### \*\*Chapter 2: Spring Boot Basics (for Payment Integration)\*\*

Before we integrate Cashfree, let’s ensure your Spring Boot application is set up correctly for handling payment requests. Here’s how we’ll cover the basics:

---

#### \*\*Topic 1: Setting Up a Spring Boot Project\*\*

To start integrating any API, including Cashfree, you need to have a Spring Boot application running. Here's a step-by-step guide to set it up:

1. \*\*Create a Spring Boot Project:\*\*

- Use Spring Initializr (https://start.spring.io/) to generate a basic Spring Boot project.

- Select the following dependencies:

- \*\*Spring Web\*\* (for REST API functionality)

- \*\*Spring Boot DevTools\*\* (optional, for auto-reloading)

- \*\*Spring Data JPA\*\* (if you're going to interact with a database)

2. \*\*Project Structure:\*\*

- Once generated, your project structure should look something like this:

```

└── src

└── main

├── java

│ └── com.example.paymentgateway

│ └── PaymentGatewayApplication.java

├── resources

│ ├── application.properties

└── test

└── java

└── com.example.paymentgateway

└── PaymentGatewayApplicationTests.java

```

3. \*\*Application Entry Point:\*\*

- The `PaymentGatewayApplication.java` will look like this:

```java

package com.example.paymentgateway;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class PaymentGatewayApplication {

public static void main(String[] args) {

SpringApplication.run(PaymentGatewayApplication.class, args);

}

}

```

4. \*\*Application Properties:\*\*

- The `application.properties` file in the `src/main/resources` directory is where you'll configure essential settings for your project (e.g., server port, Cashfree API keys, etc.).

- Example:

```properties

server.port=8080

```

---

#### \*\*Topic 2: Introduction to REST APIs in Spring Boot\*\*

You will need to create a REST API to handle requests and interact with Cashfree. Let’s start by setting up a basic REST controller.

1. \*\*Creating a REST Controller:\*\*

- In your project, create a new controller class under `com.example.paymentgateway.controllers`:

```java

package com.example.paymentgateway.controllers;

import org.springframework.web.bind.annotation.GetMapping;

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

@RestController

public class PaymentController {

@GetMapping("/healthcheck")

public String healthCheck() {

return "Payment service is up!";

}

}

```

- The `/healthcheck` endpoint simply returns a message confirming the service is running.

2. \*\*Running the Application:\*\*

- Run your Spring Boot application (either from your IDE or using the command `mvn spring-boot:run`).

- Once it's running, navigate to `http://localhost:8080/healthcheck` to verify the service is up and running.

3. \*\*Handling POST Requests:\*\*

- For payment gateways, you will primarily deal with POST requests to process data (e.g., creating payments, verifying transactions). Here’s an example of a POST endpoint:

```java

@PostMapping("/create-payment")

public ResponseEntity<String> createPayment(@RequestBody PaymentRequest paymentRequest) {

// Placeholder for payment logic

return ResponseEntity.ok("Payment created");

}

```

- The `@RequestBody` annotation allows you to pass a JSON object into the API, which is crucial for interacting with the Cashfree API later.

4. \*\*JSON Request Example:\*\*

- When sending data to this API, the request body might look like this:

```json

{

"amount": 500,

"currency": "INR",

"customerEmail": "test@example.com"

}

```

---

#### \*\*What We’ve Covered So Far:\*\*

1. \*\*Spring Boot Setup\*\*: We created a basic Spring Boot project.

2. \*\*REST Controller\*\*: We set up a simple REST controller to handle GET and POST requests.

3. \*\*Application Running\*\*: Confirmed the application is running on a specific port.

---

\*\*Next Step\*\*: In \*\*Chapter 3\*\*, we'll set up your Cashfree account, access the API keys, and start exploring how to interact with Cashfree from your Spring Boot app.

Let me know if you'd like to dive into \*\*Chapter 3: Setting Up Cashfree Account\*\* or if you have any questions so far!