

Package: riotr (via r-universe)

May 28, 2026

Title Interact with the 'Riot Games API'

Version 0.0.0.9000

Description Download and read data from the Riot Games Developer API (<<https://developer.riotgames.com/>>). Provides functions for all primary endpoints, including League of Legends, Teamfight Tactics, Legends of Runeterra, and Valorant.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Imports cli, dplyr, httr2, purrr, tibble

Suggests testthat (>= 3.0.0), vcr (>= 2.0.0)

Depends R (>= 4.1.0)

Config/testthat/edition 3

URL <https://github.com/christopherkenny/riotr>

Config/pak/sysreqs libssl-dev

Repository <https://christopherkenny.r-universe.dev>

Date/Publication 2026-02-27 04:15:58 UTC

RemoteUrl <https://github.com/christopherkenny/riotr>

RemoteRef HEAD

RemoteSha 6fbb226dbd3eb1ed7bad8e22dd50b284b8c5bcf9

Contents

key	3
lol_create_tournament_code	3
lol_get_active_game	4
lol_get_all_challenge_configs	5
lol_get_all_challenge_percentiles	6
lol_get_challenge_config	6

lol_get_challenge_leaderboards	7
lol_get_challenge_percentiles	8
lol_get_challenge_player_data	8
lol_get_challenger_league	9
lol_get_champion_masteries	10
lol_get_champion_mastery	10
lol_get_champion_mastery_score	11
lol_get_champion_rotations	12
lol_get_clash_players	12
lol_get_clash_team	13
lol_get_clash_tournament	14
lol_get_clash_tournament_by_team	14
lol_get_clash_tournaments	15
lol_get_grandmaster_league	16
lol_get_league	16
lol_get_league_entries	17
lol_get_league_entries_by_puuid	18
lol_get_league_exp_entries	18
lol_get_lobby_events	19
lol_get_master_league	20
lol_get_match	20
lol_get_match_ids	21
lol_get_match_replays	22
lol_get_match_timeline	22
lol_get_platform_data	23
lol_get_summoner_by_puuid	24
lol_get_summoner_me	24
lol_get_top_champion_masteries	25
lol_get_tournament_code	26
lol_register_provider	26
lol_register_tournament	27
lor_get_leaderboards	28
lor_get_match	28
lor_get_match_ids	29
lor_get_platform_data	30
riot_get_account_by_puuid	30
riot_get_account_by_puuid_esports	31
riot_get_account_by_riot_id	32
riot_get_account_by_riot_id_esports	32
riot_get_active_region	33
riot_get_active_shard	34
set_riot_key	35
tft_get_challenger_league	35
tft_get_grandmaster_league	36
tft_get_league_entries	37
tft_get_league_entries_by_puuid	37
tft_get_master_league	38
tft_get_match	39

<i>key</i>	3
tft_get_match_ids	39
tft_get_platform_data	40
tft_get_summoner_by_puuid	41
tft_get_summoner_me	41
tft_get_top_rated_ladder	42
val_get_content	43
val_get_leaderboard	43
val_get_platform_data	44
Index	45

<i>key</i>	<i>Check or Get RIOT Key</i>
------------	------------------------------

Description

Check or Get RIOT Key

Usage

has_riot_key()

get_riot_key()

Value

logical if has, key if get

Examples

has_riot_key()

lol_create_tournament_code	<i>Create Tournament Code (Stub)</i>
----------------------------	--------------------------------------

Description

Creates a tournament code for the given tournament.

Usage

lol_create_tournament_code(tournament_id, count = NULL, body, region = "na1")

Arguments

tournament_id	Tournament ID.
count	Number of codes to generate. Defaults to 1.
body	A list with tournament code parameters including mapType, pickType, spectatorType, teamSize, and optionally allowedParticipants and metadata.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".

Value

a character vector of tournament codes

Examples

```
lol_create_tournament_code(
  tournament_id = 1234,
  body = list(
    mapType = 'SUMMONERS_RIFT',
    pickType = 'TOURNAMENT_DRAFT',
    spectatorType = 'ALL',
    teamSize = 5
  )
)
```

lol_get_active_game *Get Active Game by PUUID*

Description

Returns active game information for the given summoner PUUID.

Usage

```
lol_get_active_game(puuid, region = "na1", clean = TRUE)
```

Arguments

puuid	PUUID of the player.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with active game data or NULL if the player is not in a game. Returns a list if `clean = FALSE`.

