

```python
    response = lol_watcher.summoner.by_name(my_region, 'this_is_probably_not_anyones_
→summoner_name')
except ApiError as err:
    if err.response.status_code == 429:
        print('We should retry in {} seconds.'.format(err.headers['Retry-After']))
        print('this retry-after is handled by default by the RiotWatcher library')
        print('future requests wait until the retry-after time passes')
    elif err.response.status_code == 404:
        print('Summoner with that ridiculous name not found.')
    else:
        raise
```

Use with kernel

RiotWatcher can integrate with the API proxy/caching server kernel. This can be done by providing the `kernel_url` parameter to the `LolWatcher` constructor.

```python
from riotwatcher import LolWatcher, ApiError

lol_watcher = LolWatcher(kernel_url="https://your-kernel-instance") # should not
↪contain trailing slash
# use watcher as normal
```

Main API and other topics

4.1 League of Legends Watcher

class riotwatcher.**LolWatcher**(*api_key: str = None*, *timeout: int = None*, *kernel_url: str = None*, *rate_limiter: riotwatcher.RateLimiter.RateLimiter = <riotwatcher.Handlers.RateLimit.BasicRateLimiter.BasicRateLimiter object>*, *deserializer: riotwatcher.Deserializer.Deserializer = <riotwatcher.Handlers.DictionaryDeserializer.DictionaryDeserializer object>*)

LolWatcher class is intended to be the main interaction point with the APIs for League of Legends.

Initialize a new instance of the RiotWatcher class.

> **Parameters**
>
> - **api_key** (`string`) – the API key to use for this instance
>
> - **timeout** (`int`) – Time to wait for a response before timing out a connection to the Riot API
>
> - **kernel_url** (`string`) – URL for the kernel instance to connect to, instead of the API. See https://github.com/meraki-analytics/kernel for details.
>
> - **rate_limiter** (`RateLimiter`) – Instance to be used for rate limiting. This defaults to Handlers.RateLimit.BasicRateLimiter. This parameter is not used when connecting to a kernel instance.
>
> - **deserializer** (`Deserializer`) – Instance to be used to deserialize responses from the Riot Api. Default is Handlers.DictionaryDeserializer.

champion
> Interface to the Champion Endpoint
>
> > **Return type** *league_of_legends.ChampionApiV3*

champion_mastery
> Interface to the ChampionMastery Endpoint

> > > **Return type** *league_of_legends.ChampionMasteryApiV4*

> **clash**
>> Interface to the Clash Endpoint
>>
>>> **Return type** *league_of_legends.ClashApiV1*

> **data_dragon**
>> Interface to the DataDragon Endpoint
>>
>>> **Return type** *league_of_legends.DataDragonApi*

> **league**
>> Interface to the League Endpoint
>>
>>> **Return type** *league_of_legends.LeagueApiV4*

> **lol_status**
>> Interface to the LoLStatus Endpoint
>>
>>> **Return type** *league_of_legends.LolStatusApiV3*

> **match**
>> Interface to the Match Endpoint
>>
>>> **Return type** *league_of_legends.MatchApiV4*

> **spectator**
>> Interface to the Spectator Endpoint
>>
>>> **Return type** *league_of_legends.SpectatorApiV4*

> **summoner**
>> Interface to the Summoner Endpoint
>>
>>> **Return type** *league_of_legends.SummonerApiV4*

> **third_party_code**
>> Interface to the Third Party Code Endpoint
>>
>>> **Return type** *league_of_legends.ThirdPartyCodeApiV4*

4.1.1 All APIs

ChampionApiV3

class riotwatcher._apis.league_of_legends.**ChampionApiV3**(*base_api:* *riot-watcher._apis.BaseApi.BaseApi*)

> This class wraps the Champion-v3 Api calls provided by the Riot API.
>
> See https://developer.riotgames.com/api-methods/#champion-v3 for more detailed information
>
> Initialize a new ChampionApiV3 which uses the provided base_api
>
>> **Parameters** **base_api** (*BaseApi*) – the root API object to use for making all requests.

rotations(*region: str*)

> Returns champion rotations, including free-to-play and low-level free-to-play rotations.
>
>> **Returns** ChampionInfo

ChampionMasteryApiV4

class riotwatcher._apis.league_of_legends.**ChampionMasteryApiV4**(*base_api: riot-watcher._apis.BaseApi.BaseApi*)

This class wraps the Champion-Mastery-v4 Api calls provided by the Riot API.

See https://developer.riotgames.com/api-methods/#champion-mastery-v4/ for more detailed information

Initialize a new ChampionMasteryApiV4 which uses the provided base_api

> **Parameters base_api** (`BaseApi`) – the root API object to use for making all requests.

by_summoner(*region: str*, *encrypted_summoner_id: str*)
Get all champion mastery entries.

