

## ข้อกำหนด

\*\*\* เมื่อ {xxx} คือ เลขท้าย 3 ตัวรหัสนักศึกษาของท่าน \*\*\*

ให้นักศึกษา สร้างและตั้งชื่อ project ตามรูปแบบดังนี้ mid\_exam\_{xxx}

นักศึกษาต้องแสดง รหัสและชื่อนักศึกษา ในทุกหน้าของเว็บเพจ

นักศึกษาสามารถจัดรูปแบบการแสดงผลในแต่ละเว็บเพจได้ตามต้องการ (แต่ต้องมีข้อมูลครบตามที่โจทย์ ระบุ)

ให้นักศึกษา ใช้ design pattern MVC เท่านั้น (ต้องมี MVC ครบในทุก flows)

บรรทัดแรกของโปรแกรม ต้องมีรูปแบบ ดังนี้ // ชื่อ-สกุล (รหัสนักศึกษา)

จงเขียนเว็บแอปพลิเคชัน โดยใช้รูปแบบการทำงานแบบ MVC เพื่อทำงานตามขั้นตอนดังต่อไปนี้

## 1. การลงทะเบียน (5 คะแนน)

1.1. เมื่อผู้ใช้ร้องขอมาที่ URL `http://host:port/mid_exam_{xxx}.../{xxx}/register` โปรแกรมจะส่ง html form ให้ผู้ใช้กรอก email address และ user name (ดังรูปที่ 1)

1.2. เมื่อผู้ใช้ submit form ข้อ 1.1 กลับมา ให้โปรแกรมทำการ จัดเก็บข้อมูลของ User (email และ user name) ไว้ เพื่อนำไปแสดงใน page อื่นๆ ต่อไป จนกว่าผู้ใช้จะปิด browser

## 2. การตรวจสอบจำนวนเฉพาะ (15 คะแนน)

2.1. เมื่อผู้ใช้ร้องขอมาที่ URL `http://host:port/mid_exam_{xxx}.../{xxx}/is_prime` โปรแกรมจะส่ง html form ให้ผู้ใช้กรอก เลขจำนวนเต็ม 1 จำนวน โดยโปรแกรมต้องแสดง user name และ email address จากข้อ 1 (ดังรูปที่ 2.1)

2.2. เมื่อผู้ใช้ submit form ข้อ 2.1 กลับมา ให้โปรแกรมทำการ ตรวจสอบและประมวลผลข้อมูลดังนี้

2.2.1. ถ้าตัวเลขที่ผู้ใช้กรอกมา มีค่าน้อยกว่า 2 หรือ เป็นอักขระ หรือ ไม่ใช่เลขจำนวนเต็ม หรือ ช่องว่าง ให้แสดงผลลัพธ์ดังรูปที่ 2.2

2.2.2. ถ้าตัวเลขที่ผู้ใช้กรอกมา มีค่าเป็น 2 ขึ้นไป ให้โปรแกรมตรวจสอบว่าเลขนั้น เป็นจำนวนเฉพาะหรือไม่ แล้วแสดงผลลัพธ์ดังรูปที่ 2.3

## 3. ประวัติการตรวจสอบจำนวนเฉพาะ (5 คะแนน)

3.1. เมื่อผู้ใช้ร้องขอมาที่ URL `http://host:port/mid_exam_{xxx}.../{xxx}/history` โปรแกรมจะแสดง user name และ email address จากข้อ 1 และประวัติการตรวจสอบจำนวนเฉพาะทั้งหมด รวมทั้งสรุปจำนวน ตัวเลขที่เป็นจำนวนเฉพาะที่ผู้ใช้คนนั้นๆ กรอกตั้งแต่เริ่มลงทะเบียนในข้อ 1 โดยให้แสดงผลลัพธ์ ดังรูปที่ 3

\*\*\* การพิจารณาคะแนน จะดูจาก โปรแกรมทำงานได้ถูกต้อง และการออกแบบ MVC ที่เหมาะสม \*\*\*

\*\*\* ตรวจและให้คะแนน เฉพาะข้อที่มี ผลลัพธ์ในไฟล์ PDF และ ปฏิบัติตาม ข้อกำหนด ของข้อสอบ เท่านั้น \*\*\*

## สิ่งที่ต้องส่ง (ZIP รวมกันตามรูปแบบที่ระบุใน LEB2)

1) บันทึกหน้าจอ (CAPTURE) โปรแกรม (MVC) ของแต่ละข้อ + ผลลัพธ์ ส่ง ในรูปแบบไฟล์ PDF \*\*\*

- ผลลัพธ์ของการทดสอบตามเงื่อนไขในแต่ละข้อเพื่อแสดงให้เห็นว่า น.ศ. สามารถทำโปรแกรมได้ตามความต้องการของโจทย์
- ส่วนของโปรแกรม ที่เกี่ยวกับโจทย์ข้อนั้นๆ (MVC) เพื่อแสดงให้เห็นว่า น.ศ. เข้าใจ Flow ของโปรแกรมที่ น.ศ. เขียน

2) Export Project (ในรูปแบบ zip)

>>> การ Hard Coding เพื่อให้ได้ผลลัพธ์ โดยไม่ได้มีการประมวลผลตามที่โจทย์กำหนด ถือว่าเป็นการทุจริต <<<

Midterm Exam Web App by Pichet 6515001000

User name:

