

# 1.Add New User

## Program

```
1 package com.TelecomDomain;
2
3 import static io.restassured.RestAssured.given;
4
5 Run ALL
6
7 public class AddUser {
8     String token;
9
10    @Test
11    Run / Debug
12    public void addNewUser() {
13        System.out.println("Add New User");
14        telecomUserDetails userDetails=new telecomUserDetails();
15        userDetails.setFirstName("Lekha");
16        userDetails.setLastName("User");
17        userDetails.setEmail("lekha"+System.currentTimeMillis()+"@fake.com");
18        userDetails.setPassword("test123");
19        Response res=given()
20            .header("accept","application/json")
21            .header("Content-Type","application/json")
22            .body(userDetails)
23            .when().post("https://thinking-tester-contact-list.herokuapp.com/users");
24        res.then().log().body();
25
26        int statusCode=res.getStatusCode();
27        Assert.assertEquals(statusCode, 201,"Status Code not matched!");
28        System.out.println("Status code matched! Code is : "+statusCode);
29        String userId=res.jsonPath().getString("_id");
30        System.out.println("UserId is : "+userId);
31        token=res.jsonPath().getString("token");
32        System.out.println("Generated Token is : "+token);
33    }
34 }
35 }
```

## Create Pojo class for UserDetails

```
public class telecomUserDetails {
```

```
    private String firstName;
```

```
    private String lastName;
```

```
    private String email;
```

```
    private String password;
```

```
    public String getFirstName() {
```

```
        return firstName;
```

```
}  
public void setFirstName(String firstName) {  
    this.firstName = firstName;  
}  
public String getLastName() {  
    return lastName;  
}  
public void setLastName(String lastName) {  
    this.lastName = lastName;  
}  
public String getEmail() {  
    return email;  
}  
public void setEmail(String email) {  
    this.email = email;  
}  
public String getPassword() {  
    return password;  
}  
public void setPassword(String password) {  
    this.password = password;  
}  
}
```

OutPut:

<terminated> AddUser [TextNG] C:\Program Files\Java\jdk-23\bin\javaw.exe (Dec 13, 2024, 8:13:39PM - 8:13:48PM) [pid: 15340]

SLF4J: See <http://www.slf4j.org/codes.html#StaticLoggerBinder> for further details.

Add New User

```
{
  "user": {
    "_id": "675c4823fe9513001317b089",
    "firstName": "Lekha",
    "lastName": "User",
    "email": "lekha1734101021788@fake.com",
    "__v": 1
  },
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfawQiOiI2NzVjNDgyM2ZlOTUxMzAwMTMxN2IwODkiLCJpY"
}
```

Status code matched! Code is : 201

UserId is : null

Generated Token is : eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfawQiOiI2NzVjNDgyM2ZlOTUxMzAwMTMxN2IwOD

PASSED: com.TelecomDomain.AddUser.addNewUser

=====

Default test

Tests run: 1, Failures: 0, Skips: 0

## 2.GetUserDetails

```
35 @Test(priority=2)
    Run / Debug
36 public void getUserDetails() {
37     System.out.println("User Details are");
38     Response res=given()
39         .header("accept","application/json")
40         .header("Content-Type","application/json")
41         .header("Authorization ","Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfawQioiI
42         .when().get("https://thinking-tester-contact-list.herokuapp.com/users/me");
43     res.then().log().body();
44     int statusCode=res.getStatusCode();
45     Assert.assertEquals(statusCode, 200,"Status Code not matched!");
46     System.out.println("Status code matched! Code is : "+statusCode);
47     String statusLine=res.getStatusLine();
48
49 }
```

SLF4J: See <http://www.slf4j.org/codes.html#StaticLoggerBinder> for further details.

