

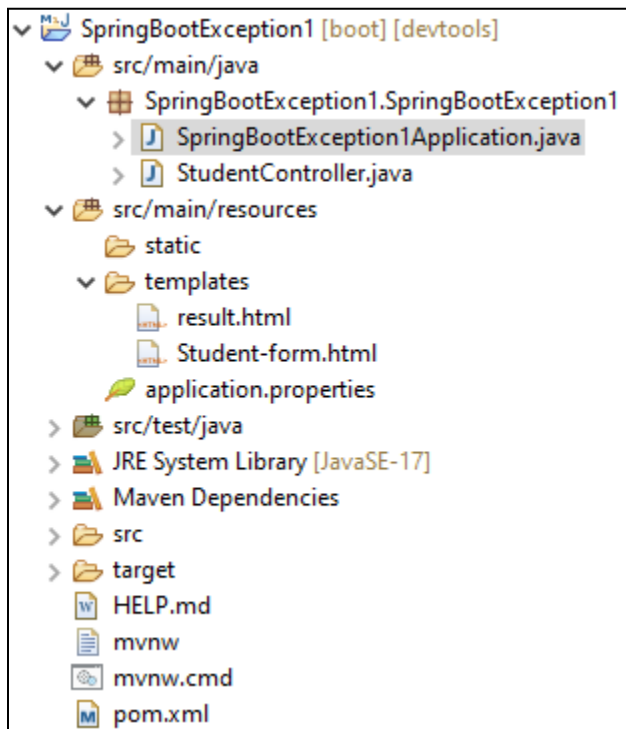
Name of Student: Pushkar Sane		
Roll Number: 45		Tutorial Number: 8
Title of Tutorial: Exception Handling with Spring boot.		
DOP: 28-10-2023		DOS: 30-10-2023
CO Mapped: CO5, CO6	PO Mapped: PO1, PO2, PO3,PO4, PO7, PSO1, PSO2	Signature:

Practical No. 8

Aim:

1. To implement a program to demonstrate Number Format Exception: Enter Roll No, Name, Marks and Subject from command line and calculate percentage.

File Structure:



Code:

SpringBootException1Application.java

```
package SpringBootException1.SpringBootException1;

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

@SpringBootApplication
public class SpringBootException1Application {
    public static void main(String[] args) {
```

```
        SpringApplication.run(SpringBootException1Application.class, args);
    }
}
```

StudentController.java

```
package SpringBootException1.SpringBootException1;

import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestParam;

@Controller
public class StudentController {
    @GetMapping("/calculate")
    public String showForm() {
        return "student-form";
    }

    @PostMapping("/calculate")
    public String ShowResult(@RequestParam String rollNo, @RequestParam
String name, @RequestParam String subject,
        @RequestParam String marks, Model model) throws
NumberFormatException {
        try {
            double mark = Double.parseDouble(marks);
            double percentage = (mark / 100) * 100;
            model.addAttribute("name", name);
            model.addAttribute("rollNo", rollNo);
            model.addAttribute("subject", subject);
            model.addAttribute("marks", marks);
            model.addAttribute("percentage", percentage);
            return "result";
        }
    }
}
```

```
        } catch (NumberFormatException ex) {
            model.addAttribute("error", "Invalid marks entered");
            return "student-form";
        }
    }
}
```

