

แบบฝึกปฏิบัติ ครั้งที่ 4

เรื่อง	การสร้างส่วนติดต่อประสานผู้ใช้งาน และการสื่อสารระหว่าง Service
วัตถุประสงค์	<ol style="list-style-type: none"> 1. เพื่อฝึกฝนการสร้างส่วนติดต่อประสานผู้ใช้งานด้วย Vaadin 2. เพื่อฝึกฝนการสื่อสารแบบ REST 3. เพื่อฝึกฝนการสื่อสารระหว่าง Service ด้วย Webclient

ข้อที่ 1 ให้นักศึกษาพัฒนาโปรแกรมเครื่องคิดเลข

วิดีโอตัวอย่างการทำงานของโปรแกรม

<https://youtu.be/JchXauRFtbA>

รายละเอียดในการพัฒนา

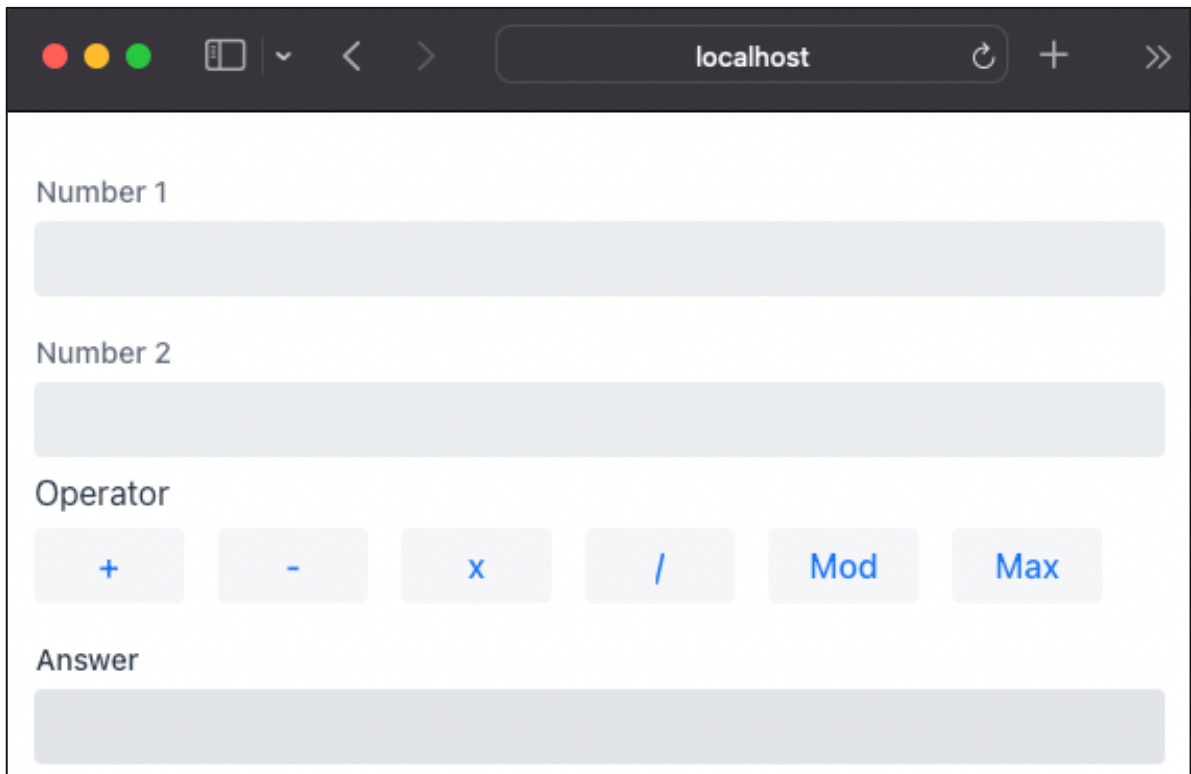
1. ให้นักศึกษาสร้างคลาสเป็น MathAPI เป็น Service ที่มีการสื่อสารแบบ REST โดยมีรายละเอียดดังนี้
 - 1.1. ให้นักศึกษาสร้างเมธอด myPlus() ที่รับค่าเลขทศนิยมเข้ามาสองจำนวน จากนั้น ให้คืนค่าผลลัพธ์ของการบวกเลขสองจำนวนดังกล่าว โดยกำหนดรูปแบบการสื่อสารแบบ GET และมี URL ในการเรียกใช้ เป็น `http://127.0.0.1:8080/plus/{n1}/{n2}`
 - 1.2. ให้นักศึกษาสร้างเมธอด myMinus() ที่รับค่าเลขทศนิยมเข้ามาสองจำนวน จากนั้น ให้คืนค่าผลลัพธ์ของการลบเลขสองจำนวนดังกล่าว โดยกำหนดรูปแบบการสื่อสารแบบ GET และมี URL ในการเรียกใช้ เป็น `http://127.0.0.1:8080/minus/{n1}/{n2}`
 - 1.3. ให้นักศึกษาสร้างเมธอด myDivide() ที่รับค่าเลขทศนิยมเข้ามาสองจำนวน จากนั้น ให้คืนค่าผลลัพธ์ของการหารเลขสองจำนวนดังกล่าว โดยกำหนดรูปแบบการสื่อสารแบบ GET และมี URL ในการเรียกใช้ เป็น `http://127.0.0.1:8080/divide/{n1}/{n2}`
 - 1.4. ให้นักศึกษาสร้างเมธอด myMulti() ที่รับค่าเลขทศนิยมเข้ามาสองจำนวน จากนั้น ให้คืนค่าผลลัพธ์ของการคูณเลขสองจำนวนดังกล่าว โดยกำหนดรูปแบบการสื่อสารแบบ GET และมี URL ในการเรียกใช้ เป็น `http://127.0.0.1:8080/multi/{n1}/{n2}`
 - 1.5. ให้นักศึกษาสร้างเมธอด myMod() ที่รับค่าเลขทศนิยมเข้ามาสองจำนวน จากนั้น ให้คืนค่าผลลัพธ์ของการหารเอาเศษของเลขสองจำนวนดังกล่าว โดยกำหนดรูปแบบการสื่อสารแบบ GET และมี URL ในการเรียกใช้ เป็น `http://127.0.0.1:8080/mod/{n1}/{n2}`