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEo0H8g1d_K01IQHboIuog80Rd7Q0H2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'  
lol_get_active_game(puuid)
```

lol_get_all_challenge_configs
Get All Challenge Configs

Description

Get All Challenge Configs

Usage

```
lol_get_all_challenge_configs(region = "na1", clean = TRUE)
```

Arguments

region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with all challenge configs or a list if `clean = FALSE`

Examples

```
lol_get_all_challenge_configs()
```

lol_get_all_challenge_percentiles
Get All Challenge Percentiles

Description

Get All Challenge Percentiles

Usage

```
lol_get_all_challenge_percentiles(region = "na1", clean = TRUE)
```

Arguments

region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a list of challenge percentiles

Examples

```
lol_get_all_challenge_percentiles()
```

lol_get_challenge_config
Get Challenge Config by ID

Description

Get Challenge Config by ID

Usage

```
lol_get_challenge_config(challenge_id, region = "na1", clean = TRUE)
```

Arguments

challenge_id	Challenge ID.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with challenge config or a list if `clean = FALSE`

Examples

```
lol_get_challenge_config(1)
```

```
lol_get_challenge_leaderboards  
      Get Challenge Leaderboards
```

Description

Get Challenge Leaderboards

Usage

```
lol_get_challenge_leaderboards(  
  challenge_id,  
  level,  
  region = "na1",  
  clean = TRUE  
)
```

Arguments

<code>challenge_id</code>	Challenge ID.
<code>level</code>	Level. Such as "MASTER", "GRANDMASTER", or "CHALLENGER".
<code>region</code>	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
<code>clean</code>	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with leaderboard entries or a list if `clean = FALSE`

Examples

```
lol_get_challenge_leaderboards(1, 'MASTER')
```

lol_get_challenge_percentiles
Get Challenge Percentiles by ID

Description

Get Challenge Percentiles by ID

Usage

```
lol_get_challenge_percentiles(challenge_id, region = "na1")
```

Arguments

challenge_id	Challenge ID.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".

Value

a list of challenge percentiles

Examples

```
lol_get_challenge_percentiles(1)
```

lol_get_challenge_player_data
Get Challenge Player Data

Description

Get Challenge Player Data

Usage

```
lol_get_challenge_player_data(puuid, region = "na1", clean = TRUE)
```

Arguments

puuid	PUUID of the player.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a list with player challenge data

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEo0H8g1d_K01IQHboIuog80Rd7Q0H2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'  
lol_get_challenge_player_data(puuid)
```

lol_get_challenger_league

Get Challenger League

Description

Get Challenger League

Usage

```
lol_get_challenger_league(queue, region = "na1", clean = TRUE)
```

Arguments

queue	Queue type. Such as "RANKED_SOLO_5x5" or "RANKED_FLEX_SR".
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a [tibble::tibble](#) with league entries or a list if clean = FALSE

Examples

```
lol_get_challenger_league('RANKED_SOLO_5x5')
```

`lol_get_champion_masteries`*Get All Champion Masteries by PUUID*

Description

Get All Champion Masteries by PUUID

Usage

```
lol_get_champion_masteries(puuid, region = "na1", clean = TRUE)
```

Arguments

<code>puuid</code>	PUUID of the player.
<code>region</code>	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
<code>clean</code>	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a [tibble::tibble](#) with champion mastery data or a list if `clean = FALSE`

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEo0H8g1d_K01IQHboIuog80Rd7Q0H2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'  
lol_get_champion_masteries(puuid)
```

`lol_get_champion_mastery`*Get Champion Mastery by PUUID and Champion ID*

Description

Get Champion Mastery by PUUID and Champion ID

Usage

```
lol_get_champion_mastery(puuid, champion_id, region = "na1", clean = TRUE)
```

Arguments

puuid	PUUID of the player.
champion_id	Champion ID.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with champion mastery data or a list if `clean = FALSE`

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEo0H8g1d_K01IQHboIuog80Rd7Q0H2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'
lol_get_champion_mastery(puuid, 1)
```

```
lol_get_champion_mastery_score
```

Get Champion Mastery Score by PUUID

Description

Get Champion Mastery Score by PUUID

Usage

```
lol_get_champion_mastery_score(puuid, region = "na1")
```

Arguments

puuid	PUUID of the player.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".