User Details are

```
{
  "_id": "675c4efcfe9513001317b0a0",
  "firstName": "Lekha",
  "lastName": "User",
  "email": "lekha1734102774793@fake.com",
  "__v": 1
}
```

Status code matched! Code is : 200

PASSED: com.TelecomDomain.AddUser.getUserDetails

```
=====
Default test
Tests run: 1, Failures: 0, Skips: 0
=====
```

## 3.Update User(Patch)

```
55 public void updateUser() {
56     System.out.println("Update User");
57     HashMap<String,Object> Details=new HashMap<String,Object>();
58     Details.put("firstName","Updated");
59     Details.put("lastName","B");
60     Details.put("email","lekhaQa@fake.com");
61     Details.put("password","lekha123");
62     Response res=given()
63         .header("accept","application/json")
64         .header("Content-Type","application/json")
65         .header("Authorization","Bearer " +token)
66         .body(Details)
67         .when().patch("https://thinking-tester-contact-list.herokuapp.com/users/me");
68     res.then().log().body();
69     statusCode=res.getStatusCode();
70     Assert.assertEquals(statusCode, 200,"Status Code not matched!");
71     System.out.println("Status code matched! Code is : "+statusCode);
72     String statusLine=res.getStatusLine();
73     System.out.println(statusLine);
74 }
```

OutPut:

Update User

```
{  
  "_id": "675fef6f12baa90013d5e64f",  
  "firstName": "Updated",  
  "lastName": "B",  
  "email": "lekhaqa@fake.com",  
  "__v": 1  
}
```

Status code matched! Code is : 200

HTTP/1.1 200 OK

PASSED: com.TelecomDomain.AddUser.addNewUser

PASSED: com.TelecomDomain.AddUser.getUserDetails

PASSED: com.TelecomDomain.AddUser.updateUser

## 4. Log In User

```
76= @Test(priority=4, enabled=true)
    Run / Debug
77 public void loginUser() {
78     System.out.println("UserLogin!!!");
79     Response res=given()
80         .header("accept","application/json")
81         .header("Content-Type","application/json")
82         .body("{\n"
83             + "    \"email\": \"lekhaQa@fake.com\",\n"
84             + "    \"password\": \"lekha123\"\n"
85             + "}")
86         .when().post("https://thinking-tester-contact-list.herokuapp.com/users/login");
87     res.then().log().body();
88     statusCode=res.getStatusCode();
89     Assert.assertEquals(statusCode, 200,"Status Code not matched!");
90     System.out.println("Status code matched! Code is : "+statusCode);
91     String statusLine=res.getStatusLine();
92     System.out.println(statusLine);
93     usertoken=res.jsonPath().getString("token");
94     System.out.println("Generated User Token is : "+usertoken);
95 }
```

### OutPut:

```
UserLogin!!!
{
  "user": {
    "_id": "675fef6f12baa90013d5e64f",
    "firstName": "Updated",
    "lastName": "B",
    "email": "lekhaqa@fake.com",
    "__v": 2
  },
  "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJfawQiOiI2NzVmZWY2ZjEyYmFhOTAwMTNkNWU2NGYiLCJpY"
}
Status code matched! Code is : 200
HTTP/1.1 200 OK
PASSED: com.TelecomDomain.AddUser.loginUser

=====
Default test
Tests run: 1, Failures: 0, Skips: 0
```

## 5.Add Contact

```
public void addContact() {
    System.out.println("Add contact");
    Response res=given()
        .header("accept","application/json")
        .header("Content-Type","application/json")
        .header("Authorization","Bearer "+usertoken)
        .body("{\n"
            + "    \"firstName\": \"Abhi\", \n"
            + "    \"lastName\": \"Sreeram\", \n"
            + "    \"birthdate\": \"2020-11-20\", \n"
            + "    \"email\": \"abhi@fake.com\", \n"
            + "    \"phone\": \"8005555555\", \n"
            + "    \"street1\": \"1 Main St.\", \n"
            + "    \"street2\": \"Apartment A\", \n"
            + "    \"city\": \"Anytown\", \n"
            + "    \"stateProvince\": \"KS\", \n"
            + "    \"postalCode\": \"12345\", \n"
            + "    \"country\": \"India\" \n"
            + "}")
        .when().post("https://thinking-tester-contact-list.herokuapp.com/contacts");
    res.then().log().body();
    statusCode=res.getStatusCode();
}
```