> **Parameters**
>
>> • **region** (`string`) – the region to execute this request on
>>
>> • **encrypted_summoner_id** (`string`) – Summoner ID associated with the player
>
> **Returns** List[ChampionMasteryDTO]: This object contains a list of Champion Mastery information for player and champion combination.

by_summoner_by_champion(*region: str*, *encrypted_summoner_id: str*, *champion_id: int*)
Get a champion mastery by player ID and champion ID.

> **Parameters**
>
>> • **region** (`string`) – the region to execute this request on
>>
>> • **encrypted_summoner_id** (`string`) – Summoner ID associated with the player
>>
>> • **champion_id** (`long`) – Champion ID to retrieve Champion Mastery for
>
> **Returns** ChampionMasteryDTO: This object contains single Champion Mastery information for player and champion combination.

scores_by_summoner(*region: str*, *encrypted_summoner_id: str*)
Get a player's total champion mastery score, which is the sum of individual champion mastery levels

> **Parameters**
>
>> • **region** (`string`) – the region to execute this request on
>>
>> • **encrypted_summoner_id** (`string`) – Summoner ID associated with the player
>
> **Returns** int

ClashApiV1

class riotwatcher._apis.league_of_legends.**ClashApiV1**(*base_api: riot-watcher._apis.BaseApi.BaseApi*)

This class wraps the Clash-v1 endpoint calls provided by the Riot API.

See https://developer.riotgames.com/apis#clash-v1 for more detailed information

Initialize a new ClashApiV1 which uses the provided base_api

> **Parameters base_api** (`BaseApi`) – the root API object to use for making all requests.

by_summoner(*region: str*, *summoner_id: str*)
This endpoint returns a list of active Clash players for a given summoner ID. If a summoner registers for multiple tournaments at the same time (e.g., Saturday and Sunday) then both registrations would appear in this list.

> **Parameters**
>
> - **region** (*string*) – The region to execute this request on.
>
> - **summoner_id** (*string*) – The summoner ID.
>
> **Returns** List[PlayerDTO]: represents the summoner's info for the current clash.

by_team(*region: str*, *team_id: str*)
> Get team by ID.
>
> **Parameters**
>
> - **region** (*string*) – The region to execute this request on
>
> - **team_id** (*string*) – Team ID
>
> **Returns** TeamDTO: represents a clash team

by_tournament(*region: str*, *tournament_id: str*)
> Get tournament by ID.
>
> **Parameters**
>
> - **region** (*string*) – The region to execute this request on
>
> - **tournament_id** (*string*) – Tournament ID
>
> **Returns** TournamentDTO: represents a clash tournament

tournament_by_team(*region: str*, *team_id: str*)
> Get tournament by team ID.
>
> **Parameters**
>
> - **region** (*string*) – The region to execute this request on
>
> - **team_id** (*string*) – Team ID
>
> **Returns** TournamentDTO: represents a clash tournament

tournaments(*region: str*)
> Returns a list of active and upcoming tournaments.
>
> **Parameters region** (*string*) – The region to execute this request on
>
> **Returns** List[TournamentDTO]: represents all of the current tournaments active

DataDragonApi

class riotwatcher._apis.league_of_legends.**DataDragonApi**(*base_api: riotwatcher._apis.BaseApi.BaseApi*)

> **champions**(*version: str*, *full: bool = False*, *locale=None*)
>
> **items**(*version: str*, *locale: str = None*)
>
> **languages**(*version: str*, *locale: str = None*)
>
> **maps**(*version: str*, *locale: str = None*)
>
> **masteries**(*version: str*, *locale: str = None*)
>
> **profile_icons**(*version: str*, *locale: str = None*)
>
> **runes**(*version: str*, *locale: str = None*)

runes_reforged(*version: str*, *locale: str = None*)

summoner_spells(*version: str*, *locale: str = None*)

versions_for_region(*region: str*)

LeagueApiV4

class riotwatcher._apis.league_of_legends.**LeagueApiV4**(*base_api: riot-watcher._apis.BaseApi.BaseApi*)

This class wraps the League-v4 Api calls provided by the Riot API.

See https://developer.riotgames.com/api-methods/#league-v4/ for more detailed information

Initialize a new LeagueApiV4 which uses the provided base_api

> **Parameters base_api** (`BaseApi`) – the root API object to use for making all requests.

by_id(*region: str*, *league_id: str*)

Get league with given ID, including inactive entries

> **Parameters**
>
> - **region** (`string`) – the region to execute this request on
>
> - **league_id** (`string`) – the league ID to query
>
> **Returns** LeagueListDTO

by_summoner(*region: str*, *encrypted_summoner_id: str*)

Get league entries in all queues for a given summoner ID

> **Parameters**
>
> - **region** (`string`) – the region to execute this request on
>
> - **encrypted_summoner_id** (`string`) – the summoner ID to query
>
> **Returns** Set[LeagueEntryDTO]

challenger_by_queue(*region: str*, *queue: str*)

Get the challenger league for a given queue.