Email Address:

- ต้องแสดง รหัสและชื่อนักศึกษา ในทุกหน้าของเว็บเพจ (ออกแบบข้อความ ตามที่ นศ. เห็นควร)
- URL ต้องมี 3 ตัวท้ายของรหัส น.ศ. ตามรูป

รูปที่ 1

Midterm Exam Web App by Pichet 6515001000

Welcome Pichet Lim (pichet@gmail.com)

Primer Checker ::

Enter number to check:

User name / Email address ที่ลงทะเบียนจากข้อ 1

รูปที่ 2.1

<p>Primer Checker ::</p> <p>Enter number to check: <input type="text" value="-2"/></p> <p>-2 is invalid input !!!</p> <p><input type="button" value="Submit"/></p>	<p>Primer Checker ::</p> <p>Enter number to check: <input type="text" value="23.12"/></p> <p>23.12 is invalid input !!!</p> <p><input type="button" value="Submit"/></p>
<p>Primer Checker ::</p> <p>Enter number to check: <input type="text" value="2X5"/></p> <p>2X5 is invalid input !!!</p> <p><input type="button" value="Submit"/></p>	<p>Primer Checker ::</p> <p>Enter number to check: <input type="text"/></p> <p>Empty is invalid input !!!</p> <p><input type="button" value="Submit"/></p>

รูปที่ 2.2 (เมื่อผู้ใช้ป้อนข้อมูลที่ มีค่าน้อยกว่า 2 หรือ เป็นอักขระ หรือ ไม่ใช่เลขจำนวนเต็ม หรือ ช่องว่าง)

Primer Checker ::		
Enter number to check:	<input type="text" value="3989"/>	<input type="button" value="Submit"/>
3989 is prime number		

Primer Checker ::		
Enter number to check:	<input type="text" value="9001"/>	<input type="button" value="Submit"/>
9001 is prime number		

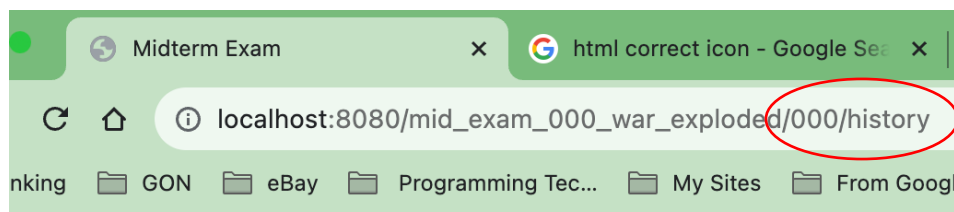
Primer Checker ::		
Enter number to check:	<input type="text" value="9007"/>	<input type="button" value="Submit"/>
9007 is prime number		

Primer Checker ::		
Enter number to check:	<input type="text" value="7123"/>	<input type="button" value="Submit"/>
7123 is not prime number		

Primer Checker ::		
Enter number to check:	<input type="text" value="10001"/>	<input type="button" value="Submit"/>
10001 is not prime number		

Primer Checker ::		
Enter number to check:	<input type="text" value="100033"/>	<input type="button" value="Submit"/>
100033 is not prime number		

รูปที่ 2.3



### Prime Checker History ::

Midterm Exam Web App by Pichet 6515001000

Welcome Pichet Lim (pichet@gmail.com)

No Number Is Prime ?

1	3989	✓
2	9001	✓
3	9007	✓
4	7123	X
5	10001	X
6	100033	X
7	9007	✓

ผู้ใช้สามารถป้อนเลขซ้ำได้

No of prime = 4

รูปที่ 3

ตัวอย่าง รูปแบบ เอกสาร PDF ที่ต้องส่ง

### ข้อ 1) การลงทะเบียน

Midterm Exam Web App by Pichet 6515001000

User name:

Email Address:

ผลลัพธ์ต้อง แสดง URL ของ browser ทุกครั้ง

```
9  @Getter
10 @Setter
11 @Entity
12 @Table(name = "employees")
13 public class Employee {
14     @Id
15     private Integer employeeNumber;
16     private String firstName;
17     private String lastName;
18     private String extension;
19     private String email;
20 }
```

```
12
13 @WebServlet(name = "OfficeListServlet", value = "/office-list")
14 public class OfficeListServlet extends HttpServlet {
15     @PersistenceContext(unitName = "classic-models", type = PersistenceContextType.EXTENDED)
16     private EntityManager entityManager ;
17     @Override
18     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException {
19         System.out.println("Entity Manager: " + entityManager);
20         OfficeRepository repository = new OfficeRepository();
21         request.setAttribute("offices", repository.findAll());
22         request.getRequestDispatcher("/office_list.jsp").forward(request, response);
23     }
24 }
```

```
9  <%@ page contentType="text/html; charset=UTF-8" language="java" %>
10 <html>
11 <body>
12 <div class="container-fluid m-4 w-75">
13     <c:forEach items="${offices}" var="office">
14         <div class="row">
15             <div class="col-1">${office.officeCode}</div>
16             <div class="col-2">${office.city}</div>
17             <div class="col-1">${office.country}</div>
18             <div class="col-2">${office.postalCode}</div>
19             <div class="col-2">${office.phone}</div>
20         </div>
21     </c:forEach>
22 </div>
23 </body>
24 </html>
```