result.html

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
    <title>Result</title>
</head>
<body>
    <h2>Student Information</h2>
    <p>Roll No: <span th:text="${rollNo}"></span></p>
    <p>Name: <span th:text="${name}"></span></p>
    <p>Marks: <span th:text="${marks}"></span></p>
    <p>Subject: <span th:text="${subject}"></span></p>
    <p>Percentage: <span th:text="${percentage}"></span>%</p>
</body>
</html>
```

Student-form.html

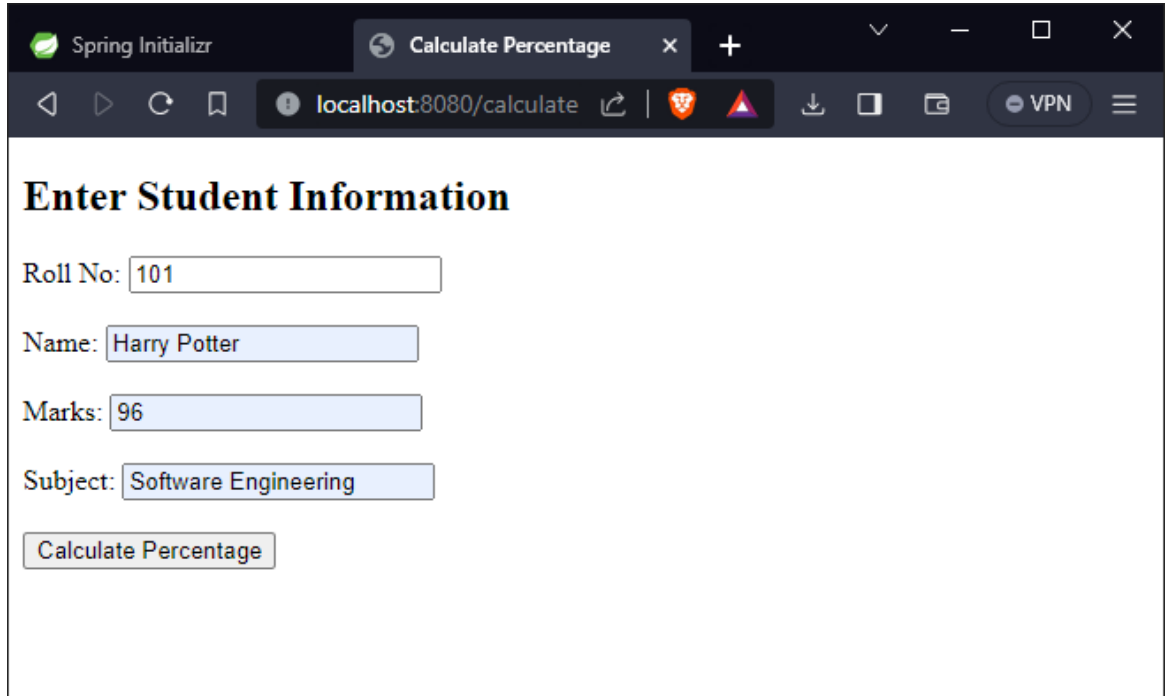
```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
    <title>Calculate Percentage</title>
</head>
<body>
    <h2>Enter Student Information</h2>
    <form action="/calculate" method="post">
        Roll No: <input type="text" name="rollNo" required /><br><br>
        Name: <input type="text" name="name" required /><br><br>
```

```
Marks: <input type="text" name="marks" required /><br><br>
Subject: <input type="text" name="subject" required /><br><br>
<button type="submit">Calculate Percentage</button>
</form>
<p th:text="{error}" style="color: red;"></p>
</body>
</html>
```

pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>3.1.5</version>
    <relativePath /> <!-- lookup parent from repository -->
  </parent>
  <groupId>SpringBootException1</groupId>
  <artifactId>SpringBootException1</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <name>SpringBootException1</name>
  <description>Exception handling</description>
  <properties>
    <java.version>17</java.version>
  </properties>
  <dependencies>
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-web</artifactId>
    </dependency>
```

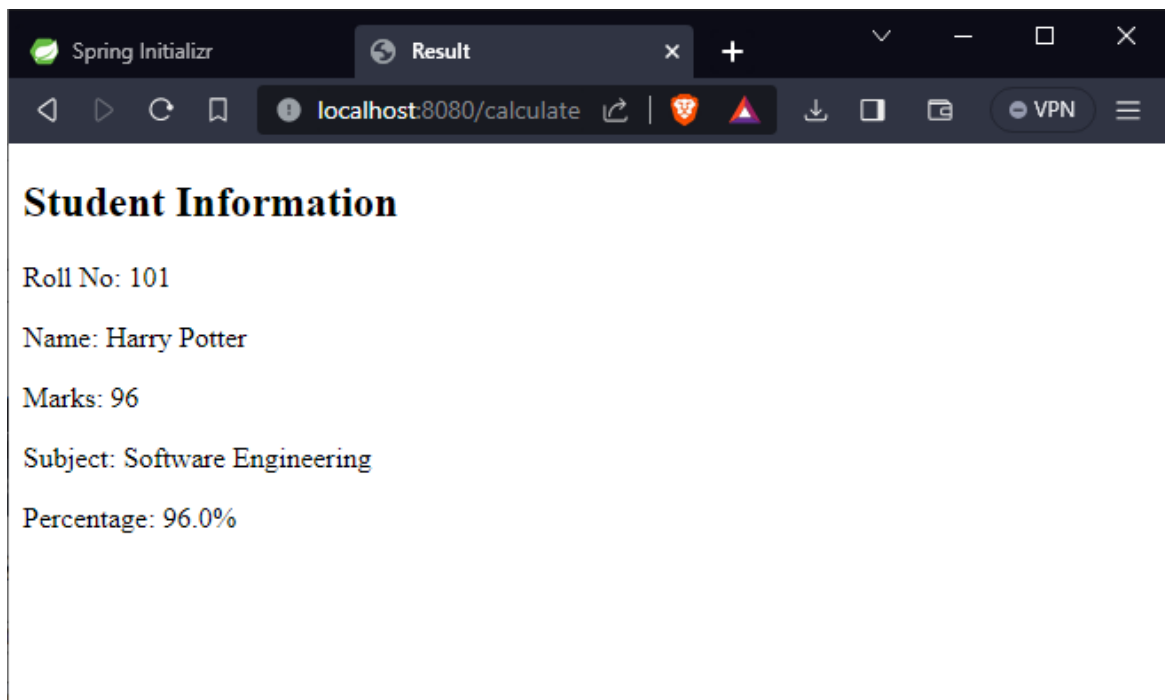
```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-test</artifactId>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-thymeleaf</artifactId>
</dependency>
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-devtools</artifactId>
</dependency>
</dependencies>
<build>
  <plugins>
    <plugin>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-maven-plugin</artifactId>
    </plugin>
  </plugins>
</build>
</project>
```

