

# Exercise

1. Create a Rest API for Student using noun plurals for endpoint and http verbs for different operations.(1 Mark)

The screenshot displays a REST client interface with two panels. The top panel shows a POST request to `http://localhost:8080/students` with a JSON body: `{ "name": "yatin", "age": 24, "contact": 849846494 }`. The bottom panel shows the corresponding GET response, which is a JSON array: `[ { "id": 1, "name": "yatin", "age": 24, "contact": 849846494 } ]`. The interface includes tabs for Authorization, Headers, Body, Pre-request Script, and Tests, as well as a Type dropdown set to 'No Auth'.

**POST Request:**

```
1 {  
2   "name": "yatin",  
3   "age": 24,  
4   "contact": 849846494  
5 }
```

**GET Response:**

```
1 [  
2   {  
3     "id": 1,  
4     "name": "yatin",  
5     "age": 24,  
6     "contact": 849846494  
7   }  
8 ]
```

GET

http://localhost:8080/students/1

AuthorizationHeaders (1)BodyPre-request ScriptTests

Type

No Auth

BodyCookiesHeaders (3)Test Results

Pretty

Raw

Preview

JSON



```
1 {
2   "id": 1,
3   "name": "yatin",
4   "age": 24,
5   "contact": 849846494
6 }
```

PUT

http://localhost:8080/studen

AuthorizationHeaders (1)Body

☐ form-data ☐ x-www-form-urlencoded

```
1 {
2   "id":1,
3   "name": "yatin",
4   "age":26,
5   "contact":849846494
6 }
```

BodyCookiesHeaders (3)Test Res

Pretty

Raw

Preview

JSON

```
1 {
2   "id": 1,
3   "name": "yatin",
4   "age": 26,
5   "contact": 849846494
6 }
```

http://localhost:8080/ http://localhost:8080/ + ...

DELETE

http://localhost:8080/students/1

AuthorizationHeaders (1)BodyPre-request ScriptTests

## 2. Create a Customise Exception Handler.(1 Mark)

The screenshot shows a Postman interface for a GET request to `http://localhost:8080/students/10`. The request is in the 'GET' tab, and the response is in the 'Body' tab. The response is a JSON object with the following fields:

```
{
  "timestamp": "2019-03-28T17:06:01.074+0000",
  "status": 404,
  "error": "Not Found",
  "message": "Student not found!!",
  "path": "/students/10"
}
```

The response is displayed in the 'Body' tab, showing the JSON structure and the status code 404.

## 3. Print the Validation Messages of student Entity in response.(1 Mark)

The screenshot shows a Postman interface for a POST request to `http://localhost:8080/students/`. The request is in the 'POST' tab, and the response is in the 'Body' tab. The request body is a JSON object with the following fields:

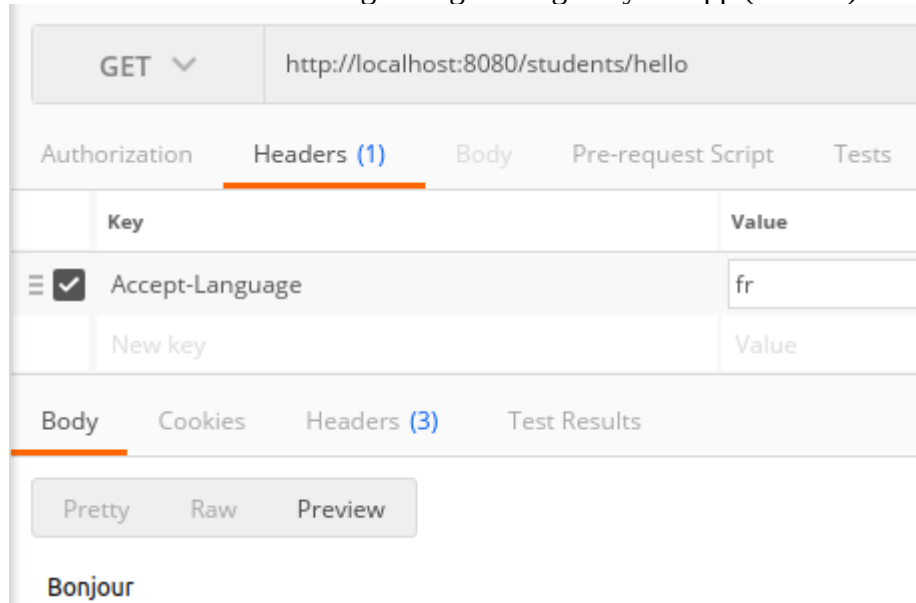
```
{
  "name": "yati",
  "age": 24,
  "contact": 849846494
}
```

The response is a JSON object with the following fields:

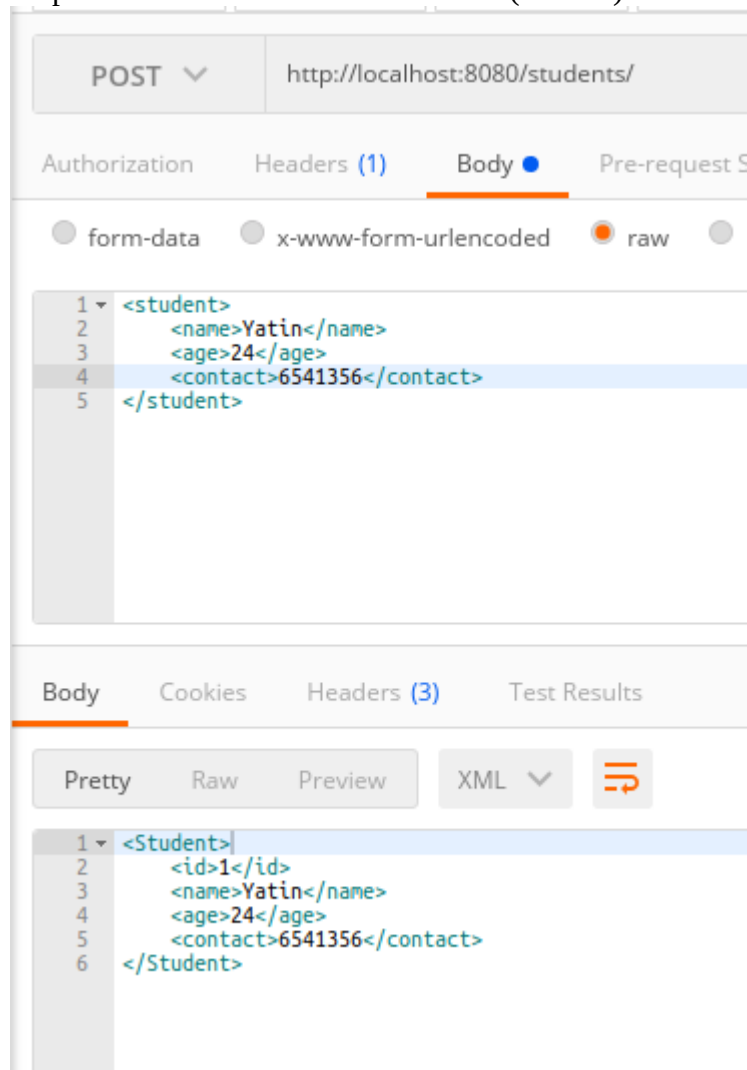
```
{
  "timestamp": "2019-03-28T18:05:35.820+0000",
  "message": "Cannot process the request.",
  "details": "[Name should have at least 5 characters]"
}
```

The response is displayed in the 'Body' tab, showing the JSON structure and the status code 422 Unprocessable Entity.

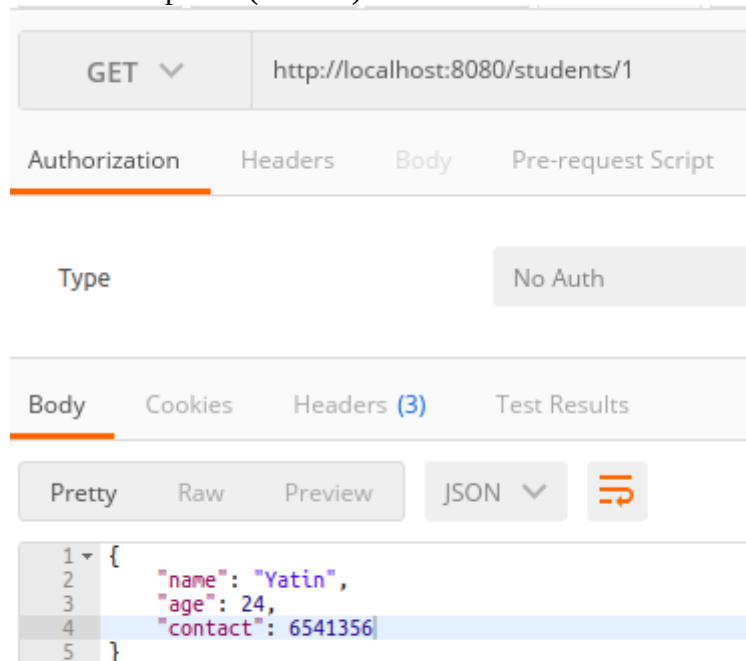
4. Perform Internationalization for a greeting message in your app.(1 Mark)