> **Parameters**
>
> - **region** (`string`) – the region to execute this request on
>
> - **queue** (`string`) – the queue to get the challenger players for
>
> **Returns** LeagueListDTO

entries(*region: str*, *queue: str*, *tier: str*, *division: str*, *page: int = 1*)

Get all the league entries

> **Parameters**
>
> - **region** (`string`) – the region to execute this request on
>
> - **queue** (`string`) – the queue to query, i.e. RANKED_SOLO_5x5
>
> - **tier** (`string`) – the tier to query, i.e. DIAMOND
>
> - **division** (`string`) – the division to query, i.e. III
>
> - **page** (`int`) – the page for the query to paginate to. Starts at 1.
>
> **Returns** Set[LeagueEntryDTO]

grandmaster_by_queue(*region: str*, *queue: str*)
　　Get the grandmaster league for a given queue.

　　　　Parameters

　　　　　　• **region** (*string*) – the region to execute this request on

　　　　　　• **queue** (*string*) – the queue to get the grandmaster players for

　　　　Returns LeagueListDTO

masters_by_queue(*region: str*, *queue: str*)
　　Get the master league for a given queue.

　　　　Parameters

　　　　　　• **region** (*string*) – the region to execute this request on

　　　　　　• **queue** (*string*) – the queue to get the master players for

　　　　Returns LeagueListDTO

LolStatusApiV3

class riotwatcher._apis.league_of_legends.**LolStatusApiV3**(*base_api: riot-watcher._apis.BaseApi.BaseApi*)
　　This class wraps the LoL-Status-v3 Api calls provided by the Riot API.

　　See https://developer.riotgames.com/api-methods/#lol-status-v3 for more detailed information

　　Initialize a new LolStatusApiV3 which uses the provided base_api

　　　　Parameters base_api (*BaseApi*) – the root API object to use for making all requests.

shard_data(*region: str*)
　　Get League of Legends status for the given shard.

　　Requests to this API are not counted against the application Rate Limits.

　　　　Parameters region (*string*) – the region to execute this request on

　　　　Returns ShardStatus

MatchApiV4

class riotwatcher._apis.league_of_legends.**MatchApiV4**(*base_api: riot-watcher._apis.BaseApi.BaseApi*)
　　This class wraps the Match-v4 endpoint calls provided by the Riot API.

　　See https://developer.riotgames.com/api-methods/#match-v4 for more detailed information

　　Initialize a new MatchApiV4 which uses the provided base_api

　　　　Parameters base_api (*BaseApi*) – the root API object to use for making all requests.

by_id(*region: str*, *match_id: int*)
　　Get match by match ID

　　　　Parameters

　　　　　　• **region** (*string*) – The region to execute this request on

　　　　　　• **match_id** (*long*) – The match ID.

　　　　Returns MatchDto

matchlist_by_account(*region: str*, *encrypted_account_id: str*, *queue: Set[str] = None*, *begin_time: int = None*, *end_time: int = None*, *begin_index: int = None*, *end_index: int = None*, *season: Set[int] = None*, *champion: Set[int] = None*)

Get matchlist for ranked games played on given account ID and platform ID and filtered using given filter parameters, if any

A number of optional parameters are provided for filtering. It is up to the caller to ensure that the combination of filter parameters provided is valid for the requested account, otherwise, no matches may be returned.

Note that if either beginIndex or endIndex are specified, then both must be specified and endIndex must be greater than beginIndex.

If endTime is specified, but not beginTime, then beginTime is effectively the start of the account's match history.

If beginTime is specified, but not endTime, then endTime is effectively the current time.

Note that endTime should be greater than beginTime if both are specified, although there is no maximum limit on their range.

> **Parameters**
> - **region** (`string`) – The region to execute this request on
> - **encrypted_account_id** (`string`) – The account ID.
> - **queue** (`Set[int]`) – Set of queue IDs for which to filtering matchlist.
> - **begin_time** (`long`) – The begin time to use for filtering matchlist specified as epoch milliseconds.
> - **end_time** (`long`) – The end time to use for filtering matchlist specified as epoch milliseconds.
> - **begin_index** (`int`) – The begin index to use for filtering matchlist.
> - **end_index** (`int`) – The end index to use for filtering matchlist.
> - **season** (`Set[int]`) – Set of season IDs for which to filtering matchlist.
> - **champion** (`Set[int]`) – Set of champion IDs for which to filtering matchlist.
>
> **Returns** MatchlistDto

timeline_by_match(*region: str*, *match_id: int*)

Get match timeline by match ID.