Output:

A screenshot of a web browser window. The address bar shows 'localhost:8080/calculate'. The page title is 'Calculate Percentage'. The form contains the following fields and values:

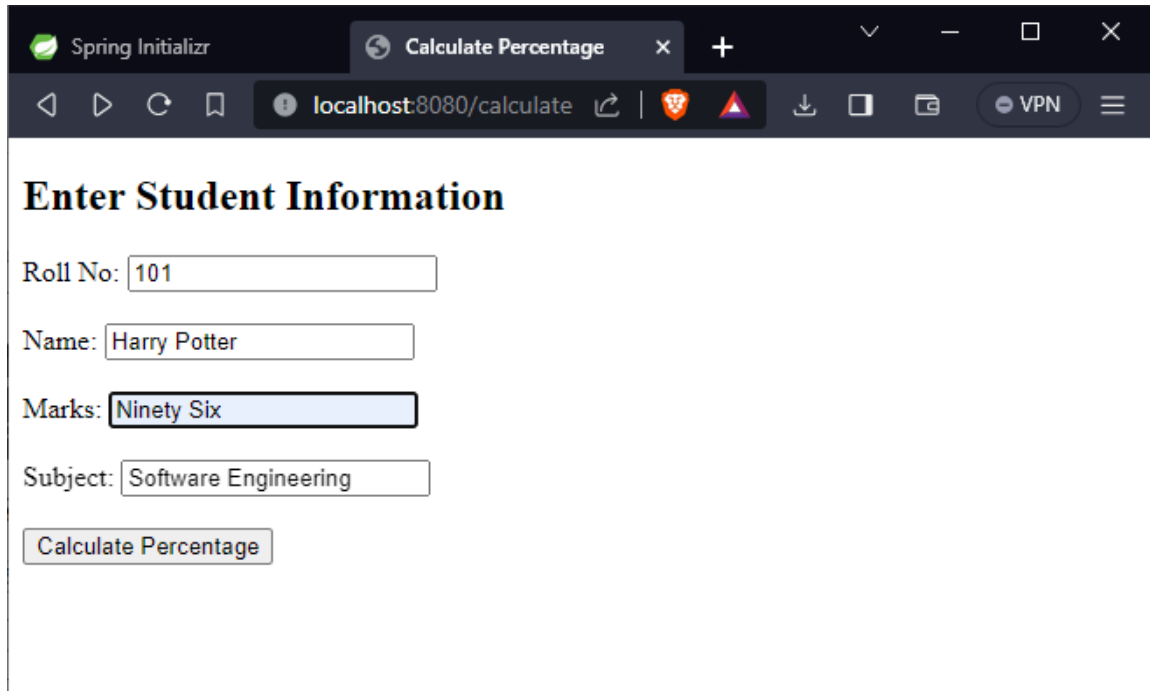
Field	Value
Roll No:	101
Name:	Harry Potter
Marks:	96
Subject:	Software Engineering

Below the fields is a button labeled 'Calculate Percentage'.



