Brewery Beer Service Docs

Alassane NIANG

Table of Contents

Sample Application demonstrating how to use Spring REST Docs with JUnit 5.

`BeerControllerTest` makes call to very simple service and produces three documentation snippets.

Get by ID

One showing how to make a request using cURL

```
$ curl 'http://localhost:8080/api/v1/beer/4b4433ba-cdb8-4ab7-97f7-
cc28ab0a611a?isCold=yes' -i -X GET \
    -H 'Accept: application/json'
```

One showing how to make a request using HTTP

```
GET /api/v1/beer/4b4433ba-cdb8-4ab7-97f7-cc28ab0a611a?isCold=yes HTTP/1.1
Accept: application/json
Host: localhost:8080
```

One showing how to make a request using HTTPIE

```
$ http GET 'http://localhost:8080/api/v1/beer/4b4433ba-cdb8-4ab7-97f7-
cc28ab0a611a?isCold=yes' \
    'Accept:application/json'
```

Path Parameters

Table 1. /api/v1/beer/{beerId}

Parameter	Description
beerId	UUID of desired beer to get

Request Parameters

Parameter	Description
isCold	Is beer cold query parameter

One showing the HTTP Response

```
HTTP/1.1 200 OK
Content-Type: application/json
Content-Length: 148

{"id":null,"version":null,"createdDate":null,"lastModifiedDate":null,"beerName":null,"
beerStyle":null,"upc":null,"price":null,"quantityOnHand":null}
```

Request Body

Response Fields

Path	Type	Description
id	UUID	Id of the beer
version	Long	Version number
createdDate	Timestamp	Date created
lastModifiedDate	Timestamp	Date updated
beerName	String	Beer name
beerStyle	String	Beer style
upc	Long	UPC of beer
price	BigDecimal	Beer price
quantityOnHand	Integer	Quantity on hand

Response Body

```
{"id":null,"version":null,"createdDate":null,"lastModifiedDate":null,"beerName":null,"beerStyle":null,"upc":null,"price":null,"quantityOnHand":null}
```

Path	Type	Description
id	UUID	Id of the beer
version	Long	Version number

Create new beer

One showing how to make a request using cURL

```
$ curl 'http://localhost:8080/api/v1/beer/' -i -X POST \
    -H 'Content-Type: application/json' \
    -d
'{"id":null,"version":null,"createdDate":null,"lastModifiedDate":null,"beerName":"My
Beer","beerStyle":"ALE","upc":123123123123,"price":2.99,"quantityOnHand":null}'
```

One showing how to make a request using HTTPIE

```
$ echo
'{"id":null,"version":null,"createdDate":null,"lastModifiedDate":null,"beerName":"My
Beer","beerStyle":"ALE","upc":123123123123,"price":2.99,"quantityOnHand":null}' | http
POST 'http://localhost:8080/api/v1/beer/' \
    'Content-Type:application/json'
```

One showing how to make a request using HTTP

```
POST /api/v1/beer/ HTTP/1.1
Content-Type: application/json
Content-Length: 162
Host: localhost:8080

{"id":null,"version":null,"createdDate":null,"lastModifiedDate":null,"beerName":"My
Beer","beerStyle":"ALE","upc":123123123123,"price":2.99,"quantityOnHand":null}
```

Request Fields

Path	Type	Description	Constraints
beerName	String	Beer name	Must not be blank. Size must be between 3 and 100 inclusive
beerStyle	String	Beer style	Must not be null
upc	Number	UPC of beer	Must be positive. Must not be null
price	Number	Beer price	Must be positive. Must not be null

Request Body

{"id":null,"version":null,"createdDate":null,"lastModifiedDate":null,"beerName":"My Beer","beerStyle":"ALE","upc":123123123123,"price":2.99,"quantityOnHand":null}

One showing the HTTP Response

```
HTTP/1.1 201 Created
```

Response Body