Not all matches have timeline data.

> **Parameters**
> - **region** (`string`) – The region to execute this request on
> - **match_id** (`long`) – The match ID.
>
> **Returns** MatchTimelineDto

SpectatorApiV4

class riotwatcher._apis.league_of_legends.**SpectatorApiV4**(*base_api: riotwatcher._apis.BaseApi.BaseApi*)

This class wraps the Spectator-v4 endpoint calls provided by the Riot API.

See https://developer.riotgames.com/api-methods/#spectator-v4 for more detailed information

Initialize a new SpectatorApiV3 which uses the provided base_api

> **Parameters base_api** (*BaseApi*) – the root API object to use for making all requests.

by_summoner (*region: str*, *encrypted_summoner_id: str*)
> Get current game information for the given summoner ID

> > **Parameters**

> > - **region** (*string*) – The region to execute this request on

> > - **encrypted_summoner_id** (*string*) – The ID of the summoner.

> > **Returns** CurrentGameInfo

featured_games (*region: str*)
> Get list of featured games.

> > **Parameters region** (*string*) – The region to execute this request on

> > **Returns** FeaturedGames

SummonerApiV4

class riotwatcher._apis.league_of_legends.**SummonerApiV4** (*base_api:* *riot-watcher._apis.BaseApi.BaseApi*)
> This class wraps the Summoner-v4 endpoint calls provided by the Riot API.

> See https://developer.riotgames.com/api-methods/#summoner-v4 for more detailed information

> Initialize a new SummonerApiV4 which uses the provided base_api

> > **Parameters base_api** (*BaseApi*) – the root API object to use for making all requests.

by_account (*region: str*, *encrypted_account_id: str*)
> Get a summoner by account ID.

> > **Parameters**

> > - **region** (*string*) – The region to execute this request on

> > - **encrypted_account_id** (*string*) – The account ID.

> > **Returns** SummonerDTO: represents a summoner

by_id (*region: str*, *encrypted_summoner_id: str*)
> Get a summoner by summoner ID.

> > **Parameters**

> > - **region** (*string*) – The region to execute this request on

> > - **encrypted_summoner_id** (*string*) – Summoner ID

> > **Returns** SummonerDTO: represents a summoner

by_name (*region: str*, *summoner_name: str*)
> Get a summoner by summoner name

> > **Parameters**

> > - **region** (*string*) – The region to execute this request on

> > - **summoner_name** (*string*) – Summoner Name

Returns SummonerDTO: represents a summoner

by_puuid(*region: str*, *encrypted_puuid: str*)

Get a summoner by PUUID.

> **Parameters**
>
> - **region** (`string`) – The region to execute this request on
>
> - **encrypted_puuid** (`string`) – PUUID
>
> **Returns** SummonerDTO: represents a summoner

ThirdPartyCodeApiV4

class riotwatcher._apis.league_of_legends.**ThirdPartyCodeApiV4**(*base_api: riot-watcher._apis.BaseApi.BaseApi*)

This class wraps the ThirdPartyCode-v4 Api calls provided by the Riot API.

See https://developer.riotgames.com/api-methods/#third-party-code-v4 for more detailed information

Initialize a new ThirdPartyCodeApiV4 which uses the provided base_api

> **Parameters** **base_api** (`BaseApi`) – the root API object to use for making all requests.

by_summoner(*region: str*, *encrypted_summoner_id: str*)

FOR KR SUMMONERS, A 404 WILL ALWAYS BE RETURNED.

Valid codes must be no longer than 256 characters and only use valid characters: 0-9, a-z, A-Z, and -

> **Parameters**
>
> - **region** (`string`) – the region to execute this request on
>
> - **encrypted_summoner_id** (`string`) – Summoner ID
>
> **Returns** string

4.2 Legends Of Runeterra Watcher

class riotwatcher.**LorWatcher**(*api_key: str = None*, *timeout: int = None*, *rate_limiter: riotwatcher.RateLimiter.RateLimiter = <riot-watcher.Handlers.RateLimit.BasicRateLimiter.BasicRateLimiter object>*, *deserializer: riotwatcher.Deserializer.Deserializer = <ri-otwatcher.Handlers.DictionaryDeserializer.DictionaryDeserializer object>*)

LorWatcher class is intended to be the main interaction point with the API for Legends of Runterra.

Initialize a new instance of the LorWatcher class.