A screenshot of a web browser window. The address bar shows 'localhost:8080/calculate'. The page title is 'Result'. The page displays the student information and the calculated percentage:

Field	Value
Roll No:	101
Name:	Harry Potter
Marks:	96
Subject:	Software Engineering
Percentage:	96.0%



Spring Initializr Calculate Percentage x +

localhost:8080/calculate

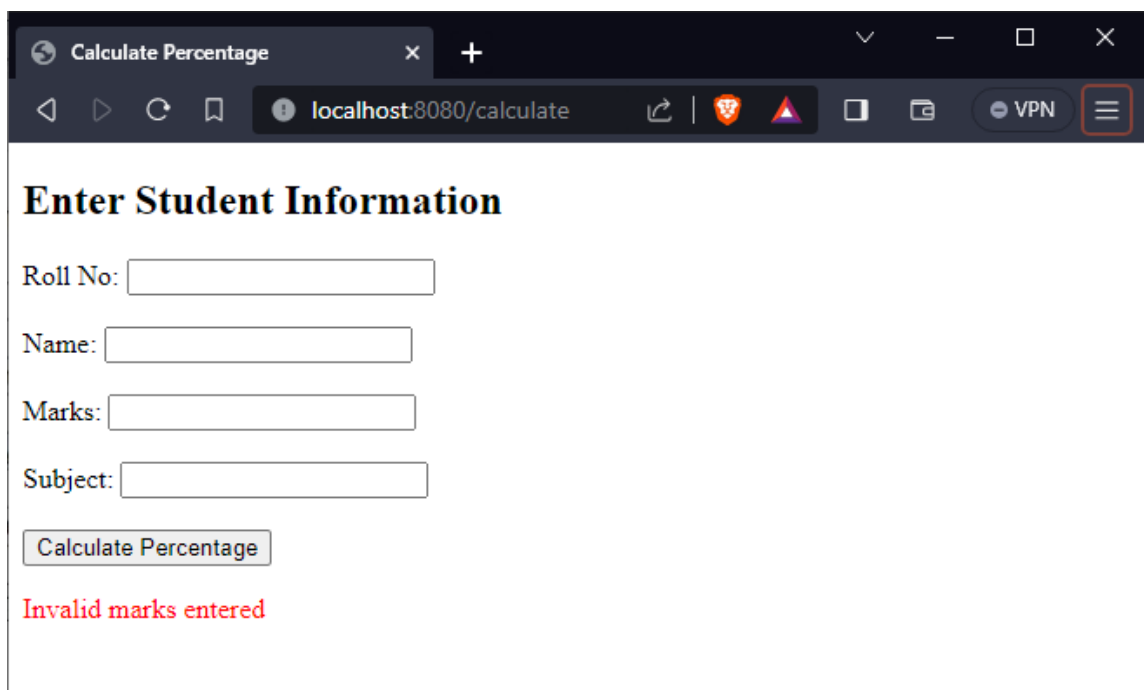
Enter Student Information

Roll No:

Name:

Marks:

Subject:



Calculate Percentage x +

localhost:8080/calculate

Enter Student Information

Roll No:

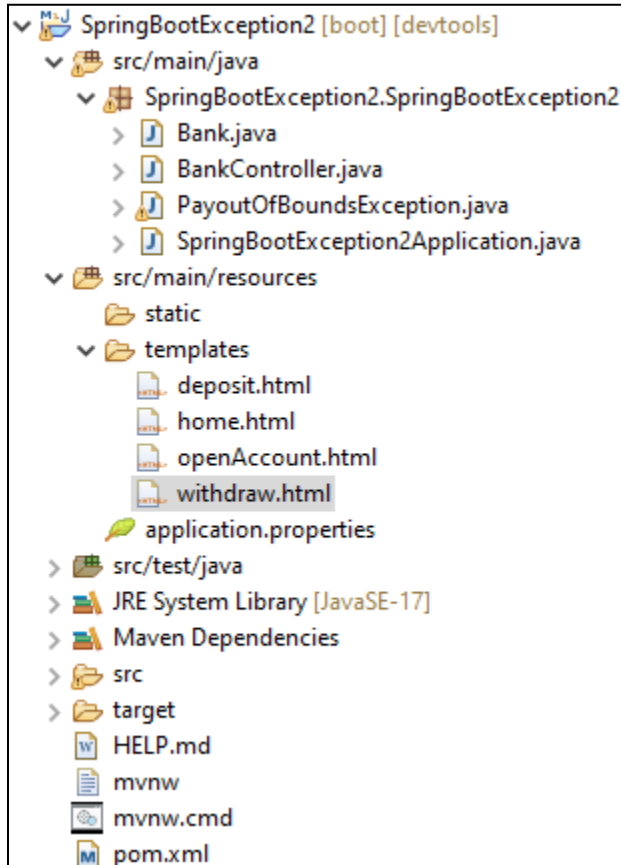
Name:

Marks:

Subject:

Invalid marks entered

2. To implement a program to create user defined exceptions. Create class Bank and define methods open (), deposit() and withdraw() with minimum balance 500. Create an exception Payout of bounds and fire except.