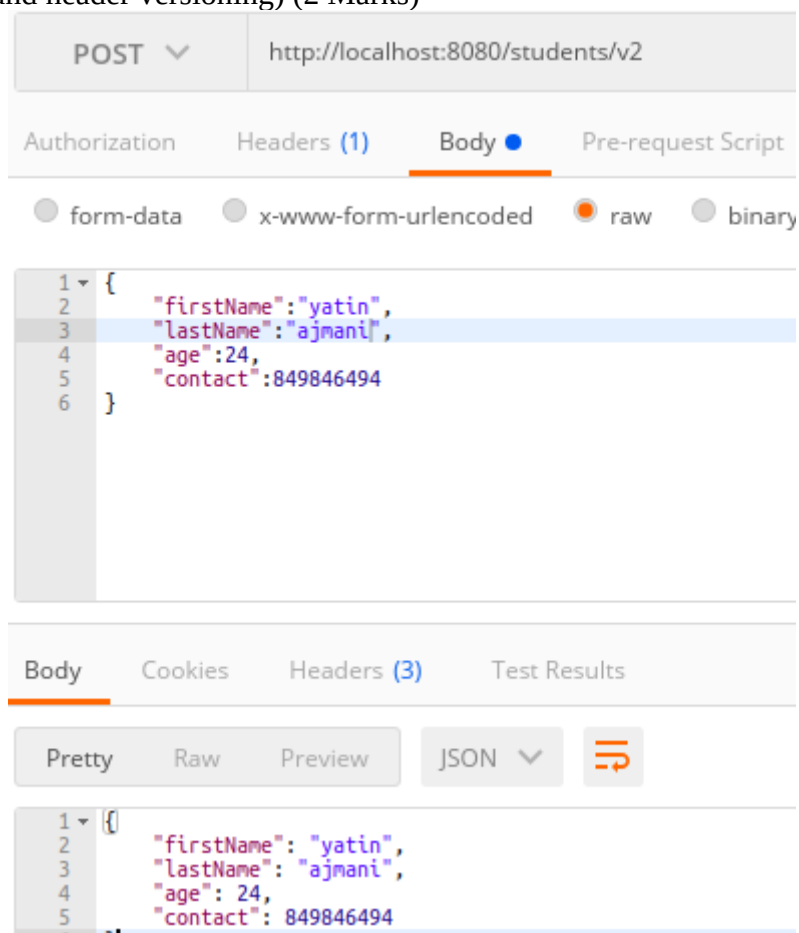
5. Return XML Response when new Student is created.(1 Mark)



6. Ignore ID field in the Response.(1 Mark)



7. Create 2 versions of your API one take represent name of the Student as single string and other showing firstname and lastname separately. (Create the Versions of the API using URI, parameter and header versioning) (2 Marks)



GET

http://localhost:8080/students/student

Authorization

Headers (1)

Body

Pre-request Script

Tests

	Key	Value
<div><div></div><div></div></div>	API-VERSION	1
<div><div></div><div></div></div>	New key	Value

Body

Cookies

Headers (3)

Test Results

Pretty

Raw

Preview

JSON

1

{

2

"name": "Yatin",

3

"age": 5,

4

"contact": 65456435

5

}

GET

http://localhost:8080/students/student

Authorization

Headers (1)

Body

Pre-request Script

Tests

	Key	Value
<div><div></div><div></div></div>	API-VERSION	2
<div><div></div><div></div></div>	New key	Value

Body

Cookies

Headers (3)

Test Results

Pretty

Raw

Preview

JSON

1

{

2

"firstName": "Yatin",

3

"lastName": "Ajmani",

4

"age": 5,

5

"contact": 65456435

6

}

8. Perform CRUD operations on the resource below using RestTemplate  
<https://jsonplaceholder.typicode.com/posts> (2 Marks)

GET <http://localhost:8080/posts/1> Parameters

Authorization Headers Body Pre-request Script Tests

Type No Auth

Body Cookies Headers (3) Test Results

Pretty Raw Preview Text

```
1 {
2   "userId": 1,
3   "id": 1,
4   "title": "sunt aut facere repellat provident occaecati excepturi optio reprehenderit",
5   "body": "quia et suscipit\nsuscipit recusandae consequuntur expedita et cum\nreprehenderit molestiae ut ut quas t
6   rem eveniet architecto"
```

POST <http://localhost:8080/posts/>

Authorization Headers (1) Body Pre-request Script Tests

☐ form-data ☐ x-www-form-urlencoded ☒ raw ☐ binary JSON

```
1 {
2   "userId": 1,
3   "id": 1,
4   "title": "title",
5   "body": "body"
6 }
```

Body Cookies Headers (3) Test Results

Pretty Raw Preview JSON

```
1 {
2   "userId": 1,
3   "id": 101,
4   "title": "title",
5   "body": "body"
6 }
```

http

http

http

http://loc

http

http

http

http

http

http

+

NO Environment

PUT

http://localhost:8080/posts/

Params

Send

Authorization

Headers (1)

Body

Pre-request Script

Tests

form-data

x-www-form-urlencoded

raw

binary

JSON (application/json)

1 {

2   " id": 1,

3   " title": "foo",

4   " body": "bar",

5   " userId": 1

6 }

Body

Cookies

Headers (2)

Test Results

Status: 200 OK

content-length

0

date

Thu, 28 Mar 2019 19:42:13 GMT

DELETE

http://localhost:8080/posts/1

Params

Send

Authorization

Headers

Body

Pre-request Script

Tests

Type

No Auth

Body

Cookies

Headers (2)

Test Results

Status: 200 OK

Pretty

Raw

Preview

Text