

# Leveraging REST Assured Configuration



**Andrejs Doronins**

Software Developer in Test

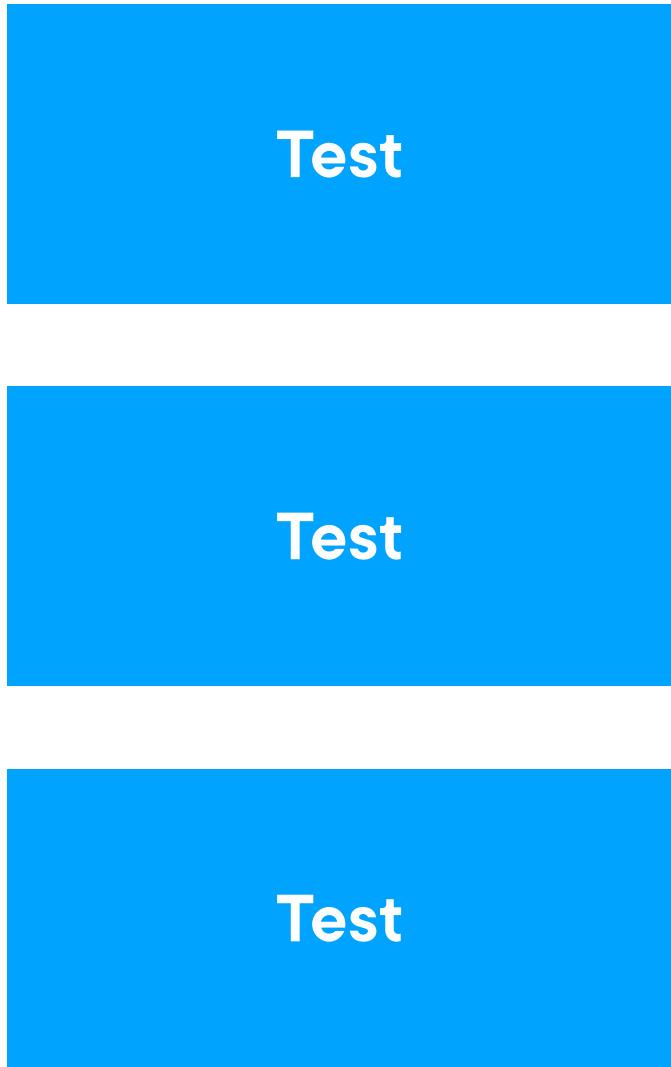


**Test  
(duplicate code)**

**Test  
(duplicate code)**

**Test  
(duplicate code)**





# Overview



**Global REST Assured variables**

**Global config**

**Request specification**

**Response specification**

**Logging**



```
@Test
```

```
void testOne() {
```

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

```
    // ...
```

```
}
```

```
@Test
```

```
void testTwo() {
```

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

```
    // ...
```

```
}
```

RestAssured.baseUri

.get()



```
public class RestAssured {  
    String baseURI  
    int port  
    String basePath  
    String rootPath  
    //...  
    RestAssuredConfig config  
    RequestSpecification requestSpecification  
    ResponseSpecification responseSpecification  
}
```