File Structure:**Code:****Bank.java**

```
package SpringBootException2.SpringBootException2;

public class Bank {
    private int amount;
    final int min = 500;
    String name;

    public Bank(int amt, String name) {
        this.amount = amt;
        this.name = name;
    }
}
```

```
        public void Withdraw(int amt) throws PayoutOfBoundsException {
            if (amount - min < amt)
                throw new PayoutOfBoundsException("Withdrawal not allowed.
Minimum balance limit is 500.");
            else
                amount -= amt;
        }

        public void deposit(int amt) {
            this.amount += amt;
        }

        public int getAmount() {
            return amount;
        }

        public String getName() {
            return name;
        }
    }
```

BankController.java

```
package SpringBootException2.SpringBootException2;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestParam;
```

@Controller

```
public class BankController {
    Bank bank;
```

```
@GetMapping("/openAccount")
public String openAccount() {
    return "openAccount";
}
```

```
@PostMapping("/home")
public String StartAccount(@RequestParam int amount, @RequestParam String
name, Model model) {
    bank = new Bank(amount, name);
    model.addAttribute("amount", amount);
    model.addAttribute("name", name);
    return "home";
}
```

```
@GetMapping("home")
public String updatedHome(Model model) {
    model.addAttribute("amount", bank.getAmount());
    model.addAttribute("name", bank.getName());
    return "home";
}
```

```
@PostMapping("/deposit")
public String DepositMoney(@RequestParam int amount, Model model) {
    bank.deposit(amount);
    model.addAttribute("message", "Deposit successful.");
    return "deposit";
}
```

```
@PostMapping("/withdraw")
public String WithdrawMoney(@RequestParam int amount, Model model) {
    try {
        bank.Withdraw(amount);
        model.addAttribute("message", "Withdraw successful.");
    } catch (Exception e) {
```

```
        model.addAttribute("error", e.getMessage());
    }
    return "withdraw";
}

@GetMapping("/deposit")
public String depositForm() {
    return "deposit";
}

@GetMapping("/withdraw")
public String withdrawForm() {
    return "withdraw";
}
}
```

PayoutOfBoundsException.java

```
package SpringBootException2.SpringBootException2;

public class PayoutOfBoundsException extends RuntimeException {
    public PayoutOfBoundsException(String message) {
        super(message);
    }
}
```

SpringBootException2Application.java

```
package SpringBootException2.SpringBootException2;

import org.springframework.boot.SpringApplication;

@SpringBootApplication
public class SpringBootException2Application {
    public static void main(String[] args) {
        SpringApplication.run(SpringBootException2Application.class, args);
    }
}
```

deposit.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Deposit</title>
</head>
<body>
    <h1>Deposit Money</h1>
    <form method="post" action="/deposit">
        <input type="text" name="amount" placeholder="Amount">
        <button type="submit">Deposit</button>
    </form>
    <p th:text="{message}" style="color: green;"></p>
    <br>
    <a href="/home">Home</a>
</body>
</html>
```

home.html

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
    <title>Welcome to SBI</title>
</head>
<body>
    <h2>Welcome to SBI</h2>
    <p>Account Holder : <span th:text="{name}"></span></p>
    <p>Account Balance : <span th:text="{amount}"></span></p>
    <a href="/withdraw">Withdraw</a>
    <br>
    <a href="/deposit">Deposit</a>
</body>
</html>
```

