

Microservices Communication in Spring Boot

1.RestApiOne:

RestApiOneApplication:

```
package com.example.test;
```

```
import org.springframework.boot.SpringApplication;
```

```
import org.springframework.boot.autoconfigure.SpringBootApplication;
```

```
@SpringBootApplication
```

```
public class RestApiOneApplication {
```

```
    public static void main(String[] args) {
```

```
        SpringApplication.run(RestApiOneApplication.class, args);
```

```
    }
```

```
}
```

PersonEntity:

```
package com.example.test;
```

```
import jakarta.persistence.Column;
```

```
import jakarta.persistence.Entity;
```

```
import jakarta.persistence.GeneratedValue;
```

```
import jakarta.persistence.GenerationType;
```

```
import jakarta.persistence.Id;
```

```
@Entity
```

```
public class PersonEntity {
```

```
    @Id
```

```
    @GeneratedValue(strategy = GenerationType.AUTO)
```

```
@Column(name = "id", updatable = false, nullable = false)
private Integer personId;
@Column
private String name;
@Column
private Integer age;

public PersonEntity() {
    super();
}

public PersonEntity(Integer personId, String name, Integer age) {
    super();
    this.personId = personId;
    this.name = name;
    this.age = age;
}

public Integer getPersonId() {
    return personId;
}

public void setPersonId(Integer personId) {
    this.personId = personId;
}

public String getName() {
    return name;
}

public void setName(String name) {
    this.name = name;
}

public Integer getAge() {
    return age;
}

public void setAge(Integer age) {
```

```
        this.age = age;
    }
}
```

PersonRepository:

```
package com.example.test;
```

```
import org.springframework.data.jpa.repository.JpaRepository;
```

```
import org.springframework.stereotype.Repository;
```

@Repository

```
public interface PersonRepository extends JpaRepository<PersonEntity, Integer> {
```

```
}
```

PersonService:

```
package com.example.test;
```

```
import java.util.HashMap;
```

```
import java.util.Map;
```

```
import org.springframework.beans.factory.annotation.Autowired;
```

```
import org.springframework.stereotype.Service;
```

```
import org.springframework.web.client.RestTemplate;
```

@Service

```
public class PersonService {
```

```
    @Autowired
```

```
    PersonRepository personRepository;
```

```
    RestTemplate restTemplate = new RestTemplate();
```

```

public PersonResponse getPerson(int personId){
    final String uri = "http://localhost:8082/webapitwo/hobby/{personId}";

    Map<String, Integer> params = new HashMap<String, Integer>();
    params.put("personId", personId);

    String result = restTemplate.getForObject(uri, String.class, params);
    PersonEntity pe=personRepository.findById(personId).get();
    PersonResponse pr=new PersonResponse();
    pr.setPersonId(pe.getPersonId());
    pr.setName(pe.getName());
    pr.setAge(pe.getAge());
    pr.setHobby(result);

    return pr;
}

public void addPerson(PersonEntity pe){
    personRepository.save(pe);
}
}

```

PersonController:

```

package com.example.test;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.PathVariable;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;
import org.springframework.web.bind.annotation.RestController;

```

```

@RestController
@RequestMapping(path = "/webapione")

public class PersonController {

    @Autowired
    PersonService personService;

    @RequestMapping("/person/{personId}")
    public PersonResponse getPerson(@PathVariable int personId){
        return personService.getPerson(personId);
    }

    @RequestMapping(method=RequestMethod.POST, value="/person")
    public void addPerson(@RequestBody PersonEntity pe ) {
        personService.addPerson(pe);
    }
}

```

PersonResponse:

```

package com.example.test;

public class PersonResponse {

    private Integer personId;
    private String name;
    private Integer age;
    private String hobby;
    public Integer getPersonId() {
        return personId;
    }
    public void setPersonId(Integer personId) {
        this.personId = personId;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public Integer getAge() {
        return age;
    }
    public void setAge(Integer age) {
        this.age = age;
    }
}

