

# GET Requests: Validating Headers

---



**Andrejs Doronins**

# Overview



**Typical headers overview**

**Use `.peek()` and `.print()`**

**Assert and validate various headers**

**Add syntactic sugar methods**

**Leverage the Hamcrest library**

**Maps of headers**

# Common and Custom Headers

**status:** 200 OK

**content-type:** application/json; charset=utf-8

**date:** Mon, 05 Apr 2021 16:11:35 GMT

**etag:** "8ccd6372bcf27b234e53e90d8e0ae9c12efd55c887b556b"

**server:** GitHub.com

**x-frame-options:** deny

**x-ratelimit-limit:** 60

[...]

**“X” prefix is used for custom headers**

# Header Groups

**1**

**Standard and  
ubiquitous**  
e.g. “content-type”

**2**

**Standard**  
e.g. “date” or “etag”

**3**

**Custom**  
prefixed with “x-”

# Debugging Methods

`.peek()`

`.prettyPeek()`

`.print()`

`.prettyPrint()`

# Getting the Headers

```
.getStatusCode()  
.getContentType()  
.getHeader(String header)  
.getHeaders()
```

Conversions from finer to coarser granularities truncate, so lose precision.

For example, converting 999 milliseconds to seconds results in 0.



Agree on a convention, and  
then code consistently



## Response

```
assertEquals  
(r.getStatusCode(), 200);
```

## ValidatableResponse

```
.assertThat()  
.statusCode(200)
```

# Further Checks



**Numbers and dates**

**String contains X**

**Collection contains Y**

**Is empty or is null**

`.assertTrue()`

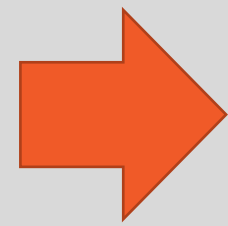
`.assertFalse()`

`.assertNotNull()`

```
.assertTrue(r.getThis() > 199)  
.assertFalse(r.getThat().contains("x"))  
.assertNotNull(r.getOther())
```

```
.statusCode(int code)
```

```
.header("x", "y")
```



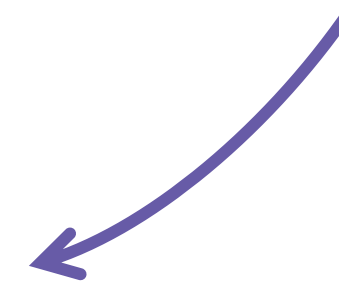
```
.statusCode(int code, Matcher m)
```

```
.header("x", "y", Matcher m)
```

**REST Assured**

**Hamcrest**

Batteries included



```
import static org.hamcrest.MatcherAssert.assertThat;
```

```
import static org.hamcrest.Matchers.*;
```

```
@Test
```

```
public void hamcrestExample() {
```

```
    assertThat(10, lessThan(11));
```

```
}
```

```
import static org.hamcrest.Matchers.*;
```

```
RestAssured.get(url)
```

```
    .then()
```

```
    .statusCode(200)
```

```
    .statusCode(equalTo(200))
```

```
    .statusCode(lessThan(300))
```

```
    .header("x", containsStringIgnoringCase("y"))
```



```
import static org.hamcrest.Matchers.*;
```

```
RestAssured.get(url)  
    .then()  
    .header("etag", notNullValue())  
    .header("etag", not(emptyString()))  
    .statusCode(anyOf(equalTo(200), equalTo(202)))
```

```
// simple
```

```
.header(String header, String v)
```

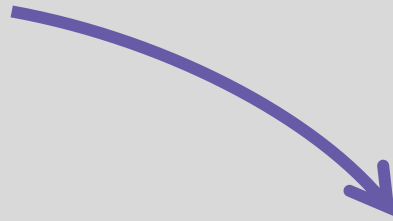
```
.header(String header, Matcher m)
```

```
// advanced
```

```
.header(String header, Function f, Matcher m)
```

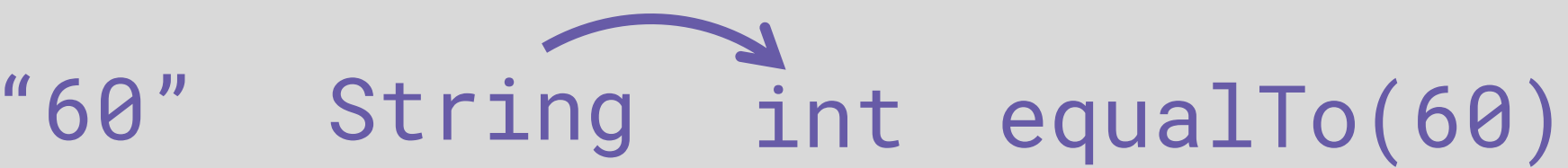
```
.header(String header, ResponseAwareMatcher m)
```

"60", "200"



```
.header(String header, String v)
```

`"60" String int equalTo(60)`  
`.header(String header, Function f, Matcher m)`

A blue curved arrow points from the word 'String' in the first line of code to the word 'int' in the same line. Another blue curved arrow points from the expression 'equalTo(60)' to the word 'int'.

`lambda`

A red curved arrow points from the word 'lambda' to the parameter 'f' in the 'Function f' part of the code above.

`.header("rate-limit", Integer::parseInt, lessThan(100))`

# Demo



## Converting Header Values with Functions

```
/**
```

```
 * [...] allows comparing a header in the response to
```

```
 * something else available in the response
```

```
 */
```

```
.header(String header, ResponseAwareMatcher m)
```

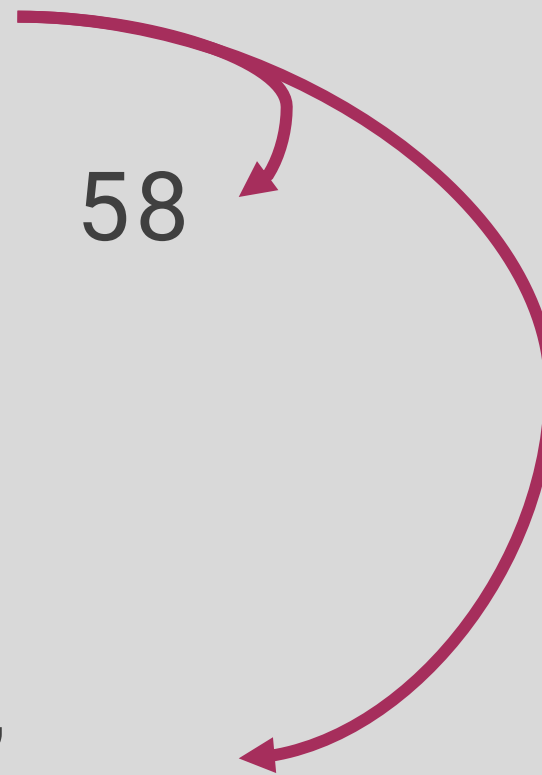
// Headers

**x-ratelimit-limit: 60**

**x-ratelimit-remaining: 58**

//Body

**SomeField: "someValue"**



```
/**
```

```
 * [...] allows comparing a header in the response to
```

```
 * something else available in the response
```

```
 */
```

```
.header(String header, ResponseAwareMatcher m)
```



```
.headers(String h1, Object v1, Object... headers)
```

```
.headers(Map<String, ?> headers)
```

# Summary



**Simple Response**

**ValidatableResponse**

- **Use then()**

**Convenience methods**

- **StatusCode()**
- **ContentType()**

**header()**

**headers()**

Up Next:  
Validating the Body

---