openAccount.html

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
    <title>Open Bank Account</title>
</head>
<body>
    <h2>Open Bank Account</h2>
    <form action="/home" method="post">
        Enter Name : <input type="text" required name="name" /> <br><br>
        Enter Amount : <input type="text" required name="amount" /> <br><br>
        <button type="submit">Open Account</button>
    </form>
</body>
</html>
```

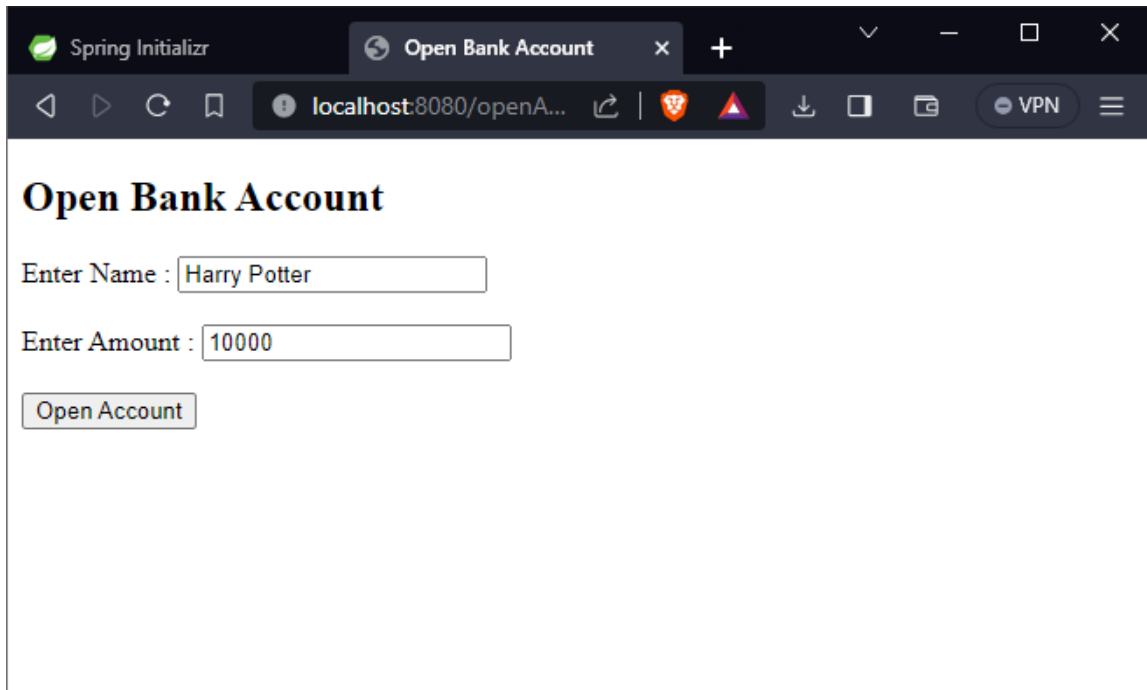
withdraw.html

```
<html>
<head>
    <title>Withdraw</title>
</head>
<body>
    <h1>Withdraw Money</h1>
    <form method="post" action="/withdraw">
        <input type="text" name="amount" placeholder="Amount">
        <button type="submit">Withdraw</button>
    </form>
    <p th:text="${error}" style="color: red;"></p>
    <p th:text="${message}" style="color: green;"></p>
    <br>
    <a href="/home">Home</a>
</body>
</html>
```

pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
https://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <parent>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-parent</artifactId>
    <version>3.1.5</version>
    <relativePath /> <!-- lookup parent from repository -->
  </parent>
  <groupId>SpringBootException2</groupId>
  <artifactId>SpringBootException2</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <name>SpringBootException2</name>
  <description>Demo project for Spring Boot</description>
  <properties>
    <java.version>17</java.version>
  </properties>
  <dependencies>
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-thymeleaf</artifactId>
    </dependency>
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-starter-web</artifactId>
    </dependency>
    <dependency>
      <groupId>org.springframework.boot</groupId>
      <artifactId>spring-boot-devtools</artifactId>
      <scope>runtime</scope>
```

```
        <optional>true</optional>
      </dependency>
      <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-test</artifactId>
        <scope>test</scope>
      </dependency>
    </dependencies>
    <build>
      <plugins>
        <plugin>
          <groupId>org.springframework.boot</groupId>
          <artifactId>spring-boot-maven-plugin</artifactId>
        </plugin>
      </plugins>
    </build>
  </project>
```

