



ฝ่ายพัฒนาระบบสารสนเทศ
องค์การบริหารสโมสรนิสิตจุฬาลงกรณ์มหาวิทยาลัย (อบจ.)

แบบทดสอบทักษะพื้นฐาน สำหรับผู้สมัครตำแหน่ง

Frontend Developer

ฝ่ายพัฒนาระบบสารสนเทศ

องค์การบริหารสโมสรนิสิตจุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2564

คำชี้แจง

1. แบบทดสอบนี้ถูกออกแบบมาให้ทำเสร็จภายใน **72 ชั่วโมง** อย่างไรก็ตามคณะกรรมการฯ จะให้เวลาในการทำทั้งหมด **120 ชั่วโมง** หลังจากที่ผู้สมัครได้รับบททดสอบนี้แล้ว เนื่องจากผู้สมัครบางท่านอาจจะมีช่วงเวลาที่สะดวกแตกต่างกันไป
2. ให้ผู้สมัครทำการ fork repository [isd-sgcu/sgcu64-frontend-recruitment](https://github.com/isd-sgcu/sgcu64-frontend-recruitment) โดยใช้ account ของผู้สมัครเอง
3. กรุณาอ่านรายละเอียดโจทย์และทำความเข้าใจให้เรียบร้อยก่อนทำโจทย์ หากมีข้อสงสัยประการใดผู้สมัครสามารถส่งอีเมลไปที่ isd.sgcu64@gmail.com เพื่อสอบถามได้ตลอดเวลา
4. พึงระลึกไว้เสมอว่าสิ่งเหล่านี้มีความสำคัญในการพัฒนาซอฟต์แวร์จริง ๆ ซึ่งมีผลต่อการพิจารณาผู้สมัคร ดังนี้
 - Code Readability – โค้ดอ่านรู้เรื่อง กระชับ เข้าใจง่าย
 - Commit Message Readability – ข้อความใน commit ต้องมีความหมาย
5. หากมีไอเดียเพิ่มเติมจากที่โจทย์ระบุไว้ที่ดีกว่าจากที่โจทย์กำหนด ผู้สมัครสามารถระบุไอเดียนั้นลงไปในส่วนของคำตอบ โดยเขียนด้วยรูปแบบดังนี้
Additional Idea: <ไอเดียเพิ่มเติมของผู้สมัคร>
6. หากมีสมมติฐานเพิ่มเติมที่โจทย์ไม่ได้ให้ข้อมูลไว้ ผู้สมัครสามารถระบุสมมติฐานนั้นลงไปในส่วนของคำตอบ โดยเขียนด้วยรูปแบบดังนี้
Assumption: <สมมติฐานของผู้สมัคร>
7. การส่ง assignment ขอให้ผู้สมัครอัปโหลด source code ทั้งหมดลง repository ดังกล่าว โดยแยกออกเป็น folder ละ 1 ข้อ เช่น “Problem #1” โดยสามารถเพิ่มไฟล์ README.md ไว้ในแต่ละข้อได้ หากต้องการอธิบายเกี่ยวกับโค้ดที่ส่งมาเพิ่มเติม
8. การตรวจ assignment จะยึดจาก **commit สุดท้ายใน branch master** ก่อนเวลาสิ้นสุดแบบทดสอบที่ระบุไว้ในอีเมลที่ผู้สมัครได้รับเท่านั้น โปรดพึงระลึกไว้เสมอว่า การตรงต่อเวลา เป็นคุณสมบัติที่สำคัญอย่างหนึ่งต่อการเป็นวิศวกรซอฟต์แวร์ที่ดี
9. ขอให้ทยอยส่ง source code เมื่อทำเสร็จแต่ละข้อ เพื่อที่จะได้ไม่มีปัญหาการส่งเมื่อเวลาใกล้หมด **ไม่จำเป็นต้องเสร็จทุกข้อแล้วค่อยส่ง**
10. ผู้สมัครสามารถค้นหาข้อมูลเพิ่มเติมจากช่องทางใดก็ได้ ในระหว่างการทำแบบทดสอบ
11. การตัดสินใจของทีมคัดเลือกถือเป็นที่สุด