**basePath**

`http://localhost:8080/`

**baseURI**

**port**



java.net.ConnectException: Connection refused: connect



# RestAssuredConfig



**Underlying HttpClient behavior**

**Redirecting**

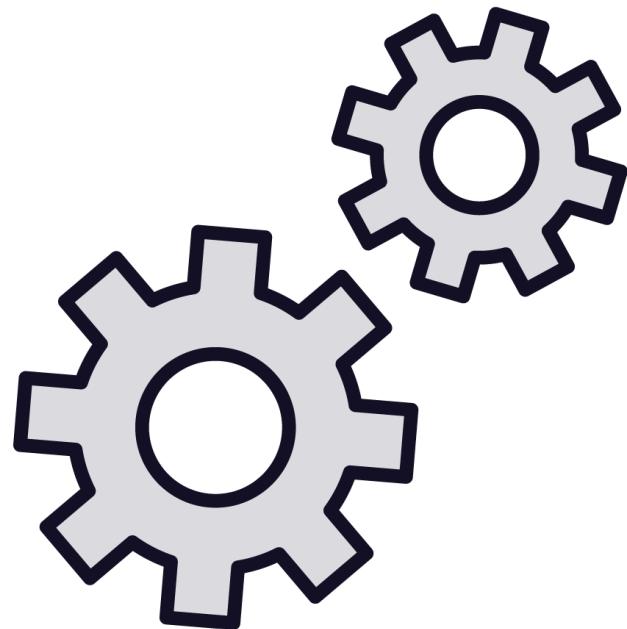
**Connection timeout**

**Logging**

**Encoding and decoding**

**Behavior in case of test failure**





### **RequestSpecification:**

- What to send on all or some requests

### **ResponseSpecification:**

- What to verify on all or some responses

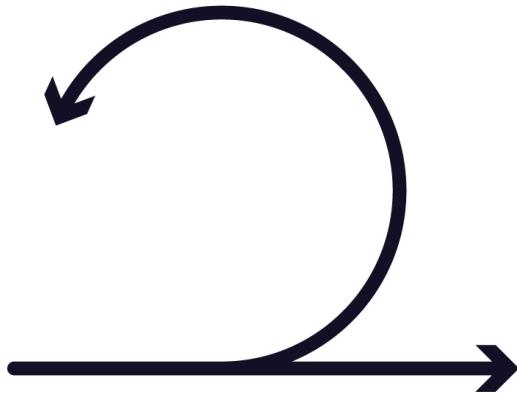


# **URL redirection**

**a.k.a. URL forwarding - a technique to give more than one URL address to a resource**



# URL Forwarding



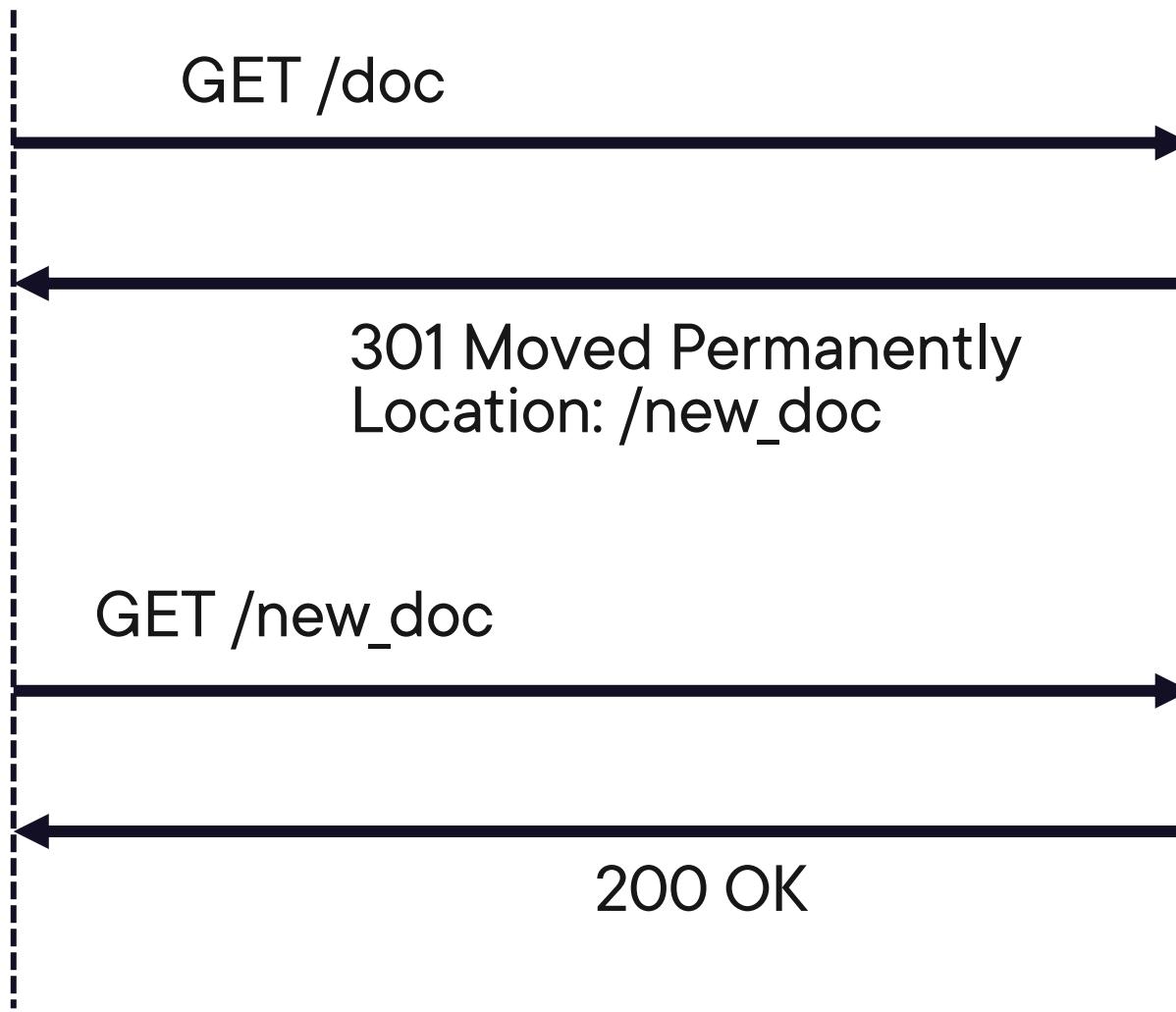
**Temporarily redirect during site maintenance or downtime**