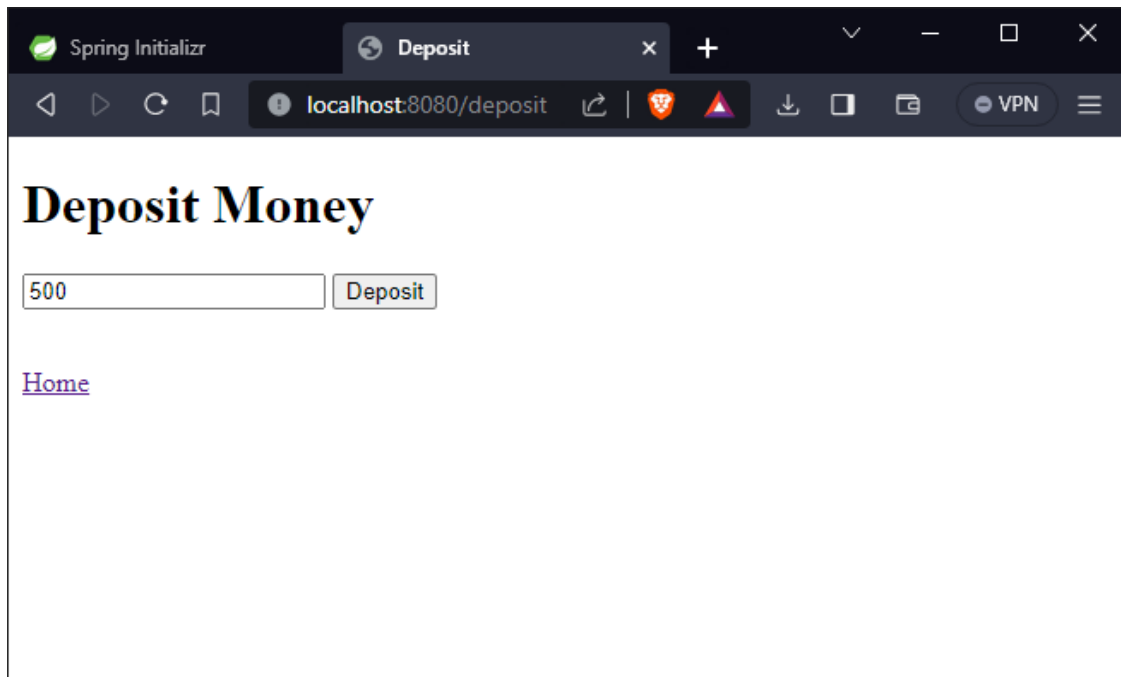
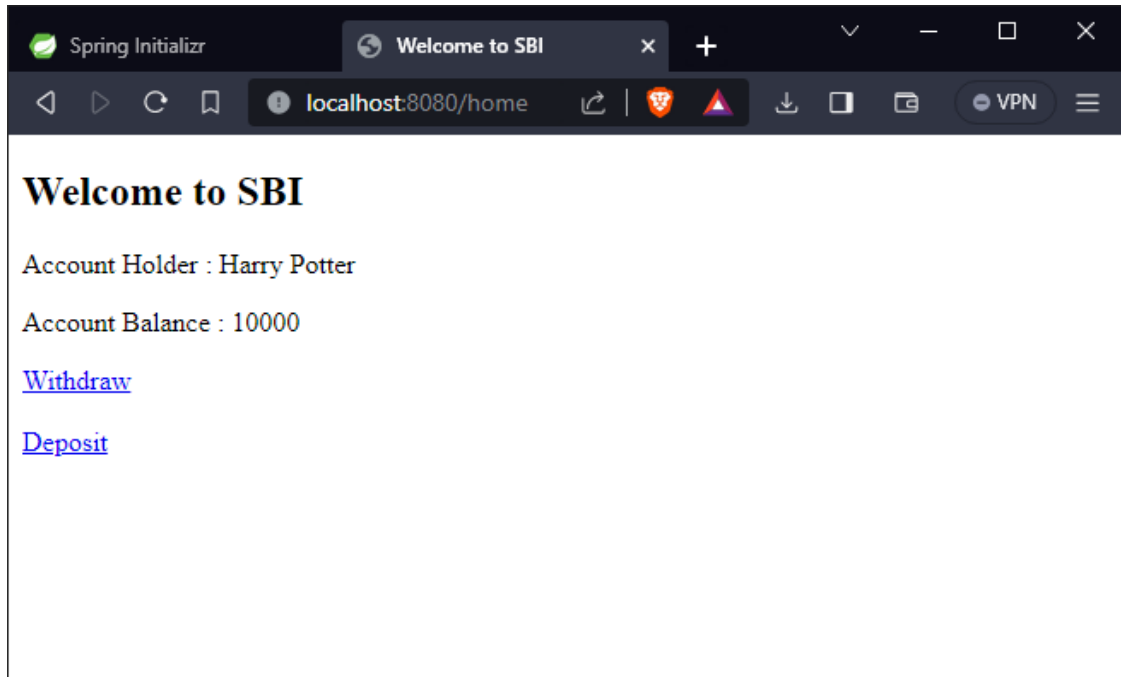
Output:

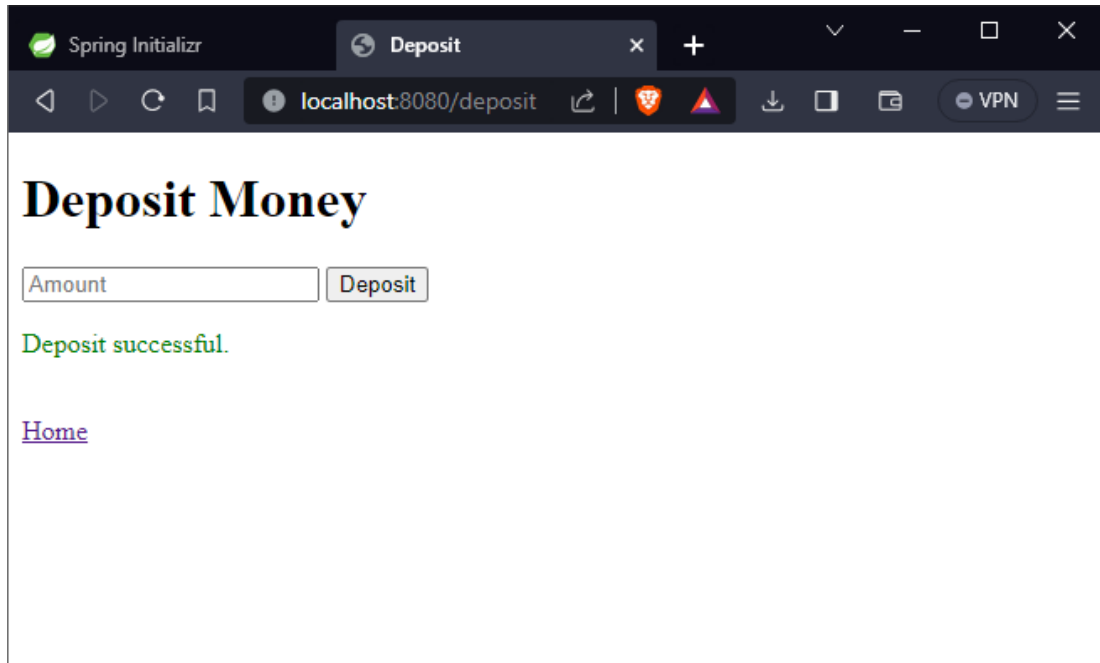
The screenshot shows a web browser window with two tabs: 'Spring Initializr' and 'Open Bank Account'. The address bar shows 'localhost:8080/openA...'. The page content includes a title 'Open Bank Account', a form with two input fields, and a button.

Open Bank Account

Enter Name :

Enter Amount :





Spring Initializr

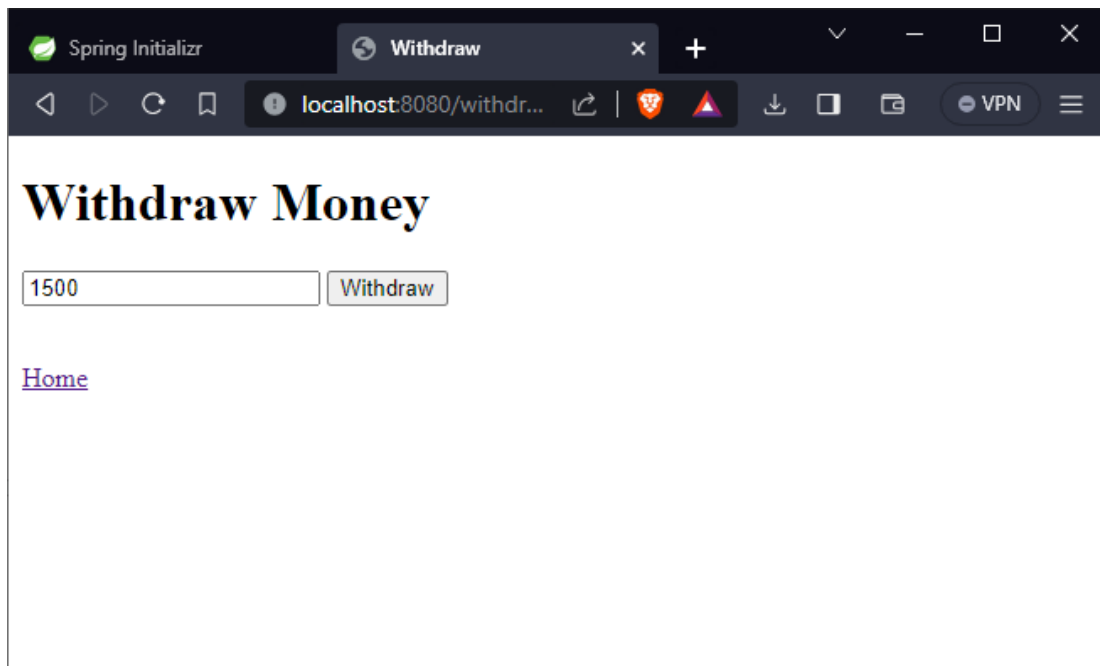
Deposit

localhost:8080/deposit

Deposit Money

Deposit successful.

[Home](#)



Spring Initializr

Withdraw

localhost:8080/withdr...

Withdraw Money

[Home](#)