โจทย์ข้อที่ 1 Seven Segments

จุดประสงค์

- เพื่อทดสอบความรู้ความเข้าใจในการเขียนโปรแกรมเบื้องต้น
- เพื่อประเมินความสามารถในการออกแบบวิธีการแก้ปัญหาอย่างง่าย

รายละเอียดโจทย์

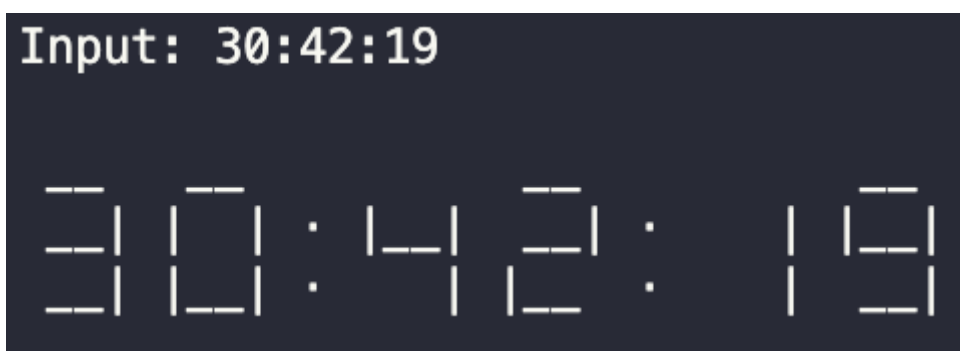
เมื่อโลกถูกปกคลุมไปด้วยความมืดมิด ท้าทายสารทศมีเพียงความหวาดผวและความกลัว โลกจึงต้องการผู้กอบกู้เพื่อเป็นแสงสว่างแห่งอนาคต จึงได้มีการจัดกิจกรรมโอลิมปิกผู้กอบกู้ เพื่อเฟ้นหายอดมนุษย์ไปจัดการกับกองทัพจอมมารผู้ชั่วร้าย ซึ่งสุดท้ายแล้วชาวโลกก็สามารถหาผู้กอบกู้ได้ทั้งหมด 7 คนด้วยกัน ซึ่งผู้กอบกู้ทั้ง 7 ก็ได้ช่วยกันฝ่าฝืนอุปสรรคจนมาถึงหน้าปราสาทของจอมมารได้แล้ว

เนื่องจากจอมมารเป็นคนที่มีความละเอียดรอบคอบจึงได้วางแผนสำรองหากตัวเองไม่สามารถปราบผู้กอบกู้ได้ นั่นก็คือจอมมารจะติดตั้งเครื่องวัดชีพจรไว้ที่ตัวเอง และหากชีพจรของจอมมารหยุดนิ่งเมื่อไร "ระเบิดเวลา" จะเริ่มนับถอยหลังแล้วแยกผืนแผ่นดินออกเป็น 7 ส่วนออกจากกันเพื่อไม่ให้ผู้กอบกู้กลับบ้านเกิดของตนเองได้ แต่ทั้งนี้ทั้งนั้น "จอนับเวลาลอยหลัง" ของจอมมารที่ชื่อมาจาก Shopee มีระบบแสดงผลเป็นแบบ "7 ส่วน" [จอมมารชื่อมาจากลิงก์นี้](#)

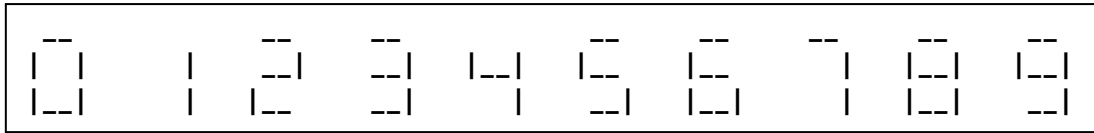
โดยในปราสาทจอมมารมีเพียงแค่คุณคนเดียวที่สามารถเขียนระบบแสดงผล 7 ส่วน ได้ ดังนั้นจอมมารจึงมอบหมายหน้าที่อันยิ่งใหญ่ให้กับคุณ ในการเขียนระบบแสดงผล 7 ส่วนให้เสร็จสมบูรณ์ให้ได้