## Output:

```
<terminated> AddUser.addContact | testVb | C:\Program Files\Java\jdk-8.0.60\bin\javaw.exe (Dec 16, 2024, 3:49:14 PM - 3:49:20 PM) [pid: 14220]
HTTP/1.1 200 OK
Generated User Token is : eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
Add contact
{
  "_id": "675ffead6326ce001390dd71",
  "firstName": "Abhi",
  "lastName": "Sreeram",
  "birthdate": "2020-11-20",
  "email": "abhi@fake.com",
  "phone": "8005555555",
  "street1": "1 Main St.",
  "street2": "Apartment A",
  "city": "Anytown",
  "stateProvince": "KS",
  "postalCode": "12345",
  "country": "India",
  "owner": "675fef6f12baa90013d5e64f",
  "__v": 0
}
Code matched!!Status Code is :201
HTTP/1.1 201 Created
```

Search:

Passed: 2 Failed: 0 Skipped: 0

Tests: 1/1 Methods: 2 (8422 ms)

All Tests | Failed Tests | Summary

- Default suite (2/0/0) (7.912 s)
  - Default test (7.912 s)
    - com.TelecomDomain.AddUser
      - loginUser (6.638 s)
      - addContact (1.274 s)

Failure Exception

## 6. Get Contact List

```
124 @Test(priority=6,dependsOnMethods="loginUser")
    Run / Debug
125 public void getContactList() {
126     System.out.println("Get Contact List :");
127     Response res=given()
128         .header("Authorization","Bearer "+usertoken)
129         .when().get("https://thinking-tester-contact-list.herokuapp.com/contacts");
130     res.then().log().body();
131     String statusLine=res.getStatusLine();
132     System.out.println(statusLine);
133     Assert.assertEquals(statusLine,"HTTP/1.1 200 OK" ,"Not Matched") ;
134     System.out.println("Status Line matched!");
135 }
136 }
137 }
```

### OutPut:

```
Get Contact List :
[
  {
    "_id": "675ffd296326ce001390dd6e",
    "firstName": "John",
    "lastName": "Doe",
    "birthdate": "1970-01-01",
    "email": "jdoe@fake.com",
    "phone": "8005555555",
    "street1": "1 Main St.",
    "street2": "Apartment A",
    "city": "Anytown",
    "stateProvince": "KS",
    "postalCode": "12345",
    "country": "USA",
    "owner": "675fef6f12baa90013d5e64f",
    "__v": 0
  },
```

