

SPRING-MVC-UPDATE-02-2025

JSON – JAVA Mapping

Links ---

<https://github.com/hong1234/Spring-Mvc-Data-Binding>

<https://springframework.guru/how-to-the-jackson-object-mapper-with-json/>

<https://github.com/spring-framework-guru/spring-6-posts/tree/main/jackson-examples>

<https://attacomsian.com/blog/processing-json-spring-boot>

<https://medium.com/@vishamberlal/understanding-data-transfer-objects-dto-in-spring-boot-ac06b575a1d5>

Overview ---

Entity ---

ID Int or UUID

Lombok @Data

Field Validation @NotNull

related DB Tables // UUID storing support in DB

H2

MySQL

SPRING MVC ---

(Input-) Data Binding / Jackson Converter: @RequestBody Customer customer

Mapping JSON -> Entity // Reading Json – Deserialization with **the Jackson ObjectMapper**

Output-Data

Mapping Entity -> JSON

Entity Data Validating: @Valid

Exception Handling @RestControllerAdvice

An Example -----

<https://attacomsian.com/blog/processing-json-spring-boot>

// customer.json

```
{
  "id": 123,
  "name": "Jovan Lee",
  "email": "jovan@example.com",
  "phone": "+49 176 14890478",
  "age": 32,

  "address": {
    "street": "Yorckstr. 75",
    "city": "Berlin",
    "zipcode": 10965,
    "country": "Germany"
  },
  "projects": [
```

```

    "Remote Job Board",
    "Data Migration"
  ],

  "paymentMethods": [
    "PayPal",
    "Stripe"
  ],

  "profileInfo": {
    "gender": "Male",
    "married": "Yes",
    "source": "LinkedIn"
  }
}

// Customer.java

import java.util.Arrays;
import java.util.List;
import java.util.Map;

import lombok.Data;

@Data
public class Customer {
    private Long id;
    private String name;
    private String email;
    private String phone;
    private int age;

    private String[] projects;
    private Address address;
    private List<String> paymentMethods;
    private Map<String, Object> profileInfo;
}

// Address.java

@Data
public class Address {
    private String street;
    private String city;
    private int zipcode;
    private String country;
}

Example2
https://github.com/hong1234/Spring-Mvc-Data-Binding

```

```
// json request body
{
    "name": "Test",

    "cards": [
        {
            "comment": "test1",
            "cardType": "HAPPY"
        },
        {
            "comment": "test2",
            "cardType": "SAD"
        }
    ],

    "cards2": {
        "dc0168e1-e0c8-44c6-984c-e0537d77840e" : {
            "comment": "test3",
            "cardType": "HAPPY"
        },

        "452e527f-0921-4972-aa0f-7e1a7a25263c" : {
            "comment": "test4",
            "cardType": "SAD"
        }
    }
}

@Data
public class RetroBoard {

    private UUID id;

    @NotBlank(message = "A name must be provided")
    private String name;

    // private Card card;

    private List<Card> cards = new ArrayList<>();

    private Map<UUID, Card> cards2 = new HashMap<>();
    // private Map<String, Card> cards = new HashMap<>();
}

@Data
public class Card {
    private UUID id;

    @NotBlank
    private String comment;

    @NotNull
    private CardType cardType;

    private UUID retroBoardId;
}
```