**Library API :**

BaseURI : http://216.10.245.166

1. **Resource** : Library/Addbook.php       **Method** : POST

**Input Payload : Json**:

{

"name":"Learn Appium Automation with Java",

"isbn":"bcd",

"aisle":"227",

"author":"John foe"

}

**Output Json**

{

"Msg": "successfully added",

"ID": "bcd227"

}

1. **Resource** : /Library/GetBook.php?AuthorName=somename **Method** : GET

**Output Json** :

Output the array of Json object books with all below  details

{

Name : “bookname”   ( String)

Isbn :  “A2fdsf”   (String)

Aisle : 32 (Integer)

}

1. **Resource** : Library/GetBook.php?ID=3389      - **Method** : GET

**Output Json :**

{

"book\_name": "Selenium automation using Java",

"isbn": "a23hd738",

"aisle": "1223"

}

1. **Resource** : /Library/DeleteBook.php      **Method** : POST

**Input Payload : Json:**

{

"ID" : "a23h345122332"

}

**Output Response** :

{  
  
msg : book is successfully deleted”

}

code download

import org.testng.annotations.DataProvider;

import org.testng.annotations.Test;

import static io.restassured.RestAssured.given;

import java.io.IOException;

import java.nio.file.Files;

import java.nio.file.Paths;

import io.restassured.RestAssured;

import io.restassured.path.json.JsonPath;

import io.restassured.response.Response;

import files.ReusableMethods;

import files.payLoad;

public class DynamicJson {

@Test(dataProvider="BooksData")

public void addBook(String isbn,String aisle)

{

RestAssured.baseURI="http://216.10.245.166";

Response resp=given().

header("Content-Type","application/json").

body(payLoad.Addbook(isbn,aisle)).

when().

post("/Library/Addbook.php").

then().assertThat().statusCode(200).

extract().response();

JsonPath js= ReusableMethods.rawToJson(resp);

   String id=js.get("ID");

   System.out.println(id);

   //deleteBOok

}

@DataProvider(name="BooksData")

public Object[][]  getData()

{

//array=collection of elements

//multidimensional array= collection of arrays

return new Object[][] {‌{"ojfwty","9363"},{"cwetee","4253"}, {"okmfet","533"} };

}

}

code download

import static io.restassured.RestAssured.given;

import java.io.IOException;

import java.nio.file.Files;

import java.nio.file.Paths;

import org.testng.annotations.Test;

import files.ReusableMethods;

import files.payLoad;

import io.restassured.RestAssured;

import io.restassured.path.json.JsonPath;

import io.restassured.response.Response;

public class StaticJson {

@Test

public void addBook() throws IOException

{

RestAssured.baseURI="http://216.10.245.166";

Response resp=given().

header("Content-Type","application/json").

body(GenerateStringFromResource("C:\\Users\\rahul\\Documents\\Addbookdetails.json")).

when().

post("/Library/Addbook.php").

then().assertThat().statusCode(200).

extract().response();

JsonPath js= ReusableMethods.rawToJson(resp);

   String id=js.get("ID");

   System.out.println(id);

   //deleteBOok

}

public static String GenerateStringFromResource(String path) throws IOException {

    return new String(Files.readAllBytes(Paths.get(path)));

}

}