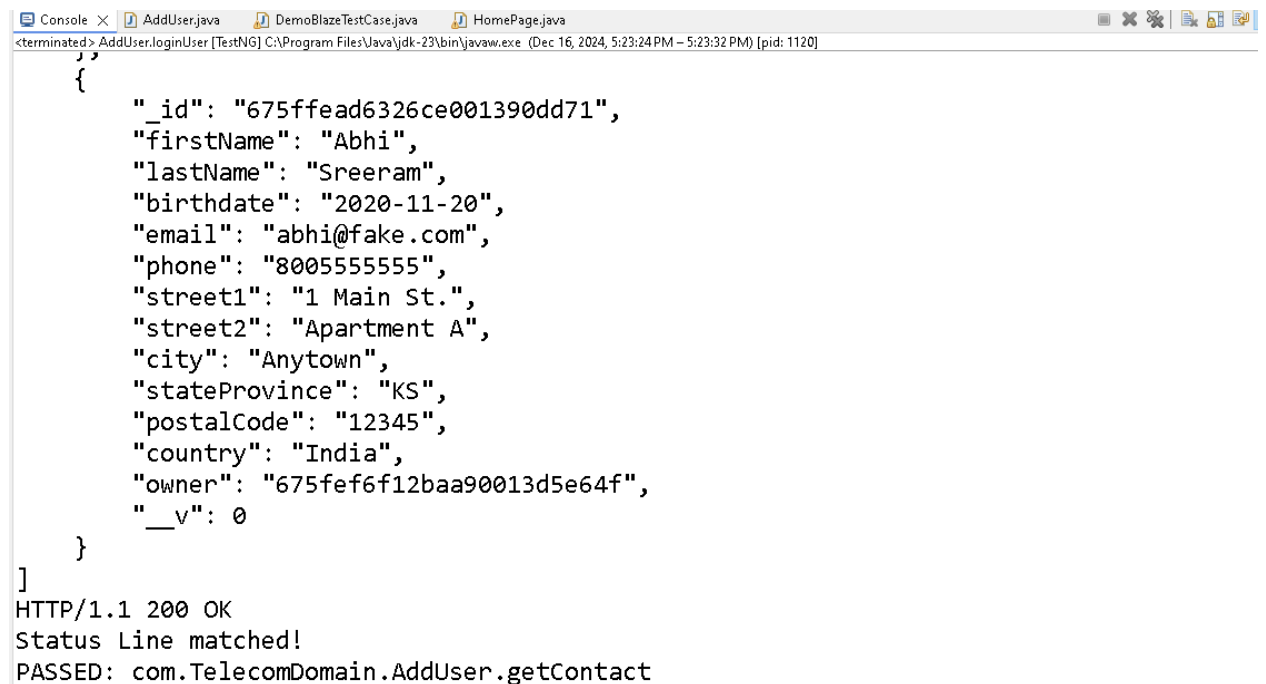


```
{
  "_id": "675ffead6326ce001390dd71",
  "firstName": "Abhi",
  "lastName": "Sreeram",
  "birthdate": "2020-11-20",
  "email": "abhi@fake.com",
  "phone": "8005555555",
  "street1": "1 Main St.",
  "street2": "Apartment A",
  "city": "Anytown",
  "stateProvince": "KS",
  "postalCode": "12345",
  "country": "India",
  "owner": "675fef6f12baa90013d5e64f",
  "__v": 0
}
]
HTTP/1.1 200 OK
Status Line matched!
PASSED: com.TelecomDomain.AddUser.loginUser
PASSED: com.TelecomDomain.AddUser.getContactList
```

## 7.Get Contact

```
137= @Test(priority=7,dependsOnMethods="loginUser")
      Run | Debug
138 public void getContact() {
139     System.out.println("User Contact details are:");
140     Response res=given()
141         .header("Authorization","Bearer "+usertoken)
142         .when().get("https://thinking-tester-contact-list.herokuapp.com/contacts/");
143     res.then().log().body();
144     statusLine=res.getStatusLine();
145     System.out.println(statusLine);
146     Assert.assertEquals(statusLine,"HTTP/1.1 200 OK" ,"Not Matched") ;
147     System.out.println("Status Line matched!");
148 }
```

## OutPut:



```
<terminated> AddUser.loginUser [TestNG] C:\Program Files\Java\jdk-23\bin\javaw.exe (Dec 16, 2024, 5:23:24 PM - 5:23:32 PM) [pid: 1120]
{
  "_id": "675ffead6326ce001390dd71",
  "firstName": "Abhi",
  "lastName": "Sreeram",
  "birthdate": "2020-11-20",
  "email": "abhi@fake.com",
  "phone": "8005555555",
  "street1": "1 Main St.",
  "street2": "Apartment A",
  "city": "Anytown",
  "stateProvince": "KS",
  "postalCode": "12345",
  "country": "India",
  "owner": "675fef6f12baa90013d5e64f",
  "__v": 0
}
HTTP/1.1 200 OK
Status Line matched!
PASSED: com.TelecomDomain.AddUser.getContact
```