**In case of permanent redirects – to preserve existing links after changing the site's URLs**



Client

Server



.spec()

```
new RequestSpecBuilder()  
    .setThingOne( ... )  
    .setThingTwo( ... )  
    .build();
```

RestAssured.requestSpecification



```
graph TD; A[new ResponseSpecBuilder<br/>.expectStatusCode(200)<br/>.expectContentType(JSON)<br/>.build();] --> B[RestAssured.responseSpecification]; A --> C[.then();<br/>.spec();]
```

.then()  
.spec()

```
new ResponseSpecBuilder()  
.expectStatusCode(200)  
.expectContentType(JSON)  
.build();
```

RestAssured.responseSpecification



```
RestAssured
    .given()
        .spec(commonRequestSpec)
    .when()
        .get(url)
    .then()
        .spec(commonResponseSpec)
    // more verifications
```



**Debugging**  
**peek(), print()**

**Logging**  
**log()**



```
RestAssured
    .given()
        .spec(commonRequestSpec)
        .log().all()
    .when()
        .get(url)
    .then()
        .spec(commonResponseSpec)
        .log().headers()
```



```
.log()  
    .ifStatusCodeMatches( ...)  
    .ifError( ...)  
    .ifValidationFails( ...)
```



```
RestAssured.baseURI = "https://your.api/";  
RestAssuredbasePath = "some/path";
```



```
var commonResponseSpec =  
    new ResponseSpecBuilder()  
        .expectStatusCode(200)  
        .expectContentType(MediaType.JSON)  
        .build();  
  
// [...]  
  
@Test  
RestAssured.get(url)  
    .then()  
        .spec(commonResponseSpec)  
    // ...
```



- ② **failureConfig**(FailureConfig... RestAssuredConfig)
- ② **logConfig**(LogConfig logConf... RestAssuredConfig)
- ② **connectionConfig**(Connection... RestAssuredConfig)
- ② **decoderConfig**(DecoderConfig... RestAssuredConfig)
- ② **csrfConfig**(CsrfConfig csrfC... RestAssuredConfig)
- ② **redirect**(RedirectConfig red... RestAssuredConfig)
- ② **objectMapperConfig**(ObjectMa... RestAssuredConfig)
- ② **and()** RestAssuredConfig
- ② **encoderConfig**(EncoderConfig... RestAssuredConfig)
- ② **getFailureConfig()** FailureConfig
- ② **headerConfig**(HeaderConfig h... RestAssuredConfig)
- ② **httpClient**(HttpClientConfig... RestAssuredConfig)

Press Enter to insert, Tab to replace [Next Tip](#) :

RestAssured.config() .



Up Next:

## Writing Efficient and Effective Test Automation Code

---