Value

an integer representing the total champion mastery score

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEo0H8g1d_K01IQHboIuog80Rd7Q0H2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'
lol_get_champion_mastery_score(puuid)
```

 lol_get_champion_rotations

Get Champion Rotation Information

Description

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

Usage

```
lol_get_champion_rotations(region = "na1", clean = TRUE)
```

Arguments

region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with champion rotation data or a list if `clean = FALSE`

Examples

```
lol_get_champion_rotations()
```

 lol_get_clash_players *Get Clash Players by PUUID*

Description

Get Clash Players by PUUID

Usage

```
lol_get_clash_players(puuid, region = "na1", clean = TRUE)
```

Arguments

puuid	PUUID of the player.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with player data or a list if `clean = FALSE`

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEo0H8g1d_K01IQHboIuog80Rd7QOH2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'  
lol_get_clash_players(puuid)
```

lol_get_clash_team	<i>Get Clash Team by ID</i>
--------------------	-----------------------------

Description

Get Clash Team by ID

Usage

```
lol_get_clash_team(team_id, region = "na1", clean = TRUE)
```

Arguments

team_id	Team ID.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with team data, or NULL if not found. Returns a list if `clean = FALSE`.

Examples

```
lol_get_clash_team('PLACEHOLDER_CLASH_TEAM_ID')
```

lol_get_clash_tournament

Get Clash Tournament by ID

Description

Get Clash Tournament by ID

Usage

```
lol_get_clash_tournament(tournament_id, region = "na1", clean = TRUE)
```

Arguments

tournament_id	Tournament ID.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a [tibble::tibble](#) with tournament data, or NULL if not found. Returns a list if clean = FALSE.

Examples

```
lol_get_clash_tournament('PLACEHOLDER_CLASH_TOURNAMENT_ID')
```

lol_get_clash_tournament_by_team

Get Clash Tournament by Team ID

Description

Get Clash Tournament by Team ID

Usage

```
lol_get_clash_tournament_by_team(team_id, region = "na1", clean = TRUE)
```

Arguments

team_id	Team ID.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with tournament data, or NULL if not found. Returns a list if `clean = FALSE`.

Examples

```
lol_get_clash_tournament_by_team('PLACEHOLDER_CLASH_TEAM_ID')
```

```
lol_get_clash_tournaments
```

Get All Clash Tournaments

Description

Get All Clash Tournaments

Usage

```
lol_get_clash_tournaments(region = "na1", clean = TRUE)
```

Arguments

region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with tournament data or a list if `clean = FALSE`

Examples

```
lol_get_clash_tournaments()
```

lol_get_grandmaster_league
Get Grandmaster League

Description

Get Grandmaster League

Usage

```
lol_get_grandmaster_league(queue, region = "na1", clean = TRUE)
```

Arguments

queue	Queue type. Such as "RANKED_SOLO_5x5" or "RANKED_FLEX_SR".
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with league entries or a list if `clean = FALSE`

Examples

```
lol_get_grandmaster_league('RANKED_SOLO_5x5')
```

lol_get_league *Get League by ID*

Description

Get League by ID

Usage

```
lol_get_league(league_id, region = "na1", clean = TRUE)
```

Arguments

league_id	League ID.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with league entries or a list if `clean = FALSE`

Examples

```
lol_get_league('7ccf5091-303b-3f5a-927b-7705b93e3564')
```

lol_get_league_entries

Get League Entries

Description

Get League Entries

Usage

```
lol_get_league_entries(queue, tier, division, region = "na1", clean = TRUE)
```

Arguments

queue	Queue type. Such as "RANKED_SOLO_5x5" or "RANKED_FLEX_SR".
tier	Tier. Such as "DIAMOND", "MASTER", "GRANDMASTER", or "CHALLENGER".
division	Division. Such as "I", "II", "III", or "IV".
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with league entries or a list if `clean = FALSE`

Examples

```
lol_get_league_entries('RANKED_SOLO_5x5', 'DIAMOND', 'I')
```

lol_get_league_entries_by_puuid
Get League Entries by PUUID

Description

Get League Entries by PUUID

Usage

```
lol_get_league_entries_by_puuid(puuid, region = "na1", clean = TRUE)
```