Features ที่ต้องมี

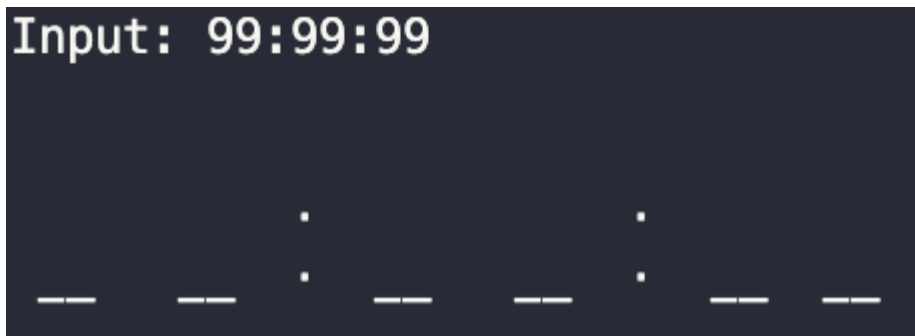
1. สามารถรับ input ใน format xx:xx:xx ได้ เมื่อ x คือเลขจำนวนโดดฐานสิบ โดยแสดงผล
2. สามารถแสดงผล output เป็นรูปแบบ seven segment โดย output จะประกอบไปด้วยตัวอักษร _ หรือ | หรือ · เท่านั้น เช่น



3. ตัวอย่างการแสดงผลแต่ละตัวเลข



4. สามารถบอกได้ว่า input ที่ส่งมาเป็นเวลาที่ถูกต้องหรือไม่ โดยเวลาในหลักนาที่และวินาทีอยู่ได้แค่ในช่วง 0 – 59 แต่หลักชั่วโมงจะเป็นเท่าไรก็ได้ หาก input ไม่ถูกต้องให้แสดงผลเป็น __:__:__ ดังนี้



5. **ไม่จำเป็น** ต้องให้เวลานับถอยหลังได้จริง ๆ แต่ถ้าอยากโชว์ของก็จัดมา
6. ผู้สมัครสามารถเขียนโปรแกรมโดยใช้ภาษาใดก็ได้ โดยขอให้ระบุวิธีการรันโปรแกรกดังกล่าวไว้ใน README.md ของ folder “Problem #1”
- หากเป็นภาษาที่สามารถ compile ออกมาเป็นไฟล์ executable ได้ (เช่น C/C++, etc.) ขอให้อัปไฟล์ executable นั้นขึ้นมาด้วย
 - หากเป็นภาษาที่ต้องใช้ interpreter ในการทำงาน (เช่น Python, Node.js, etc.) ขอให้ระบุ command ที่ใช้ในการรันโปรแกรมด้วย เช่น “node index.js”
7. ผู้สมัครจะเขียนอธิบายอะไรเพิ่มใน README.md หรือ comment ในไฟล์ก็ได้ตามอัธยาศัย

Tips

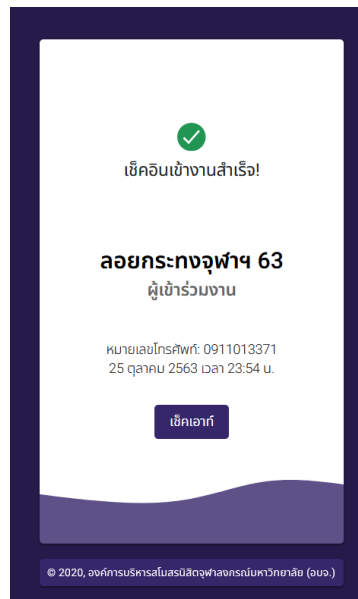
1. **สำหรับโจทย์ข้อนี้** ผู้สมัครไม่จำเป็นต้องกังวลเรื่อง Time and Memory Complexity (เวลาและ memory ที่ใช้) ขอเพียงให้โปรแกรมทำงานถูกต้องก็พอ