```

```

    }
    public String getHobby() {
        return hobby;
    }
    public void setHobby(String result) {
        this.hobby = result;
    }
}

```

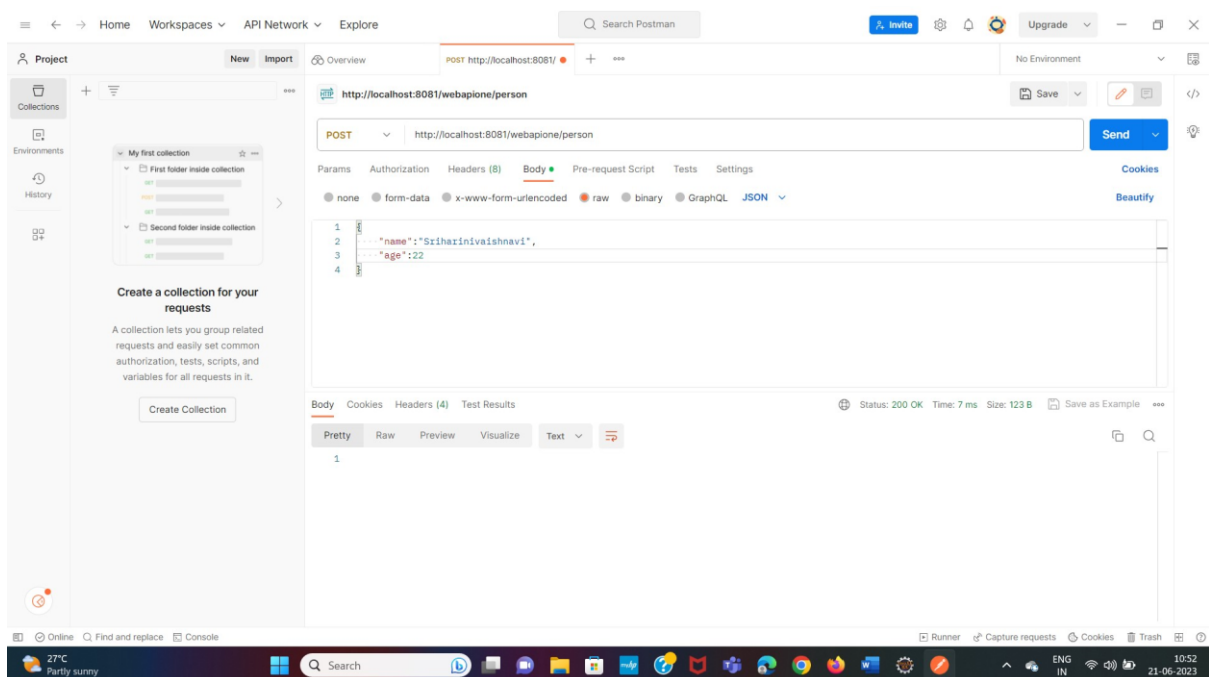
Application Properties:

```

server.port=8081
spring.application.name=RestApiOne

```

Output:



2.RestApiTwo

RestApiTwoApplication:

```
package com.example.test;
```

```
import org.springframework.boot.SpringApplication;
```

```
import org.springframework.boot.autoconfigure.SpringBootApplication;
```

```
@SpringBootApplication
```

```
public class RestApiTwoApplication {
```

```
public static void main(String[] args) {  
    SpringApplication.run(RestApiTwoApplication.class, args);  
}  
  
}
```

HobbyEntity:

```
package com.example.test;
```

```
import jakarta.persistence.Column;  
import jakarta.persistence.Entity;  
import jakarta.persistence.GeneratedValue;  
import jakarta.persistence.GenerationType;  
import jakarta.persistence.Id;
```