Arguments

puuid	PUUID of the player.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a [tibble::tibble](#) with league entries or a list if clean = FALSE

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEo0H8g1d_K01IQHboIuog80Rd7Q0H2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'  
lol_get_league_entries_by_puuid(puuid)
```

lol_get_league_exp_entries
Get Experimental League Entries

Description

Get Experimental League Entries

Usage

```
lol_get_league_exp_entries(queue, tier, division, region = "na1", clean = TRUE)
```

Arguments

queue	Queue type. Such as "RANKED_SOLO_5x5" or "RANKED_FLEX_SR".
tier	Tier. Such as "DIAMOND", "MASTER", "GRANDMASTER", or "CHALLENGER".
division	Division. Such as "I", "II", "III", or "IV".
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with league entries or a list if `clean = FALSE`

Examples

```
lol_get_league_exp_entries('RANKED_SOLO_5x5', 'DIAMOND', 'I')
```

lol_get_lobby_events *Get Lobby Events by Tournament Code*

Description

Get Lobby Events by Tournament Code

Usage

```
lol_get_lobby_events(tournament_code, region = "na1", clean = TRUE)
```

Arguments

tournament_code	Tournament code.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with lobby events or a list if `clean = FALSE`

Examples

```
lol_get_lobby_events('tournament-code')
```

lol_get_master_league *Get Master League*

Description

Get Master League

Usage

```
lol_get_master_league(queue, region = "na1", clean = TRUE)
```

Arguments

queue	Queue type. Such as "RANKED_SOLO_5x5" or "RANKED_FLEX_SR".
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a [tibble::tibble](#) with league entries or a list if clean = FALSE

Examples

```
lol_get_master_league('RANKED_SOLO_5x5')
```

lol_get_match *Get Match by ID*

Description

Get Match by ID

Usage

```
lol_get_match(match_id, regional = "americas", clean = TRUE)
```

Arguments

match_id	Match ID.
regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with match data or a list if `clean = FALSE`

Examples

```
lol_get_match('NA1_5494453720')
```

lol_get_match_ids	<i>Get Match IDs by PUUID</i>
-------------------	-------------------------------

Description

Get Match IDs by PUUID

Usage

```
lol_get_match_ids(
  puuid,
  start = NULL,
  count = NULL,
  queue = NULL,
  type = NULL,
  start_time = NULL,
  end_time = NULL,
  regional = "americas"
)
```

Arguments

puuid	PUUID of the player.
start	Start index. Defaults to 0.
count	Number of match IDs to return. Defaults to 20.
queue	Queue ID to filter by.
type	Type of match to filter by. Such as "ranked", "normal", "tourney", or "tutorial".
start_time	Epoch timestamp in seconds to filter matches after.
end_time	Epoch timestamp in seconds to filter matches before.
regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".

Value

a character vector of match IDs

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEo0H8g1d_K01IQHboIuog80Rd7Q0H2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'  
lol_get_match_ids(puuid)
```

```
lol_get_match_replays Get Match Replays by PUUID
```

Description

Get Match Replays by PUUID

Usage

```
lol_get_match_replays(puuid, regional = "americas")
```

Arguments

puuid	PUUID of the player.
regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".

Value

a character vector of replay URLs

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEo0H8g1d_K01IQHboIuog80Rd7Q0H2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'  
lol_get_match_replays(puuid)
```

```
lol_get_match_timeline  
Get Match Timeline
```

Description

Get Match Timeline

Usage

```
lol_get_match_timeline(match_id, regional = "americas", clean = TRUE)
```

Arguments

match_id	Match ID.
regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a list with timeline data (timelines are complex nested structures)

Examples

```
lol_get_match_timeline('NA1_5494453720')
```

lol_get_platform_data *Get LoL Platform Data (Status)*

Description

Get LoL Platform Data (Status)

Usage

```
lol_get_platform_data(region = "na1", clean = TRUE)
```

Arguments

region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a [tibble::tibble](#) with platform status data or a list if clean = FALSE

Examples

```
lol_get_platform_data()
```

 lol_get_summoner_by_puuid

Get Summoner by PUUID

Description

Get Summoner by PUUID

Usage

```
lol_get_summoner_by_puuid(puuid, region = "na1", clean = TRUE)
```

Arguments

puuid	PUUID of the player.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a [tibble::tibble](#) with summoner data or a list if clean = FALSE

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEoOH8g1d_K01IQHboIuog80Rd7QOH2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'
lol_get_summoner_by_puuid(puuid)
```

 lol_get_summoner_me

Get Summoner by Access Token

Description

Returns the summoner associated with the current access token.