โจทย์ข้อที่ 2 Chula Chana

จุดประสงค์

- เพื่อทดสอบความรู้ความเข้าใจในการเขียนโปรแกรมเบื้องต้น
- เพื่อทดสอบความสามารถในการจัดการข้อมูลที่ได้รับมา ตั้งแต่การจับเก็บไปจนถึงการค้นข้อมูล ได้อย่างถูกต้องและแม่นยำ

รายละเอียดโจทย์



(ภาพหน้าเว็บ CU Check In)

CU Check In เป็น Web Application ที่ทาง ISD SGC63 ทำขึ้นเพื่อทำการ track จำนวนคนในพื้นที่ ในช่วงอีเวนต์ลอยกระทงจุฬาลงกรณ์ ในปีที่ผ่านมา ซึ่งใน app ผู้ใช้จะต้องสแกน QR Code ที่หน้างานลอยกระทง และกรอกเบอร์โทรศัพท์เพื่อ check in แล้วจึงจะสามารถเข้างานได้ และสามารถสแกน QR Code เพื่อ check out เมื่อจะออกจากงาน

ใน 1 ปีถัดมา ตัวคุณเองพบว่า Web Application ดังกล่าวของ ISD SGC63 มี User Experience ที่ใช้งานยาก จึงได้ทำการเขียนโปรแกรม Chula Chana (จุฬา ชนะ) เพื่อแข่งขันกับ CU Check In และ ไทยชนะ เพื่อที่จะก้าวขึ้นเป็นที่หนึ่งของ Tracking Application ของประเทศไทยให้ได้เลย

```
Welcome to Chula Chana!!!
Available commands:
  1. Check in user
  2. Check out user
  3. Print people count
Please input any number: 1
-----

Check In
Enter phone number: 0855555555
  1. Mahamakut Building
  2. Sara Phra Keaw
  3. CU Sport Complex
  4. Sanam Juub
  5. Samyan Mitr Town
Select the place: 3
Checking in 0855555555 into CU Sport Complex
-----

Welcome to Chula Chana!!!
Available commands:
  1. Check in user
  2. Check out user
  3. Print people count
Please input any number: 3
-----

Current Population
  1. Mahamakut Building: 20
  2. Sara Phra Keaw: 9
  3. CU Sport Complex: 35
  4. Sanam Juub: 2
  5. Samyan Mitr Town: 17
-----

Welcome to Chula Chana!!!
Available commands:
  1. Check in user
  2. Check out user
  3. Print people count
Please input any number:
```

(ตัวอย่าง UI อันสวยงามของโปรแกรมจุฬาฯ ชนะ)

โดยจุฬาฯ ชนะ ที่คุณจะสร้างขึ้นนั้น เป็นโปรแกรมง่าย ๆ บนคอมพิวเตอร์ ที่สามารถรับข้อมูลเป็นเบอร์โทรศัพท์ เพื่อ check in หรือ check out ในพื้นที่ไหนก็ได้ที่ตั้งค่าไว้ พร้อมทั้งสามารถดูจำนวนคนที่อยู่ในพื้นที่ต่างๆ ได้ และแสดงผลออกมาทาง standard output ดังเช่นตามในรูปข้างต้น

ทั้งนี้ รูปแบบการรับข้อมูลเข้าหรือการแสดงผลนั้น ผู้สมัครสามารถออกแบบเป็นอย่างไรก็ได้ตามความเหมาะสม ไม่มีความจำเป็นจะต้องเขียนเหมือนแบบในตัวอย่างข้างต้น เพียงแต่จะมี feature บางประการที่จำเป็นจะต้องมีอยู่ในโปรแกรมของคุณ ดังนี้