1.6. ให้นักศึกษาร่างเมธอด myMax() ที่รับค่าเลขทศนิยมเข้ามาสองจำนวน จากนั้น ให้คืนค่าผลลัพธ์จำนวนที่มีค่าสูงที่สุดออกมา โดยกำหนดรูปแบบการสื่อสารแบบ **POST** และมี URL ในการเรียกใช้เป็น <http://127.0.0.1:8080/max>

2. ให้นักศึกษาร่างส่วนติดต่อผู้ใช้งาน โดยกำหนดชื่อคลาสเป็น MathView

2.1. ให้นักศึกษากำหนด URL ในการเรียกใช้เป็น <http://127.0.0.1:8080/index1>

2.2. ให้นักศึกษาร่างส่วนติดต่อผู้ใช้งานดังภาพด้วย Vaadin



2.3. ให้นักศึกษาเขียนโปรแกรมเพื่อจัดการเหตุการณ์เมื่อผู้ใช้งานกดปุ่มต่อไปนี้

2.3.1. เมื่อผู้ใช้งานกดปุ่ม + โปรแกรมจะเรียกใช้เมธอด myPlus() จาก Service ผ่าน WebClient พร้อมกับส่งค่าในช่อง Number1 และ Number2 ไป จากนั้น จะนำผลลัพธ์ที่ Service ตอบกลับมาแสดงผลบริเวณช่อง Answer

2.3.2. เมื่อผู้ใช้งานกดปุ่ม - โปรแกรมจะเรียกใช้เมธอด myMinus() จาก Service ผ่าน WebClient พร้อมกับส่งค่าในช่อง Number1 และ Number2 ไป จากนั้น จะนำผลลัพธ์ที่ Service ตอบกลับมาแสดงผลบริเวณช่อง Answer