Usage

```
lol_get_summoner_me(region = "na1", clean = TRUE)
```

Arguments

region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with summoner data, or NULL if the request fails. Returns a list if `clean = FALSE`.

Examples

```
lol_get_summoner_me()
```

```
lol_get_top_champion_masteryes
      Get Top Champion Masteries by PUUID
```

Description

Get Top Champion Masteries by PUUID

Usage

```
lol_get_top_champion_masteryes(
  puuid,
  count = NULL,
  region = "na1",
  clean = TRUE
)
```

Arguments

<code>puuid</code>	PUUID of the player.
<code>count</code>	Number of entries to return. Defaults to 3.
<code>region</code>	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
<code>clean</code>	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with top champion mastery data or a list if `clean = FALSE`

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEo0H8g1d_K01IQHboIuog80Rd7Q0H2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'
lol_get_top_champion_masteryes(puuid)
```

 lol_get_tournament_code

Get Tournament Code

Description

Get Tournament Code

Usage

```
lol_get_tournament_code(tournament_code, region = "na1", clean = TRUE)
```

Arguments

tournament_code	
region	Tournament code.
clean	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with tournament code data or a list if `clean = FALSE`

Examples

```
lol_get_tournament_code('tournament-code')
```

 lol_register_provider *Register Tournament Provider (Stub)*

Description

Registers a tournament provider.

Usage

```
lol_register_provider(body, region = "na1")
```

Arguments

body	A list with region and url fields.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".

Value

an integer provider ID

Examples

```
lol_register_provider(  
  body = list(region = 'NA', url = 'https://example.com')  
)
```

lol_register_tournament

Register Tournament (Stub)

Description

Registers a tournament.

Usage

```
lol_register_tournament(body, region = "na1")
```

Arguments

body	A list with name and providerId fields.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".

Value

an integer tournament ID

Examples

```
lol_register_tournament(  
  body = list(name = 'My Tournament', providerId = 1)  
)
```

lor_get_leaderboards *Get LoR Ranked Leaderboards*

Description

Get LoR Ranked Leaderboards

Usage

```
lor_get_leaderboards(regional = "americas", clean = TRUE)
```

Arguments

regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a [tibble::tibble](#) with ranked leaderboard entries or a list if clean = FALSE

Examples

```
lor_get_leaderboards()
```

lor_get_match *Get LoR Match by ID*

Description

Get LoR Match by ID

Usage

```
lor_get_match(match_id, regional = "americas", clean = TRUE)
```

Arguments

match_id	Match ID.
regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with match data, or NULL if not found. Returns a list if `clean = FALSE`.

Examples

```
lor_get_match('PLACEHOLDER_LOR_MATCH_ID')
```

lor_get_match_ids	<i>Get LoR Match IDs by PUUID</i>
-------------------	-----------------------------------

Description

Get LoR Match IDs by PUUID

Usage

```
lor_get_match_ids(puuid, regional = "americas")
```

Arguments

puuid	PUUID of the player.
regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".

Value

a character vector of match IDs

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEo0H8g1d_K01IQHboIuog80Rd7Q0H2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'  
lor_get_match_ids(puuid)
```

lor_get_platform_data *Get LoR Platform Data (Status)*

Description

Get LoR Platform Data (Status)

Usage

```
lor_get_platform_data(regional = "americas", clean = TRUE)
```

Arguments

regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a [tibble::tibble](#) with platform status data or a list if clean = FALSE

Examples

```
lor_get_platform_data()
```

riot_get_account_by_puuid
Get Account by PUUID

Description

Get Account by PUUID

Usage

```
riot_get_account_by_puuid(puuid, regional = "americas", clean = TRUE)
```

Arguments

puuid	PUUID of the player.
regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with account data or a list if `clean = FALSE`

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEo0H8g1d_K01IQHboIuog80Rd7Q0H2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'  
riot_get_account_by_puuid(puuid)
```

```
riot_get_account_by_puuid_esports  
  Get Account by PUUID (Esports)
```

Description

Get Account by PUUID (Esports)

Usage