> **Parameters**
>
> - **api_key** (`string`) – the API key to use for this instance
>
> - **timeout** (`int`) – Time to wait for a response before timing out a connection to the Riot API
>
> - **rate_limiter** (`RateLimiter`) – Instance to be used for rate limiting. This defaults to Handlers.RateLimit.BasicRateLimiter.
>
> - **deserializer** (`Deserializer`) – Instance to be used to deserialize responses from the Riot Api. Default is Handlers.DictionaryDeserializer.

ranked
> Interface to the Ranked Endpoint

>> **Return type** *legends_of_runeterra.RankedApi*

4.2.1 All APIs

RankedApi

class riotwatcher._apis.legends_of_runeterra.**RankedApi**(*base_api: riot-watcher._apis.BaseApi.BaseApi*)
> This class wraps the LoR-Ranked-V1 Api calls provided by the Riot API.

> See https://developer.riotgames.com/apis#lor-ranked-v1 for more detailed information

> Initialize a new RankedApi which uses the provided base_api

>> **Parameters** **base_api** (*BaseApi*) – the root API object to use for making all requests.

leaderboards(*region: str*)
> Get the players in Master tier.

>> **Returns** LeaderboardDto

4.3 Riot Watcher

class riotwatcher.**RiotWatcher**(*api_key: str*, *timeout: int = None*, *rate_limiter: riotwatcher.RateLimiter.RateLimiter   =   <riot-watcher.Handlers.RateLimit.BasicRateLimiter.BasicRateLimiter object>*, *deserializer: riotwatcher.Deserializer.Deserializer = <ri-otwatcher.Handlers.DictionaryDeserializer.DictionaryDeserializer object>*)
> RiotWatcher class is intended to be the main interaction point with the generic Riot APIs.

> Initialize a new instance of the RiotWatcher class.

>> **Parameters**

>> - **api_key** (*string*) – the API key to use for this instance

>> - **timeout** (*int*) – Time to wait for a response before timing out a connection to the Riot API

>> - **rate_limiter** (*RateLimiter*) – Instance to be used for rate limiting. This defaults to Handlers.RateLimit.BasicRateLimiter.

>> - **deserializer** (*Deserializer*) – Instance to be used to deserialize responses from the Riot Api. Default is Handlers.DictionaryDeserializer.

account
> Interface to the Account Endpoint

>> **Return type** *riot.AccountApi*

4.3.1 All APIs

AccountApi

class riotwatcher._apis.riot.**AccountApi**(*base_api: riotwatcher._apis.BaseApi.BaseApi*)

    This class wraps the Account-v1 Api calls provided by the Riot API.

    See https://developer.riotgames.com/apis#account-v1 for more detailed information

    Initialize a new AccountApi which uses the provided base_api

        Parameters base_api (`BaseApi`) – the root API object to use for making all requests.

active_shard(*region: str*, *game: str*, *puuid: str*)

    Get active shard for a player

        Returns ActiveShardDto

by_puuid(*region: str*, *puuid: str*)

    Get account by puuid

        Returns AcountDto

by_riot_id(*region: str*, *game_name: str*, *tag_line: str*)

    Get account by riot id

        Returns AccountDto

4.4 Team Fight Tactics Watcher

class riotwatcher.**TftWatcher**(*api_key: str = None*, *timeout: int = None*, *rate_limiter: riotwatcher.RateLimiter.RateLimiter = <riotwatcher.Handlers.RateLimit.BasicRateLimiter.BasicRateLimiter object>*, *deserializer: riotwatcher.Deserializer.Deserializer = <riotwatcher.Handlers.DictionaryDeserializer.DictionaryDeserializer object>*)

    TftWatcher class is intended to be the main interaction point with the APIs for Team Fight Tactics.

    Initialize a new instance of the TftWatcher class.

        Parameters

- **api_key** (`string`) – the API key to use for this instance
- **timeout** (`int`) – Time to wait for a response before timing out a connection to the Riot API
- **rate_limiter** (`RateLimiter`) – Instance to be used for rate limiting. This defaults to Handlers.RateLimit.BasicRateLimiter.
- **deserializer** (`Deserializer`) – Instance to be used to deserialize responses from the Riot Api. Default is Handlers.DictionaryDeserializer.

league

    Interface to the League Endpoint

        Return type *team_fight_tactics.LeagueApi*

match

    Interface to the Match Endpoint

        Return type *team_fight_tactics.MatchApi*

summoner
> Interface to the Summoner Endpoint
>
> > **Return type** *team_fight_tactics.SummonerApi*

