REST Assured POST Request

In **REST Assured**, you can perform a **POST request** to send data (usually in JSON or XML format) to a server, often to create a new resource.

✅ Basic Syntax of POST Request in REST Assured

```
java

given()
    .header("Content-Type", "application/json") // set header
    .body(jsonPayload) // set payload
.when()
    .post("https://api.example.com/endpoint") // set URL
.then()
    .statusCode(201); // validate response status code
```

🧪 Example: POST Request with JSON Body

Let's say we want to create a new user:

➤ JSON Payload

```
json

{
    "name": "John Doe",
    "job": "Software Developer"
}
```

➤ REST Assured Code

```
java

import io.restassured.RestAssured;
import io.restassured.response.Response;

import static io.restassured.RestAssured.*;
import static org.hamcrest.Matchers.*;

public class PostExample {
    public static void main(String[] args) {
        // Define the base URI
        RestAssured.baseURI = "https://reqres.in";

        // Define the JSON body
        String jsonBody = "{ \"name\": \"John Doe\", \"job\": \"Software Developer\" }";

        // Send POST request
        Response response =
            given()
                .header("Content-Type", "application/json")
                .body(jsonBody)
            .when()
                .post("/api/users")
            .then()
                .statusCode(201)
                .body("name", equalTo("John Doe"))
                .body("job", equalTo("Software Developer"))
                .extract().response();

        // Print response
        System.out.println("Response: " + response.asString());
    }
}
```

🔧 Optional Enhancements

Feature	Example
Add auth token	`.header("Authorization", "Bearer your_token")`
Use external JSON file	`new String(Files.readAllBytes(Paths.get("payload.json")))`
Validate headers	`.header("Content-Type", containsString("application/json"))`

📦 Maven Dependency (if not added)

Add this to your `pom.xml`:

```xml
<dependency>
    <groupId>io.rest-assured</groupId>
    <artifactId>rest-assured</artifactId>
    <version>5.4.0</version>
    <scope>test</scope>
</dependency>
```

Let me know if you want an example with external file, POJO class, or any specific API.

✅ Validating POST Response in REST Assured

When you make a **POST request**, it's essential to **validate** that the response meets your expectations — like status code, response body fields, headers, etc.

🔍 What Can You Validate?

Validation Type	Example
Status Code	`201`, `200`
Response Body	Keys, values, structure
Headers	Content-Type, Server, etc.
Response Time	`lessThan(2000L)` in ms

🧪 Full Example: Validate POST Response

📤 Request Body:

```json
{
    "name": "John Doe",
    "job": "Software Developer"
}
```

✅ REST Assured Code with Validations:

```java
import io.restassured.RestAssured;
import io.restassured.response.Response;

import static io.restassured.RestAssured.*;
import static org.hamcrest.Matchers.*;

public class PostValidationExample {
    public static void main(String[] args) {

        RestAssured.baseURI = "https://reqres.in";

        String jsonBody = "{ \"name\": \"John Doe\", \"job\": \"Software Developer\" }";

        Response response =
            given()
                .header("Content-Type", "application/json")
                .body(jsonBody)
            .when()
                .post("/api/users")
            .then()
                // ✔ Status Code
                .statusCode(201)

                // ✔ Response Headers
                .header("Content-Type", containsString("application/json"))

                // ✔ Response Body
                .body("name", equalTo("John Doe"))
                .body("job", equalTo("Software Developer"))
                .body("id", notNullValue()) // usually an auto-generated ID
                .body("createdAt", notNullValue()) // auto timestamp

                // ✔ Response Time (optional)
                .time(lessThan(2000L))

                .extract().response();

        System.out.println("Response Body:\n" + response.asPrettyString());
    }
}
```

📌 Explanation of Validations

Line	What it Validates
`.statusCode(201)`	Resource created successfully
`.header("Content-Type", ...)`	Response format is JSON
`.body("name", equalTo(...))`	Server echoed or processed name correctly
`.body("id", notNullValue())`	Resource ID was generated
`.time(lessThan(2000L))`	Response returned within 2 seconds

Let me know if you want:

- ✅ Schema validation
- ✅ Custom assertions
- ✅ Response saved to a file
- ✅ Data-driven tests (e.g. TestNG + Excel)

I'll help you with examples.

✅ Extracting `id` or `token` from a POST Response in REST Assured

When you send a **POST request** (like login or create user), the server often returns a **`token`** or **`id`** in the **JSON response**.

You can extract these values using **`.jsonPath().getString("key")`**.

🧪 Example 1: Extracting `id` from a Create User Response

🔹 Sample Response from `https://reqres.in/api/users`

```
json

{
    "name": "John",
    "job": "Developer",
    "id": "123",
    "createdAt": "2025-07-28T08:30:00.000Z"
}
```

✅ Code to Extract `id`:

```
java

import io.restassured.RestAssured;
import io.restassured.response.Response;

import static io.restassured.RestAssured.*;

public class ExtractIDExample {
    public static void main(String[] args) {

        RestAssured.baseURI = "https://reqres.in";

        String jsonBody = "{ \"name\": \"John\", \"job\": \"Developer\" }";

        Response response =
            given()
                .header("Content-Type", "application/json")
                .body(jsonBody)
            .when()
                .post("/api/users")
            .then()
                .statusCode(201)
                .extract().response();

        // Extract ID
        String id = response.jsonPath().getString("id");

        System.out.println("Extracted ID: " + id);
    }
}
```

🧪 Example 2: Extracting `token` from Login Response

🔹 Sample Response from Login:

```json
{
    "token": "QpwL5tke4Pnpja7X4"
}
```

✅ Code to Extract `token`:

```java
String jsonBody = "{ \"email\": \"eve.holt@reqres.in\", \"password\": \"cityslicka\" }";

Response response =
    given()
        .header("Content-Type", "application/json")
        .body(jsonBody)
    .when()
        .post("/api/login")
    .then()
        .statusCode(200)
        .extract().response();

// Extract token
String token = response.jsonPath().getString("token");
System.out.println("Extracted Token: " + token);
```

🔁 Bonus: Reuse Extracted Token in Next Request

```java
given()
    .header("Authorization", "Bearer " + token)
    .when()
    .get("/api/protected")
    .then()
    .statusCode(200);
```

Let me know if you need:

- 🔁 Chaining requests using token
- 🧪 TestNG/JUnit integration
- 📂 Save ID/token to external file for reuse

I'll help you with step-by-step code.