```
riot_get_account_by_puuid_esports(puuid, regional = "esports", clean = TRUE)
```

Arguments

<code>puuid</code>	PUUID of the player.
<code>regional</code>	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".
<code>clean</code>	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with account data, or NULL if not found. Returns a list if `clean = FALSE`.

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEo0H8g1d_K01IQHboIuog80Rd7Q0H2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'  
riot_get_account_by_puuid_esports(puuid)
```

riot_get_account_by_riot_id
Get Account by Riot ID

Description

Get Account by Riot ID

Usage

```
riot_get_account_by_riot_id(  
  game_name,  
  tag_line,  
  regional = "americas",  
  clean = TRUE  
)
```

Arguments

game_name	Game name of the Riot account.
tag_line	Tag line of the Riot account.
regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a [tibble::tibble](#) with account data or a list if `clean = FALSE`

Examples

```
riot_get_account_by_riot_id('Doublelift', 'NA1')
```

riot_get_account_by_riot_id_esports
Get Account by Riot ID (Esports)

Description

Get Account by Riot ID (Esports)

Usage

```
riot_get_account_by_riot_id_esports(
  game_name,
  tag_line,
  regional = "esports",
  clean = TRUE
)
```

Arguments

game_name	Game name of the Riot account.
tag_line	Tag line of the Riot account.
regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a [tibble::tibble](#) with account data, or NULL if not found. Returns a list if clean = FALSE.

Examples

```
riot_get_account_by_riot_id_esports('Doublelift', 'NA1')
```

```
riot_get_active_region
```

Get Active Region for a Player

Description

Get Active Region for a Player

Usage

```
riot_get_active_region(game, puuid, regional = "americas", clean = TRUE)
```

Arguments

game	Game identifier. Such as "val" or "lor".
puuid	PUUID of the player.
regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with active region data, or NULL if the request fails. Returns a list if `clean = FALSE`.

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEo0H8g1d_K01IQHboIuog80Rd7Q0H2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'  
riot_get_active_region('val', puuid)
```

riot_get_active_shard *Get Active Shard for a Player*

Description

Get Active Shard for a Player

Usage

```
riot_get_active_shard(game, puuid, regional = "americas", clean = TRUE)
```

Arguments

game	Game identifier. Such as "val" or "lor".
puuid	PUUID of the player.
regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with active shard data or a list if `clean = FALSE`

Examples

```
puuid <- '02q8aNv6VB1Fz9ABjEo0H8g1d_K01IQHboIuog80Rd7Q0H2L8nGgsYgK-TN16ucwUQsPpihH-tLtAA'  
riot_get_active_shard('val', puuid)
```

set_riot_key	<i>Add Entry to Renviron</i>
--------------	------------------------------

Description

Adds RIOT API key to .Renviron.

Usage

```
set_riot_key(key, overwrite = FALSE, install = FALSE, r_env = NULL)
```

Arguments

key	Character. API key to add to add.
overwrite	Defaults to FALSE. Boolean. Should existing RIOT_KEY in Renviron be overwritten?
install	Defaults to FALSE. Boolean. Should this be added '~/.Renviron' file?
r_env	Path to install to if install is TRUE.

Value

key, invisibly

Examples

```
example_env <- tempfile(fileext = '.Renviron')
set_riot_key('1234', r_env = example_env)
# r_env should likely be: file.path(Sys.getenv('HOME'), '.Renviron')
```

tft_get_challenger_league	<i>Get TFT Challenger League</i>
---------------------------	----------------------------------

Description

Get TFT Challenger League

Usage

```
tft_get_challenger_league(queue = "RANKED_TFT", region = "na1", clean = TRUE)
```

Arguments

queue	Queue type. Defaults to "RANKED_TFT".
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with league entries or a list if `clean = FALSE`

Examples

```
tft_get_challenger_league()
```

```
tft_get_grandmaster_league
      Get TFT Grandmaster League
```

Description

Get TFT Grandmaster League

Usage

```
tft_get_grandmaster_league(queue = "RANKED_TFT", region = "na1", clean = TRUE)
```

Arguments

queue	Queue type. Defaults to "RANKED_TFT".
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with league entries or a list if `clean = FALSE`

Examples

```
tft_get_grandmaster_league()
```

`tft_get_league_entries`*Get TFT League Entries*

Description

Get TFT League Entries

Usage