2.3.3. เมื่อผู้ใช้งานกดปุ่ม x โปรแกรมจะเรียกใช้เมธอด myMulti() จาก Service ผ่าน WebClient พร้อมกับส่งค่าในช่อง Number1 และ Number2 ไป จากนั้น จะนำผลลัพธ์ที่ Service ตอบกลับมาแสดงผลบริเวณช่อง Answer

- 2.3.4.เมื่อผู้ใช้งานกดปุ่ม / โปรแกรมจะเรียกใช้เมธอด myDivide() จาก Service ผ่าน WebClient พร้อมกับส่งค่าในช่อง Number1 และ Number2 ไป จากนั้น จะนำผลลัพธ์ที่ Service ตอบกลับมาแสดงผลบริเวณช่อง Answer
- 2.3.5.เมื่อผู้ใช้งานกดปุ่ม Mod โปรแกรมจะเรียกใช้เมธอด myMod() จาก Service ผ่าน WebClient พร้อมกับส่งค่าในช่อง Number1 และ Number2 ไป จากนั้น จะนำผลลัพธ์ที่ Service ตอบกลับมาแสดงผลบริเวณช่อง Answer
- 2.3.6.เมื่อผู้ใช้งานกดปุ่ม Max โปรแกรมจะเรียกใช้เมธอด myMax() จาก Service ผ่าน WebClient พร้อมกับส่งค่าในช่อง Number1 และ Number2 ไป จากนั้น จะนำผลลัพธ์ที่ Service ตอบกลับมาแสดงผลบริเวณช่อง Answer

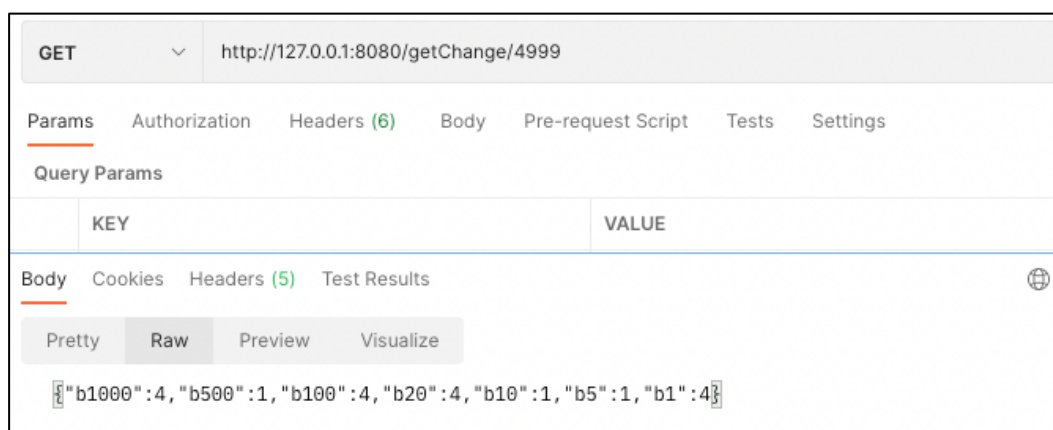
ข้อที่ 2 ให้นักศึกษาพัฒนาโปรแกรมทอนเงิน

รายละเอียดในการพัฒนา

1. ให้นักศึกษาสร้างคลาส Change ตามข้อกำหนดต่อไปนี้
 - 1.1. กำหนดให้ implements Serializable และ @Component ให้กับคลาส Change
 - 1.2. ให้ประกาศแอตทริบิวต์ b1000, b500, b100, b20, b10, b5, และ b1 เป็นชนิด int

ชื่อแอตทริบิวต์	ความหมาย
b1000	จำนวนแบงค์พัน
b500	จำนวนแบงค์ห้าร้อย
b100	จำนวนแบงค์ร้อย
b20	จำนวนแบงค์ยี่สิบ
b10	จำนวนเหรียญสิบ
b5	จำนวนเหรียญห้า
b1	จำนวนเหรียญบาท

