

MY API Reference

This API reference describes resources created in a **Spring Boot** Web services application running locally. It is a REST API with simple resources to use as the basis to create sample API Reference Documentation.

Note: The environment and code, and the actual reference documentation are works in progress.

About the API

- uses resource URIs and standard HTTP requests
- the resources are simple operations using ideas found on the web
- is a Java Spring Boot project on the backend (I created the server and wrote the Java code)
- uses working Java methods
- Postman and cURL used to test the requests and confirm the expected responses

About the Documentation

The documentation design is based on various public API References. While reviewing the public API references, the primary aspects considered were content and format.

Contents

- Authentication
- Resources
 - Needles
 - Numbers
- Responses

Authentication

This service uses **Spring Web Security** API keys and secrets.

Base URL

http://localhost:8080/api/

Resources

Needles

Send up to five different words (needles) and a single text block (haystack). Find the needles are in the haystack.

POST /needles

Request		
Parameter	Type	Example
haystack	String <i>required</i>	"haystack": "Like finding a needle in a haystack"
needles	String[] <i>required</i>	"needles": ["finding", "needle", "a", "haystack"]

Response				
	200	400	404	500
Field	Description		Example	
needle	a word in the set		"needle": "finding"	
total	match count		"total": 2	

If the request contains more than five needles, the response will return the following:

"The maximum number of needles allowed is 5."

```
curl -L 'http://localhost:8080/api/needles' \
-H 'Content-Type: application/json' \
-X-API-KEY: '{key}'
-d '{"haystack": "Like finding a needle in a
haystack", "needles": ["finding", "needle", "a", "haystack"
, "needles"]}'
```

```
200 OK

{"needles": [
  {"count": "1", "needle": "finding"},
  {"count": "1", "needle": "needle"},
  {"count": "2", "needle": "a"},
  {"count": "1", "needle": "haystack"},
  {"count": "0", "needle": "needles"}
]}

200 OK

"The maximum number of needles allowed is 5."
```

Numbers

Send a lottery game type and receive a set of random numbers for that game.

GET /numbers

Request

Note: The *game* parameter can only have one of three values: *pick4*, *pick5*, *lotto*

Parameter	Type	Example
game	String <i>required</i>	game=(pick4 pick5 lotto)

Response

	200	400	404	500
Field	Description			Example
pick4	the set of four random numbers			“pick4”: [3,21,5,15]
pick5	the set of five random numbers			“pick5”: [17,5,22,40,46]
lotto	the set of six random numbers			“lotto”: [17,3,19,20,4,43]

If the request contains an invalid game, the response will return the following:

<<game>> is not a valid game type. Value must be pick4, pick5, or lotto.

```
curl -L
'http://localhost:8080/api/numbers?api_key={}&game={}'
```

```
200 OK

{"pick4": [3,21,5,15]}

{"pick5": [17,5,22,40,46]}

{"lotto": [17,3,19,20,4,43]}
```

```
200 OK

<<game>> is not a valid game type. Value must be pick4,
pick5, or lotto.
```

Responses

200

Standard response for a successful HTTP request. The actual response will depend on the actual request method used. Refer to the applicable resource section for specific 200 response message.

400

The request cannot be filled due to bad syntax.

Field	Example
timestamp	YYYY-MM-DDT00:00:00.000+00:00
status	400
error	Bad Request
path	/api/example

```
400 Bad Request

{
  "timestamp": "2024-08-15T00:03:04.554+00:00",
  "status": 400,
  "error": "Bad Request",
  "path": "/api/numbers"
}
```

404

The requested resource could not be found but may be available again in the future. Subsequent requests by the client are permissible.

Field	Example
timestamp	YYYY-MM-DDT00:00:00.000+00:00
status	404
error	Not Found
path	/

```
404 Not Found

{
  "timestamp": "2024-08-16T12:24:11.041+00:00",
  "status": 404,
  "error": "Not Found",
  "message": "No static resource .",
  "path": "/"
}
```

500

A generic error message given when no more specific message is suitable.

For authentication error, the response will return the following:

“Missing API Key or Secret”

Field	Example
timestamp	YYYY-MM-DDT00:00:00.000+00:00
status	500
error	Internal Server Error
path	/api/example

```
500 Internal Server Error

{
  "timestamp": "2024-08-16T13:18:45.511+00:00",
  "status": 500,
  "error": "Internal Server Error",
  "message": "Missing API Key or Secret",
  "path": "/api/data"
}
```