## 8.Update Contact (PUT)

```
159 @Test(priority=8,dependsOnMethods="addContact")
    Run / Debug
160 public void updateContact() {
161     System.out.println("Updated contact details are:");
162     Response res=given()
163         .header("Authorization","Bearer "+usertoken)
164         .header("accept","application/json")
165         .header("Content-Type","application/json")
166         .body("{\n"
167             + "    \"firstName\": \"abhi\", \n"
168             + "    \"lastName\": \"B\", \n"
169             + "    \"birthdate\": \"2020-11-20\", \n"
170             + "    \"email\": \"abhinav@fake.com\", \n"
171             + "    \"phone\": \"8005554242\", \n"
172             + "    \"street1\": \"13 School St.\", \n"
173             + "    \"street2\": \"Apt. 5\", \n"
174             + "    \"city\": \"Washington\", \n"
175             + "    \"stateProvince\": \"QC\", \n"
176             + "    \"postalCode\": \"A1A1A1\", \n"
177             + "    \"country\": \"India\" \n"
178             + "}")

179     .when().put("https://thinking-tester-contact-list.herokuapp.com/contacts/"+userI
180 //676044ff3bea55001378cd14
181 res.then().log().body();
182 statusLine=res.getStatusLine();
183 System.out.println(statusLine);
184 Assert.assertEquals(statusLine,"HTTP/1.1 200 OK" ,"Not Matched") ;
185 System.out.println("Status Line matched!");
186 String email=res.jsonPath().getString("email");
187 System.out.println(email);
188 Assert.assertEquals(email,"abhinav@fake.com" ,"Not Matched") ;
189 System.out.println("Email Id Matched and contact got updated");
190 }
```

## OutPut:

```
{
  "_id": "6760586d6326ce001390e053",
  "firstName": "abhi",
  "lastName": "B",
  "birthdate": "2020-11-20",
  "email": "abhinav@fake.com",
  "phone": "8005554242",
  "street1": "13 School St.",
  "street2": "Apt. 5",
  "city": "Washington",
  "stateProvince": "QC",
  "postalCode": "A1A1A1",
  "country": "India",
  "owner": "675fef6f12baa90013d5e64f",
  "__v": 0
}
```

HTTP/1.1 200 OK

Status Line matched!

abhinav@fake.com

Email Id Matched and contact got updated

---

## 9. Partial contactUPdate(Patch)

```
191 @Test(priority=9,dependsOnMethods="addContact")
    Run | Debug
192 public void partialContactUpdate() {
193     System.out.println("Updating partial details of contact");
194     Response res=given()
195         .header("Authorization","Bearer "+usertoken)
196         .body("{\n"
197             + "    \"firstName\": \"Abhi\"\n"
198             + "}")
199     .when().patch("https://thinking-tester-contact-list.herokuapp.com/contacts/"+userId);
200     res.then().log().body();
201     statusLine=res.getStatusLine();
202     System.out.println(statusLine);
203     Assert.assertEquals(statusLine,"HTTP/1.1 200 OK" ,"Not Matched") ;
204     System.out.println("Status Line matched!");
205     //validate for firstName="AbhinavSreeram"
206     String fName=res.jsonPath().get("firstName");
207     System.out.println(fName);
208     Assert.assertEquals(fName,"Abhi" ,"FirstName Not Matched") ;
209     System.out.println("FirstName matched!");
210 }
```