```
tft_get_league_entries(tier, division, region = "na1", clean = TRUE)
```

Arguments

<code>tier</code>	Tier. Such as "DIAMOND", "MASTER", "GRANDMASTER", or "CHALLENGER".
<code>division</code>	Division. Such as "I", "II", "III", or "IV".
<code>region</code>	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
<code>clean</code>	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a [tibble::tibble](#) with league entries or a list if `clean = FALSE`

Examples

```
tft_get_league_entries('DIAMOND', 'I')
```

`tft_get_league_entries_by_puuid`*Get TFT League Entries by PUUID*

Description

Get TFT League Entries by PUUID

Usage

```
tft_get_league_entries_by_puuid(puuid, region = "na1", clean = TRUE)
```

Arguments

puuid	PUUID of the player.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with league entries or a list if `clean = FALSE`

Examples

```
puuid <- 'GmNPNjUbzbWN8YIA79V4p308eFpf21xPO2Lw8o2xJTJMPW83W54t4GfyfpwOn7-hBz5c3MHpg-Dc3w'
tft_get_league_entries_by_puuid(puuid)
```

tft_get_master_league *Get TFT Master League*

Description

Get TFT Master League

Usage

```
tft_get_master_league(queue = "RANKED_TFT", region = "na1", clean = TRUE)
```

Arguments

queue	Queue type. Defaults to "RANKED_TFT".
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with league entries or a list if `clean = FALSE`

Examples

```
tft_get_master_league()
```

tft_get_match	<i>Get TFT Match by ID</i>
---------------	----------------------------

Description

Get TFT Match by ID

Usage

```
tft_get_match(match_id, regional = "americas", clean = TRUE)
```

Arguments

match_id	Match ID.
regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with match data or a list if `clean = FALSE`

Examples

```
tft_get_match('NA1_5501463494')
```

tft_get_match_ids	<i>Get TFT Match IDs by PUUID</i>
-------------------	-----------------------------------

Description

Get TFT Match IDs by PUUID

Usage

```
tft_get_match_ids(  
  puuid,  
  start = NULL,  
  count = NULL,  
  start_time = NULL,  
  end_time = NULL,  
  regional = "americas"  
)
```

Arguments

puuid	PUUID of the player.
start	Start index. Defaults to 0.
count	Number of match IDs to return. Defaults to 20.
start_time	Epoch timestamp in seconds to filter matches after.
end_time	Epoch timestamp in seconds to filter matches before.
regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".

Value

a character vector of match IDs

Examples

```
puuid <- 'GmNPNjUbzbWN8YIA79V4p308eFpf21xP02Lw8o2xJTJMPW83W54t4GfyfpwOn7-hBz5c3MHpg-Dc3w'
tft_get_match_ids(puuid)
```

tft_get_platform_data *Get TFT Platform Data (Status)*

Description

Get TFT Platform Data (Status)

Usage

```
tft_get_platform_data(region = "na1", clean = TRUE)
```

Arguments

region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with platform status data or a list if `clean = FALSE`

Examples

```
tft_get_platform_data()
```

```
tft_get_summoner_by_puuid
```

Get TFT Summoner by PUUID

Description

Get TFT Summoner by PUUID

Usage

```
tft_get_summoner_by_puuid(puuid, region = "na1", clean = TRUE)
```

Arguments

puuid	PUUID of the player.
region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a [tibble::tibble](#) with summoner data or a list if `clean = FALSE`

Examples

```
puuid <- 'GmNPNjUbzbWN8YIA79V4p308eFpf21xP02Lw8o2xJTJMPW83W54t4GfyfPwOn7-hBz5c3MHpg-Dc3w'
tft_get_summoner_by_puuid(puuid)
```

```
tft_get_summoner_me
```

Get TFT Summoner by Access Token

Description

Returns the TFT summoner associated with the current access token.

Usage

```
tft_get_summoner_me(region = "na1", clean = TRUE)
```

Arguments

region	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with summoner data, or NULL if the request fails. Returns a list if `clean = FALSE`.

Examples

```
tft_get_summoner_me()
```

```
tft_get_top_rated_ladder
```

Get TFT Top Rated Ladder

Description

Get TFT Top Rated Ladder

Usage