Features ที่ต้องมี

1. Check In

- Input มีสองอย่าง คือ เบอร์โทรศัพท์ (สมมติว่าผู้ใช้กรอกถูก Format ได้เลย) พื้นที่ที่ต้องการ check in (สามารถ hard code ลงไปใน source code ได้)

- ไม่จำเป็นต้องมี output แต่อาจมีได้ เพื่อการใช้งานที่ง่ายขึ้น
- ในกรณีที่เบอร์โทรศัพท์นั้น เคย check in อยู่ในพื้นที่ที่ check in ไว้อยู่แล้ว แล้วจะต้องทราบได้ว่าเป็นคนเดียวกัน และไม่ส่งผลกระทบต่อจำนวนคนในพื้นที่นั้น
- ในกรณีที่เบอร์โทรศัพท์นั้น เคย check in อยู่ในพื้นที่อื่น ไม่ใช่พื้นที่ที่กำลังจะ check in ขอให้เข้าใจว่าผู้ใช้ลืม checkout พื้นที่เดิม จะต้องนำเบอร์โทรศัพท์นั้น ออกจากพื้นที่เก่า แล้วนำมาใส่พื้นที่ใหม่ (แสดงว่าจำนวนคนในพื้นที่เก่าจะต้องลดลง และจำนวนคนในพื้นที่ใหม่จะต้องมากขึ้น)
- ในสองกรณีข้างต้น สามารถแจ้งเตือนผู้ใช้ให้รู้ตัวหรือไม่ก็ได้ แล้วแต่ออกแบบ

2. Check Out

- Input มีอย่างเดียว คือ เบอร์โทรศัพท์
- ไม่จำเป็นต้องมีข้อมูลส่งออก แต่อาจมีได้ เพื่อการใช้งานที่ง่ายขึ้น
- การ check out จะต้องตรวจสอบว่า เบอร์โทรศัพท์นั้นๆ อยู่ในพื้นที่ใด แล้วจึง check out ออกจากพื้นที่นั้นให้โดยอัตโนมัติ (แสดงว่าจำนวนคนในพื้นที่ที่ผู้ใช้เคยอยู่ต้องลดลง)
- ในกรณีที่เบอร์โทรศัพท์นั้นไม่ได้ถูก check in อยู่พื้นที่ใด ไม่จำเป็นต้องทำอะไรใด ๆ สามารถแจ้งเตือนผู้ใช้ให้รู้ตัวหรือไม่ก็ได้ แล้วแต่ออกแบบ

3. แสดงผลจำนวนคนในแต่ละพื้นที่

- ไม่มี input เพิ่มเติม
- Output จะต้องแสดงจำนวนคนในแต่ละพื้นที่ใน format ที่เข้าใจง่าย

สำหรับจำนวนสถานที่ที่ให้ผู้ใช้งานใช้นั้น สามารถ hard code ลงไปในโปรแกรมได้เลย เพียงแต่ต้อง ≥ 5 สถานที่ หรือสามารถทำเป็น dynamic (เพิ่มลดได้) ซึ่งคือมี feature สำหรับให้ผู้ใช้งานสามารถเพิ่มสถานที่เองได้

Feature อื่น ๆ นอกจากที่กล่าวมาข้างต้น เป็นสิ่งที่ไม่ได้บังคับ สามารถทำเพิ่มได้ หากอยากใช้ของ

โจทย์ข้อที่ 3 มหาภัยพิบัติวันสิ้นโลก

จุดประสงค์

- เพื่อทดสอบความรู้ HTML, CSS และ JavaScript เบื้องต้น
- เพื่อประเมินความสามารถในการแก้ปัญหาของผู้สมัคร
- เพื่อประเมินความคิดสร้างสรรค์ของผู้สมัคร