### OutPut:

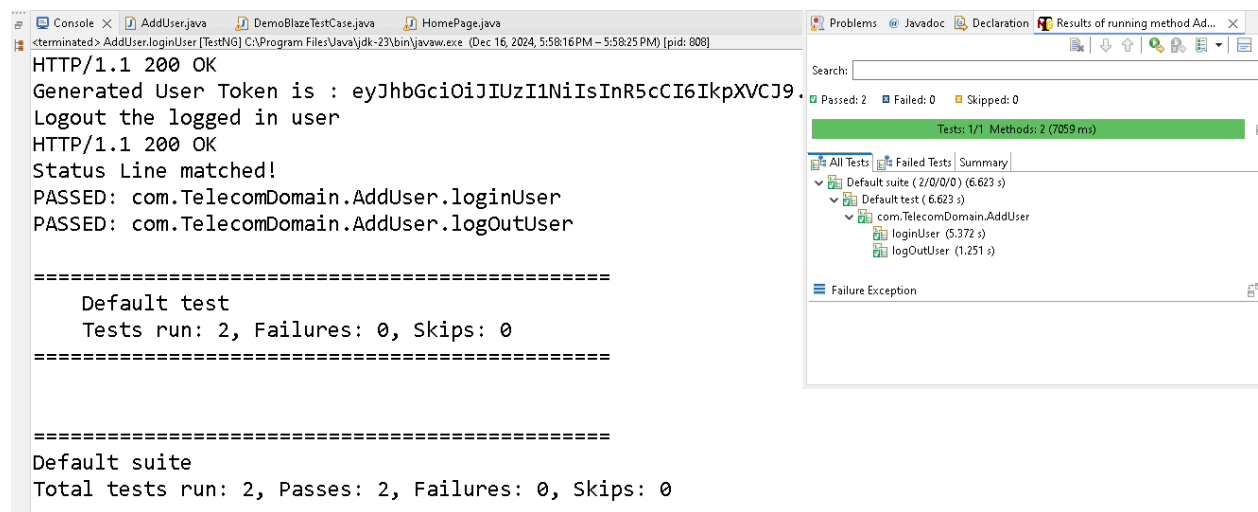
```
Updating partial details of contact
{
  "_id": "676056cd3bea55001378cd85",
  "firstName": "Abhi",
  "lastName": "Sreeram",
  "birthdate": "2020-11-20",
  "email": "abhi@fake.com",
  "phone": "8005555555",
  "street1": "1 Main St.",
  "street2": "Apartment A",
  "city": "Anytown",
  "stateProvince": "KS",
  "postalCode": "12345",
  "country": "India",
  "owner": "675fef6f12baa90013d5e64f",
  "__v": 0
}
HTTP/1.1 200 OK
Status Line matched!
Abhi
FirstName matched!
```



## 10.LogOut User

```
198 @Test(priority=10,dependsOnMethods="loginUser")
    Run / Debug
199 public void logOutUser() {
200     System.out.println("Logout the logged in user");
201     Response res=given()
202         .header("Authorization","Bearer "+usertoken)
203         .when().post("https://thinking-tester-contact-list.herokuapp.com/users/logout");
204     statusLine=res.getStatusLine();
205     System.out.println(statusLine);
206     Assert.assertEquals(statusLine,"HTTP/1.1 200 OK" ,"Not Matched" );
207     System.out.println("Status Line matched!");
208 }
```

## OutPut:



The screenshot displays an IDE with two main panels. The left panel shows the console output of a test run, and the right panel shows the test results summary.

**Console Output:**

```
<terminated> AddUser.loginUser [TestNG] C:\Program Files\Java\jdk-23\bin\javaw.exe (Dec 16, 2024, 5:58:16 PM - 5:58:25 PM) [pid: 808]
HTTP/1.1 200 OK
Generated User Token is : eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
Logout the logged in user
HTTP/1.1 200 OK
Status Line matched!
PASSED: com.TelecomDomain.AddUser.loginUser
PASSED: com.TelecomDomain.AddUser.logOutUser

=====
      Default test
      Tests run: 2, Failures: 0, Skips: 0
=====

=====
Default suite
Total tests run: 2, Passes: 2, Failures: 0, Skips: 0
=====
```

**Test Results Summary:**

- Passed: 2, Failed: 0, Skipped: 0
- Tests: 1/1, Methods: 2 (7059 ms)
- Summary:
  - Default suite (2/0/0/0) (6.623 s)
    - Default test (6.623 s)
      - com.TelecomDomain.AddUser
        - loginUser (5.372 s)
        - logOutUser (1.251 s)
- Failure Exception