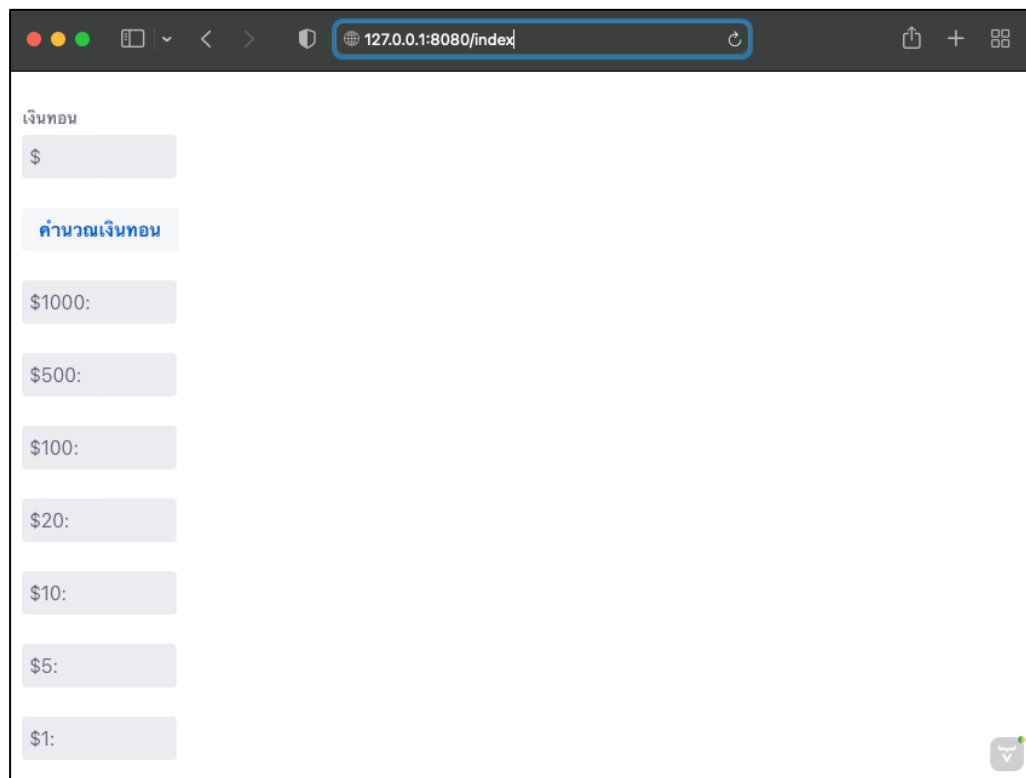
- 1.3. ให้สร้างเมธอด setter และ getter ของแอตทริบิวต์ b1000, b500, b100, b20, b10, b5, และ b1
2. ให้นักศึกษาสร้างคลาส Cashier ตามข้อกำหนดต่อไปนี้
 - 2.1. กำหนดให้สร้างคลาส Cashier เป็น Service ที่มีการสื่อสารแบบ REST
 - 2.2. ให้นักศึกษาสร้างเมธอดชื่อ getChange() ที่รับค่าเงินทอนเข้ามาในรูปแบบเลขจำนวนเต็ม และให้คืนค่าผลลัพธ์เป็นจำนวนแบงค์ที่ต้องทอนในแต่ละแบงค์ใน Data type ชนิด Change อาทิเช่น



โดยกำหนดรูปแบบการสื่อสารแบบ **GET** และมี URL ในการเรียกใช้เป็น http://127.0.0.1:8080/getChange/[จำนวนเงินทอน]

3. ให้นักศึกษาสร้างคลาส CashierView ตามข้อกำหนดต่อไปนี้
 - 3.1. ให้นักศึกษากำหนด URL ในการเรียกใช้เป็น http://127.0.0.1:8080/index2

3.2. ให้นักศึกษาสร้างส่วนติดต่อผู้ใช้งานดังภาพด้วย Vaadin



3.3. ให้นักศึกษาเขียนโปรแกรมเพื่อจัดการเหตุการณ์เมื่อผู้ใช้กดปุ่ม “คำนวณเงินทอน” โปรแกรมจะเรียกใช้ service จาก `getChange()` ผ่าน `WebClient` พร้อมกับส่งค่าในช่อง “เงินทอน” ไป จากนั้น จะนำผลลัพธ์ที่ Service ตอบกลับมาแสดงผลบริเวณช่อง ช่องด้านล่าง ตัวอย่างดังภาพ