4.4.1 All APIs

LeagueApi

class `riotwatcher._apis.team_fight_tactics.`**LeagueApi**(*base_api:* *riot-watcher._apis.BaseApi.BaseApi*)
> This class wraps the Tft-League-v1 Api calls provided by the Riot API.
>
> See https://developer.riotgames.com/apis#tft-league-v1 for more detailed information
>
> Initialize a new LeagueApi which uses the provided base_api
>
> > **Parameters** **base_api** (`BaseApi`) – the root API object to use for making all requests.
>
> **by_id**(*region: str*, *league_id: str*)
> > Get league with given ID, including inactive entries
> >
> > > **Returns** LeagueListDTO
>
> **by_summoner**(*region: str*, *encrypted_summoner_id: str*)
> > Get league entries for a given summoner ID
> >
> > > **Returns** Set[LeagueEntryDTO]
>
> **challenger**(*region: str*)
> > Get the challenger league
> >
> > > **Returns** LeagueListDTO
>
> **entries**(*region: str*, *tier: str*, *division: str*, *page: int = 1*)
> > Get all the league entries
> >
> > > **Returns** Set[LeagueEntryDTO]
>
> **grandmaster**(*region: str*)
> > Get the grandmaster league.
> >
> > > **Returns** LeagueListDTO
>
> **master**(*region: str*)
> > Get the master league
> >
> > > **Returns** LeagueListDTO

MatchApi

class `riotwatcher._apis.team_fight_tactics.`**MatchApi**(*base_api:* *riot-watcher._apis.BaseApi.BaseApi*)
> This class wraps the TFT-Match-v1 Api calls provided by the Riot API.
>
> See https://developer.riotgames.com/apis#tft-match-v1 for more detailed information
>
> Initializes a new MatchApi which uses the provided base_api
>
> > **Parameters** **base_api** (`BaseApi`) – the root API object to use for making all requests.
>
> **by_id**(*region: str*, *match_id: str*)
> > Get a match by match id.

> **Returns** MatchDto

by_puuid(*region: str*, *puuid: str*, *count: int = 20*)
> Get a list of match ids by PUUID.
>
> > **Returns** List[string]

SummonerApi

class riotwatcher._apis.team_fight_tactics.**SummonerApi**(*base_api: riot-watcher._apis.BaseApi.BaseApi*)
> This class wraps the TFT-Summoner-v1 Api calls provided by the Riot API.
>
> See https://developer.riotgames.com/apis#tft-summoner-v1 for more detailed information.
>
> Initializes a new SummonerApi which uses the provided base_api
>
> > **Parameters base_api** (`BaseApi`) – the root API object to use for making all requests.

by_account(*region: str*, *encrypted_account_id: str*)
> Get a summoner by account ID
>
> > **Returns** SummonerDTO

by_id(*region: str*, *encrypted_summoner_id: str*)
> Get a summoner by summoner ID
>
> > **Returns** SummonerDTO

by_name(*region: str*, *summoner_name: str*)
> Get a summoner by summoner name.
>
> > **Returns** SummonerDTO

by_puuid(*region: str*, *puuid: str*)
> Get a summoner by PUUID.
>
> > **Returns** SummonerDTO

4.5 Valorant Watcher

class riotwatcher.**ValWatcher**(*api_key: str,   timeout: int = None,   rate_limiter: riotwatcher.RateLimiter.RateLimiter   =   <riot-watcher.Handlers.RateLimit.BasicRateLimiter.BasicRateLimiter object>, deserializer: riotwatcher.Deserializer.Deserializer = <ri-otwatcher.Handlers.DictionaryDeserializer.DictionaryDeserializer object>*)
> ValWatcher class is intended to be the main interaction point with the API for Valorant.
>
> Initialize a new instance of the ValWatcher class.
>
> > **Parameters**
> >
> > - **api_key** (`string`) – the API key to use for this instance
> >
> > - **timeout** (`int`) – Time to wait for a response before timing out a connection to the Riot API
> >
> > - **rate_limiter** (`RateLimiter`) – Instance to be used for rate limiting. This defaults to Handlers.RateLimit.BasicRateLimiter.

- **deserializer** (*Deserializer*) – Instance to be used to deserialize responses from the Riot Api. Default is Handlers.DictionaryDeserializer.

content
> Interface to the Content Endpoint

>> **Return type** *valorant.ContentApi*

match
> Interface to the Match Endpoint

>> **Return type** *valorant.MatchApi*

4.5.1 All APIs

ContentApi

class riotwatcher._apis.valorant.**ContentApi**(*base_api: riot-watcher._apis.BaseApi.BaseApi*)
> This class wraps the Val-Content-v1 Api calls provided by the Riot API.

> See https://developer.riotgames.com/apis#val-content-v1 for more detailed information

> Initialize a new ContentApi which uses the provided base_api

>> **Parameters base_api** (*BaseApi*) – the root API object to use for making all requests.

> **contents**(*region: str*, *locale: str = None*)
>> Get content optionally filtered by locale

>>> **Returns** ContentDto

ValMatchApi

class riotwatcher._apis.valorant.**MatchApi**(*base_api: riotwatcher._apis.BaseApi.BaseApi*)
> This class wraps the Val-Match-v1 Api calls provided by the Riot API.

> See https://developer.riotgames.com/apis#val-match-v1 for more detailed information

> Initialize a new MatchApi which uses the provided base_api

>> **Parameters base_api** (*BaseApi*) – the root API object to use for making all requests.