@Entity

```
public class HobbyEntity {
```

```
    @Id
```

```
    @GeneratedValue(strategy = GenerationType.AUTO)
```

```
    @Column(name = "id", updatable = false, nullable = false)
```

```
    private Integer id;
```

```
    @Column
```

```
    private Integer personId;
```

```
    @Column
```

```
    private String name;
```

```
    public HobbyEntity() {
```

```
        super();
```

```
    }
```

```
    public HobbyEntity(Integer personId, String name) {
```

```
        super();
```

```

        this.personId = personId;
        this.name = name;
    }
    public Integer getPersonId() {
        return personId;
    }
    public void setPersonId(Integer personId) {
        this.personId = personId;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
}

```

HobbyRepository:

```
package com.example.test;
```

```
import org.springframework.data.jpa.repository.JpaRepository;
```

```
import org.springframework.data.jpa.repository.Query;
```

```
import org.springframework.stereotype.Repository;
```

```
@Repository
```

```
public interface HobbyRepository extends JpaRepository<HobbyEntity, Integer> {
```

```
    @Query("SELECT h.name FROM HobbyEntity h WHERE h.personId=:personId")
```

```
    public String findByPersonId(Integer personId);
```

```
}
```


HobbyService:

```
package com.example.test;
```

```
import org.springframework.beans.factory.annotation.Autowired;
```

```
import org.springframework.stereotype.Service;
```

```
@Service
```

```
public class HobbyService {
```

```
    @Autowired
```

```
    HobbyRepository hobbyRepository;
```

```
    public String findByPersonId(int personid){
```

```
        return hobbyRepository.findByPersonId(personid);
```

```
    }
```

```
    public void addHobby(HobbyEntity he){
```

```
        hobbyRepository.save(he);
```

```
    }
```

```
}
```

HobbyController:

```
package com.example.test;
```

```
import org.springframework.beans.factory.annotation.Autowired;
```

```
import org.springframework.web.bind.annotation.PathVariable;
```

```
import org.springframework.web.bind.annotation.RequestBody;
```

```
import org.springframework.web.bind.annotation.RequestMapping;
```

```
import org.springframework.web.bind.annotation.RequestMethod;
```

```
import org.springframework.web.bind.annotation.RestController;
```

```
@RestController
```

```

@RequestMapping(path = "/webapitwo")
public class HobbyController {

    @Autowired
    HobbyService hobbyService;

    @RequestMapping("/hobby/{personid}")
    public String findByPersonId(@PathVariable int personid){
        return hobbyService.findByPersonId(personid);
    }

    @RequestMapping(method=RequestMethod.POST, value="/hobby")
    public void addHobby(@RequestBody HobbyEntity he ) {
        hobbyService.addHobby(he);
    }

}

```

Application properties:

```

server.port=8082
spring.application.name=RestApiTwo

```

Output:

MAIN OUTPUT:

Postman interface showing a POST request to `http://localhost:8082/webapiwo/hobby`. The request body is a JSON object:

```
{  "person-id": 1,  "name": null}
```

The response status is 200 OK, Time: 231 ms, Size: 123 B. The response body is:

```
1
```

The interface includes a sidebar with Collections, Environments, History, and a 'Create a collection for your requests' prompt. The top bar shows navigation links (Home, Workspaces, API Network, Explore) and a search bar.

Postman interface showing a GET request to `http://localhost:8081/webapione/person/1`. The request body is a JSON object:

```
{  "personId": 1,  "name": "Sriharinivashnavi",  "age": 22,  "hobby": null}
```

The response status is 200 OK, Time: 169 ms, Size: 228 B. The response body is:

```
1 2 3 4 5 6 {  "personId": 1,  "name": "Sriharinivashnavi",  "age": 22,  "hobby": null}
```

The interface includes a sidebar with Collections, Environments, History, and a 'Create a collection for your requests' prompt. The top bar shows navigation links (Home, Workspaces, API Network, Explore) and a search bar.