รายละเอียดโจทย์

วันหนึ่ง ดาวเคราะห์ดวงที่สามของระบบสุริยะทางช้างเผือกหรืออีกชื่อหนึ่ง "โลก" ได้ล่มสลายไปจากมหาภัยพิบัติทางธรรมชาติที่ใช้เวลาก่อร่างสร้างตัวกว่า 2.745 ล้านปีจนทำให้สิ่งมีชีวิตเกือบทั้งหมดในโลกมลายหายไปไปในที่สุด คุณและเหล่าผองเพื่อนเป็นผู้รอดจากมหาภัยพิบัตินั้น ซึ่งคุณต้องการที่จะกอบกู้โลกใบเดิมให้กลับมาให้เร็วที่สุด แต่ทั้งนี้ทั้งนั้นคุณได้พบว่าโลกใบเดิมของคุณมีระบบจัดการฐานข้อมูลที่ไม่ดีเท่าที่ควร ระบบราชการและสวัสดิการของบางประเทศมีความจำเป็นที่จะต้องกรอกข้อมูลซ้ำ ๆ ทั้ง ๆ ที่มนุษย์สมัยก่อนทุกคนมีสิ่งที่เอาไว้ยืนยันตัวตนเช่น บัตรประชาชน ที่ข้อมูลพื้นฐานทุกอย่างเช่น ชื่อ นามสกุล อยู่ในนั้นทั้งหมด

คุณผู้เป็นผู้รอดชีวิตจากภัยพิบัติและเห็นถึงปัญหานั้นจึงได้เริ่มทำ "ระบบลงทะเบียน" ที่จะให้มนุษย์ที่เหลือรอดทุกคนลงทะเบียนและมอบสวัสดิการที่ดีให้อย่างทั่วถึง ซึ่งคุณได้รวบรวมเหล่าเพื่อนพ้องที่ต้องการจะแก้ปัญหานี้มาได้จำนวนหนึ่งแล้ว ซึ่งเพื่อน ๆ ของคุณก็ทำหน้าที่ต่าง ๆ กันไป โดยหน้าที่ของคุณคือ "พัฒนาเว็บลงทะเบียน" ที่จะมอบประสบการณ์ในการใช้งานที่ดีให้กับผู้ใช้ มีความสวยงาม รองรับได้ทั้งโทรศัพท์และบนคอมพิวเตอร์ และทำงานได้อย่างถูกต้องตามที่ควรจะเป็น โดยคุณทำร่วมกับเพื่อนอีกคนหนึ่งแต่เพื่อนคนนั้นดันลาป่วยยาว ทำให้งานหยุดอยู่ที่กลางทาง คุณจึงมีหน้าที่สานต่องานที่ค้างไว้ให้เสร็จสมบูรณ์ ซึ่งเพื่อนของคุณได้ออกแบบ [UX/UI มาให้คุณแล้วเรียบร้อยแล้ว](#) คุณสามารถเข้าไปดูและพัฒนาเว็บให้มีหน้าตาตามนั้นได้เลย

แต่บางครั้งงานออกแบบที่เพื่อนคุณทำนั้นก็ไม่ได้สมบูรณ์แบบเสมอไป คุณสามารถปรับแต่งอะไรได้นิดหน่อยตามความเหมาะสมตามที่คุณเห็นสมควร แต่อย่าลืมที่จะไปบอกเพื่อนของคุณด้วยว่าคุณแก้อะไรไปและบอกเหตุผลด้วยว่าทำไม (ในที่นี้คืออย่าลืมเขียน Additional Idea ด้วยยย)

Features ที่ต้องมี

1. เว็บไซต์สามารถใช้งานได้จริง
2. เว็บไซต์มีหน้าตาเหมือนกับ UX/UI ที่กำหนดไว้ (หากเพิ่มเติม feature อื่น ขอให้ใกล้เคียง UX/UI ที่กำหนดไว้ให้มากที่สุด)
3. เว็บไซต์มีระบบ Form Validation เช่น email มี @ ไหม, รหัสผ่านกับยืนยันรหัสผ่านตรงกันหรือไม่ เป็นต้น
4. เว็บไซต์สามารถ Responsive ได้ (เว็บใน Mobile และ Desktop ต้องไม่ละ)
5. เว็บไซต์สามารถ Navigate ผู้ใช้ไปยัง Social Media อื่น ๆ ได้เช่น Facebook, Instagram และ Twitter