> **by_id**(*region: str*, *match_id: str*)
>> Get match by id

>>> **Returns** MatchDto

> **matchlist_by_puuid**(*region: str*, *puuid: str*)
>> Get matchlist for games played by puuid

>>> **Returns** MatchlistDto

4.6 Handlers

4.6.1 Submodules

4.6.2 Rate Limit Modules

See rate-limiter for more info

class riotwatcher.Handlers.RateLimiterAdapter.**RateLimiterAdapter**(*limiter: riot-watcher.RateLimiter.RateLimiter*)

    Bases: *riotwatcher.Handlers.RequestHandler.RequestHandler*

    after_request(*region: str, endpoint_name: str, method_name: str, url: str, response: re-quests.models.Response*) → requests.models.Response
        Called after a response is received and before it is returned to the user.

        Parameters

- **region** (*string*) – the region of this request
- **endpoint_name** (*string*) – the name of the endpoint that was requested
- **method_name** (*string*) – the name of the method that was requested
- **url** – The url that was requested
- **response** – the response received. This is a response from the Requests library

    preview_request(*region: str, endpoint_name: str, method_name: str, url: str, query_params: dict*)
        called before a request is processed.

        Parameters

- **region** (*string*) – the region of this request
- **endpoint_name** (*string*) – the name of the endpoint being requested
- **method_name** (*string*) – the name of the method being requested
- **url** – the URL that is being requested.
- **query_params** – dict: the parameters to the url that is being queried, e.g. ?key1=val&key2=val2

4.6.3 Serialization Modules

class riotwatcher.Handlers.DictionaryDeserializer.**DictionaryDeserializer**
    Bases: riotwatcher.Deserializer.Deserializer

    deserialize(*endpoint_name: str, method_name: str, data: str*) → dict

class riotwatcher.Handlers.DeserializerAdapter.**DeserializerAdapter**(*deserializer: riot-watcher.Deserializer.Deserializer*)

    Bases: *riotwatcher.Handlers.RequestHandler.RequestHandler*

    after_request(*region: str, endpoint_name: str, method_name: str, url: str, response: re-quests.models.Response*) → Any
        Called after a response is received and before it is returned to the user.

        Parameters

- **region** (*string*) – the region of this request

- **endpoint_name** (*string*) – the name of the endpoint that was requested

- **method_name** (*string*) – the name of the method that was requested

- **url** – The url that was requested

- **response** – the response received. This is a response from the Requests library

after_static_request(*url: str*, *response: requests.models.Response*) → Any
   Called after a response is received and before it is returned to the user.

   Parameters

   - **url** – The url that was requested

   - **response** – the response received. This is a response from the Requests library

4.6.4 Error Handler Modules

class riotwatcher.Handlers.DeprecationHandler.**DeprecationHandler**
   Bases: *riotwatcher.Handlers.RequestHandler.RequestHandler*

   after_request(*region: str*, *endpoint_name: str*, *method_name: str*, *url: str*, *response: requests.models.Response*) → requests.models.Response
   Called after a response is received and before it is returned to the user.

   Parameters

   - **region** (*string*) – the region of this request

   - **endpoint_name** (*string*) – the name of the endpoint that was requested

   - **method_name** (*string*) – the name of the method that was requested

   - **url** – The url that was requested

   - **response** – the response received. This is a response from the Requests library

class riotwatcher.Handlers.ThrowOnErrorHandler.**ThrowOnErrorHandler**
   Bases: *riotwatcher.Handlers.RequestHandler.RequestHandler*

   after_request(*region: str*, *endpoint_name: str*, *method_name: str*, *url: str*, *response: requests.models.Response*) → None
   Called after a response is received and before it is returned to the user.

   Parameters

   - **region** (*string*) – the region of this request

   - **endpoint_name** (*string*) – the name of the endpoint that was requested

   - **method_name** (*string*) – the name of the method that was requested

   - **url** – The url that was requested

   - **response** – the response received. This is a response from the Requests library

class riotwatcher.Handlers.TypeCorrectorHandler.**TypeCorrectorHandler**
   Bases: *riotwatcher.Handlers.RequestHandler.RequestHandler*

   The TypeCorrector class is meant to correct any inconsistencies in the types of objects provided as query parameters.

   Currently this only involves changing boolean values into strings, as the API only accepts lower case booleans for some reason.

preview_request(*region: str*, *endpoint_name: str*, *method_name: str*, *url: str*, *query_params: dict*)
    called before a request is processed.