```
tft_get_top_rated_ladder(queue, region = "na1", clean = TRUE)
```

Arguments

<code>queue</code>	Queue type for TFT rated ladder.
<code>region</code>	Region routing value. Platform regions include "na1", "euw1", "eun1", "kr", "br1", "jp1", "la1", "la2", "oc1", "tr1", "ru", "ph2", "sg2", "th2", "tw2", "vn2".
<code>clean</code>	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with top rated ladder entries or a list if `clean = FALSE`

Examples

```
tft_get_top_rated_ladder('RANKED_TFT_TURBO')
```

val_get_content	<i>Get Valorant Content</i>
-----------------	-----------------------------

Description

Get Valorant Content

Usage

```
val_get_content(locale = NULL, regional = "na", clean = TRUE)
```

Arguments

locale	Locale. Such as "en-US", "ja-JP", etc.
regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a list with Valorant content data

Examples

```
val_get_content()
```

val_get_leaderboard	<i>Get Valorant Ranked Leaderboard by Act</i>
---------------------	---

Description

Get Valorant Ranked Leaderboard by Act

Usage

```
val_get_leaderboard(  
  act_id,  
  size = NULL,  
  start_index = NULL,  
  regional = "na",  
  clean = TRUE  
)
```

Arguments

act_id	Act ID.
size	Number of entries to return.
start_index	Start index for pagination.
regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with ranked leaderboard entries or a list if `clean = FALSE`

Examples

```
val_get_leaderboard('3ea2b318-423b-cf86-25da-7cbb0eefbe2d')
```

`val_get_platform_data` *Get Valorant Platform Data (Status)*

Description

Get Valorant Platform Data (Status)

Usage

```
val_get_platform_data(regional = "na", clean = TRUE)
```

Arguments

regional	Regional routing value. One of "americas", "asia", "europe", "sea", or "esports".
clean	Default is TRUE. Should output be returned as a tibble (TRUE) or list.

Value

a `tibble::tibble` with platform status data or a list if `clean = FALSE`

Examples

```
val_get_platform_data()
```

Index

get_riot_key(key), 3

has_riot_key(key), 3

key, 3

lol_create_tournament_code, 3

lol_get_active_game, 4

lol_get_all_challenge_configs, 5

lol_get_all_challenge_percentiles, 6

lol_get_challenge_config, 6

lol_get_challenge_leaderboards, 7

lol_get_challenge_percentiles, 8

lol_get_challenge_player_data, 8

lol_get_challenger_league, 9

lol_get_champion_masteries, 10

lol_get_champion_mastery, 10

lol_get_champion_mastery_score, 11

lol_get_champion_rotations, 12

lol_get_clash_players, 12

lol_get_clash_team, 13

lol_get_clash_tournament, 14

lol_get_clash_tournament_by_team, 14

lol_get_clash_tournaments, 15

lol_get_grandmaster_league, 16

lol_get_league, 16

lol_get_league_entries, 17

lol_get_league_entries_by_puuid, 18

lol_get_league_exp_entries, 18

lol_get_lobby_events, 19

lol_get_master_league, 20

lol_get_match, 20

lol_get_match_ids, 21

lol_get_match_replays, 22

lol_get_match_timeline, 22

lol_get_platform_data, 23

lol_get_summoner_by_puuid, 24

lol_get_summoner_me, 24

lol_get_top_champion_masteries, 25

lol_get_tournament_code, 26

lol_register_provider, 26

lol_register_tournament, 27

lor_get_leaderboards, 28

lor_get_match, 28

lor_get_match_ids, 29

lor_get_platform_data, 30

riot_get_account_by_puuid, 30

riot_get_account_by_puuid_esports, 31

riot_get_account_by_riot_id, 32

riot_get_account_by_riot_id_esports, 32

riot_get_active_region, 33

riot_get_active_shard, 34

set_riot_key, 35

tft_get_challenger_league, 35

tft_get_grandmaster_league, 36

tft_get_league_entries, 37

tft_get_league_entries_by_puuid, 37

tft_get_master_league, 38

tft_get_match, 39

tft_get_match_ids, 39

tft_get_platform_data, 40

tft_get_summoner_by_puuid, 41

tft_get_summoner_me, 41

tft_get_top_rated_ladder, 42

tibble::tibble, 5, 7, 9–21, 23–26, 28–34, 36–42, 44

val_get_content, 43

val_get_leaderboard, 43

val_get_platform_data, 44