Tips

1. เราจะมี "โครง" ของเว็บที่เขียนด้วย HTML, CSS และ JavaScript ไว้ให้แล้วซึ่งคุณมีหน้าที่เดิมคือเพิ่มอีกนิดหน่อยเพื่อให้เว็บลงทะเบียนสมบูรณ์ โดยคุณอาจเลือกหรือไม่เลือกเขียนต่อจากโครงที่มีไว้ก็ได้ โครงของเว็บจะอยู่ใน repository [isd-sgcu/sgcu64-frontend-recruitment](https://github.com/isd-sgcu/sgcu64-frontend-recruitment)
2. คุณสามารถค้นหาข้อมูลเพิ่มเติมจากแหล่งความรู้ไหนก็ได้ไม่จำกัด ยกเว้นถามเพื่อน ๆ ทั้งนี้ เราคาดหวังว่าคุณจะสามารถแก้ไขปัญหาได้ด้วยตัวเอง
3. หากคุณใช้ VS Code ในการพัฒนา แนะนำว่าติดตั้ง [VS Code Extension: Live Server](#) เพื่อความง่ายในการพัฒนาเว็บไซต์
 - ก่อนอื่นทำการ Install Extension มาก่อนตามลิงก์ที่แปะไว้ข้างบนเลย
 - เมื่อ Install เสร็จแล้ว ให้คลิกขวาที่ไฟล์ `index.html` แล้วจะมี Menu หนึ่งมาว่า **Open with Live Server**
 - ซึ่ง Live Server จะรันอยู่ที่ <http://localhost:5500> ถ้าเปิดแล้วได้หน้าต่างคล้าย ๆ folder ไม่ต้องตกใจให้คุณกดเลือก Problem #3 หน้าเว็บก็จะแสดงออกมา หรือสามารถเข้าได้จาก <http://localhost:5500/Problem%20%233/>
 - ทุกครั้งที่คุณ save file, Live Server จะทำการ Rerender หน้าเว็บใหม่ให้ ดังนั้นคุณไม่ได้ต้องเปิด ๆ ปิด ๆ ระหว่างทำงานอีกต่อไป
4. สำหรับ `normalize.css` คุณไม่จำเป็นต้องเข้าใจว่ามันคืออะไร แค่ใช้เพียงแค่ `index.css` ก็เพียงพอแล้ว หากมี class ใดที่มีทั้งใน `index.css` และ `normalize.css` ตัวเว็บจะเลือก `index.css` มาแสดงผลก่อน
5. ก่อนที่เพื่อนของคุณจะลงงาน เขาได้ทำการ comment บอกคุณไว้แล้วว่าต้อง แก่ หรือ เพิ่ม ตรงไหนบ้าง แต่ถ้าคุณรู้สึกว่าคุณมีทำอื่นอยู่ในใจ ก็สามารถใช้ทำของคุณได้เลยแน่นอนว่าสามารถแก้ไขจุดอื่น ๆ นอกเหนือจากใน comment ได้เลย เพียงแต่ขอให้หน้าต่างเว็บสุดท้ายออกมาเหมือน design ก็พอ
6. คุณไม่จำเป็นต้องใช้ framework ใด ๆ ในการทำข้อนี้ แต่หากต้องการใช้ก็สามารถใช้ได้
7. หากคุณอยากเพิ่มอะไรมากกว่าที่กำหนดก็สามารถทำได้เลย!

การส่งงาน

ขอให้อัปโหลด source code **ทั้งหมด** (ยกเว้น dependencies ที่มีขนาดใหญ่ เช่น `node_modules` ถ้ามี) หากไม่ได้เขียนด้วย pure HTML, CSS, JavaScript ขอให้ระบุวิธีการเปิดเว็บขึ้นมาดูด้วย