> **Parameters**
>
> - **endpoint_name** (`string`) – the name of the endpoint being requested
>
> - **method_name** (`string`) – the name of the method being requested
>
> - **url** – the URL that is being requested.
>
> - **query_params** – dict: the parameters to the url that is being queried, e.g. ?key1=val&key2=val2

4.6.5 Request Handler Modules

class `riotwatcher.Handlers.RequestHandler.`**RequestHandler**
    Bases: `object`

after_request(*region: str*, *endpoint_name: str*, *method_name: str*, *url: str*, *response*)
    Called after a response is received and before it is returned to the user.

> **Parameters**
>
> - **region** (`string`) – the region of this request
>
> - **endpoint_name** (`string`) – the name of the endpoint that was requested
>
> - **method_name** (`string`) – the name of the method that was requested
>
> - **url** – The url that was requested
>
> - **response** – the response received. This is a response from the Requests library

after_static_request(*url: str*, *response*)
    Called after a response is received and before it is returned to the user.

> **Parameters**
>
> - **url** – The url that was requested
>
> - **response** – the response received. This is a response from the Requests library

preview_request(*region: str*, *endpoint_name: str*, *method_name: str*, *url: str*, *query_params: dict*)
    called before a request is processed.

> **Parameters**
>
> - **region** (`string`) – the region of this request
>
> - **endpoint_name** (`string`) – the name of the endpoint being requested
>
> - **method_name** (`string`) – the name of the method being requested
>
> - **url** – the URL that is being requested.
>
> - **query_params** – dict: the parameters to the url that is being queried, e.g. ?key1=val&key2=val2

preview_static_request(*url: str*, *query_params: dict*)
    Called before a request to DataDragon is processed

> **Parameters** **url** – The url that was requested

4.7 Testing

Unit and integration tests can be run with the following command from the RiotWatcher folder:

```
tox
```

CHAPTER 5

Indices and tables

- genindex
- modindex
- search

Python Module Index

r

Index

A

AccountApi (*class in riotwatcher._apis.riot*), 19

active_shard() (*riotwatcher._apis.riot.AccountApi method*), 19

after_request() (*riotwatcher.Handlers.DeprecationHandler.DeprecationHandler method*), 24

after_request() (*riotwatcher.Handlers.DeserializerAdapter.DeserializerAdapter method*), 23

after_request() (*riotwatcher.Handlers.RateLimiterAdapter.RateLimiterAdapter method*), 23

after_request() (*riotwatcher.Handlers.RequestHandler.RequestHandler method*), 25

after_request() (*riotwatcher.Handlers.ThrowOnErrorHandler.ThrowOnErrorHandler method*), 24

after_static_request() (*riotwatcher.Handlers.DeserializerAdapter.DeserializerAdapter method*), 24

after_static_request() (*riotwatcher.Handlers.RequestHandler.RequestHandler method*), 25

B

by_account() (*riotwatcher._apis.league_of_legends.SummonerApiV4 method*), 16

by_account() (*riotwatcher._apis.team_fight_tactics.SummonerApi method*), 21

by_id() (*riotwatcher._apis.league_of_legends.LeagueApiV4 method*), 13

by_id() (*riotwatcher._apis.league_of_legends.MatchApiV4 method*), 14

by_id() (*riotwatcher._apis.league_of_legends.SummonerApiV4 method*), 16

by_id() (*riotwatcher._apis.team_fight_tactics.LeagueApi method*), 20

by_id() (*riotwatcher._apis.team_fight_tactics.MatchApi method*), 20

by_id() (*riotwatcher._apis.team_fight_tactics.SummonerApi method*), 21

by_id() (*riotwatcher._apis.valorant.MatchApi method*), 22

by_name() (*riotwatcher._apis.league_of_legends.SummonerApiV4 method*), 16

by_name() (*riotwatcher._apis.team_fight_tactics.SummonerApi method*), 21

by_puuid() (*riotwatcher._apis.league_of_legends.SummonerApiV4 method*), 17

by_puuid() (*riotwatcher._apis.riot.AccountApi method*), 19

by_puuid() (*riotwatcher._apis.team_fight_tactics.MatchApi method*), 21

by_puuid() (*riotwatcher._apis.team_fight_tactics.SummonerApi method*), 21

by_riot_id() (*riotwatcher._apis.riot.AccountApi method*), 19

by_summoner() (*riotwatcher._apis.league_of_legends.ChampionMasteryApiV4 method*), 11

by_summoner() (*riotwatcher._apis.league_of_legends.ClashApiV1 method*), 11

by_summoner() (*riotwatcher._apis.league_of_legends.LeagueApiV4 method*), 13

by_summoner() (*riotwatcher._apis.league_of_legends.SpectatorApiV4 method*), 16

by_summoner() (*riotwatcher._apis.league_of_legends.ThirdPartyCodeApiV4 method*), 17

by_summoner() (*riotwatcher._apis.team_fight_tactics.LeagueApi method*), 20

31