

## ใบงานการทดลองที่ 10

### เรื่อง การควบคุมเวอร์ชันการทำงานผ่านโปรแกรม Eclipse

#### 1. จุดประสงค์ทั่วไป

- 1.1. รู้และเข้าใจการติดต่อกับผู้ใช้งาน และการหลายงานพร้อมกัน
- 1.2. รู้และเข้าใจการติดต่อระหว่างงาน

#### 2. เครื่องมือและอุปกรณ์

เครื่องคอมพิวเตอร์ 1 เครื่อง ที่ติดตั้งโปรแกรม Eclipse

#### 3. ทฤษฎีการทดลอง

- 3.1. Version Control System (VCS) คืออะไร? มีประโยชน์อย่างไร?

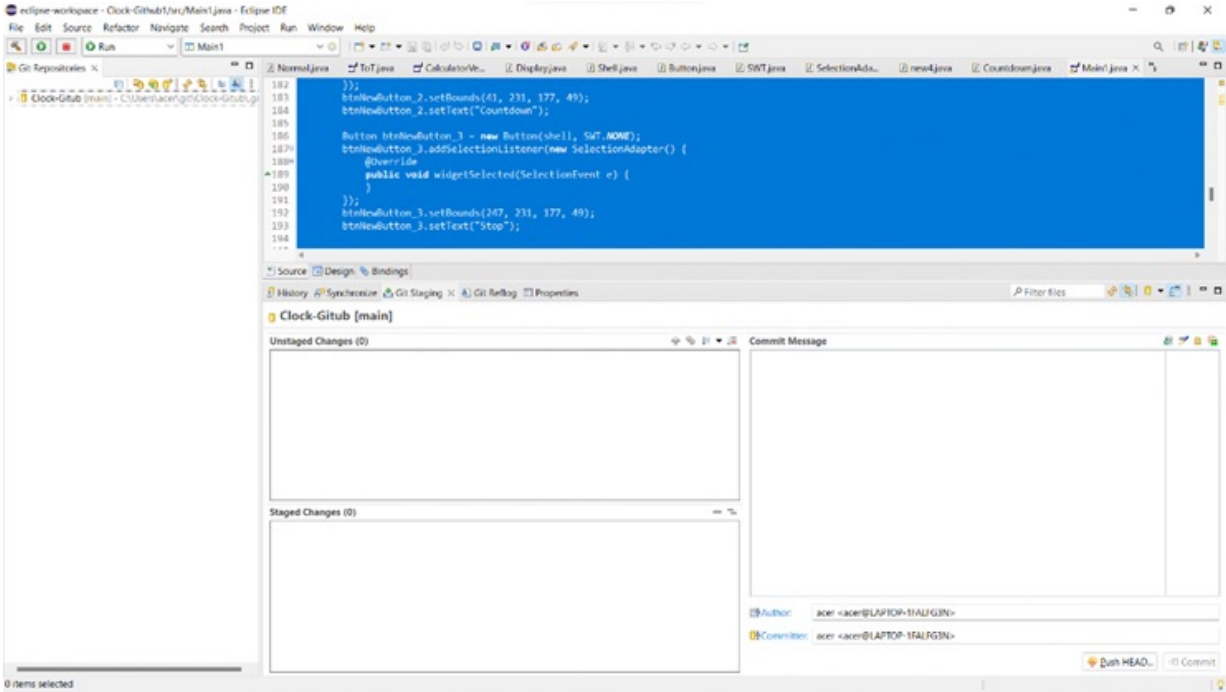
คือ คือระบบซอฟต์แวร์ที่จะคอยบันทึกเวอร์ชันการเปลี่ยนแปลงของโค้ดหรือเอกสารต่างๆ โดยจะทำการบันทึกไว้ด้วยว่าการเปลี่ยนแปลงแต่ละครั้งนั้นทำเพื่ออะไร และทำโดยใครประโยชน์ ช่วยให้สามารถย้อนไฟล์บางไฟล์หรือแม้กระทั่ง โปรเจคกลับไปเป็นเวอร์ชันเก่าได้ นอกจากนั้นระบบ VCS ยังจะช่วยให้เปรียบเทียบการแก้ไขที่เกิดขึ้นในอดีต ดูว่าใครเป็นคนแก้ไขคนสุดท้ายที่อาจทำให้เกิดปัญหา แก้ไขเมื่อไร

- 3.2. Git ต่างกับ Github อย่างไร?

Git เป็นระบบที่ช่วยจัดการการแก้ไข ใน Repository ส่วน Github เป็นบริการจัดเก็บ Repository ออนไลน์พร้อมกับพีเจอรอำนวยความสะดวกต่าง ๆ ที่ให้เราไปทำงานร่วมกันคนอื่นได้

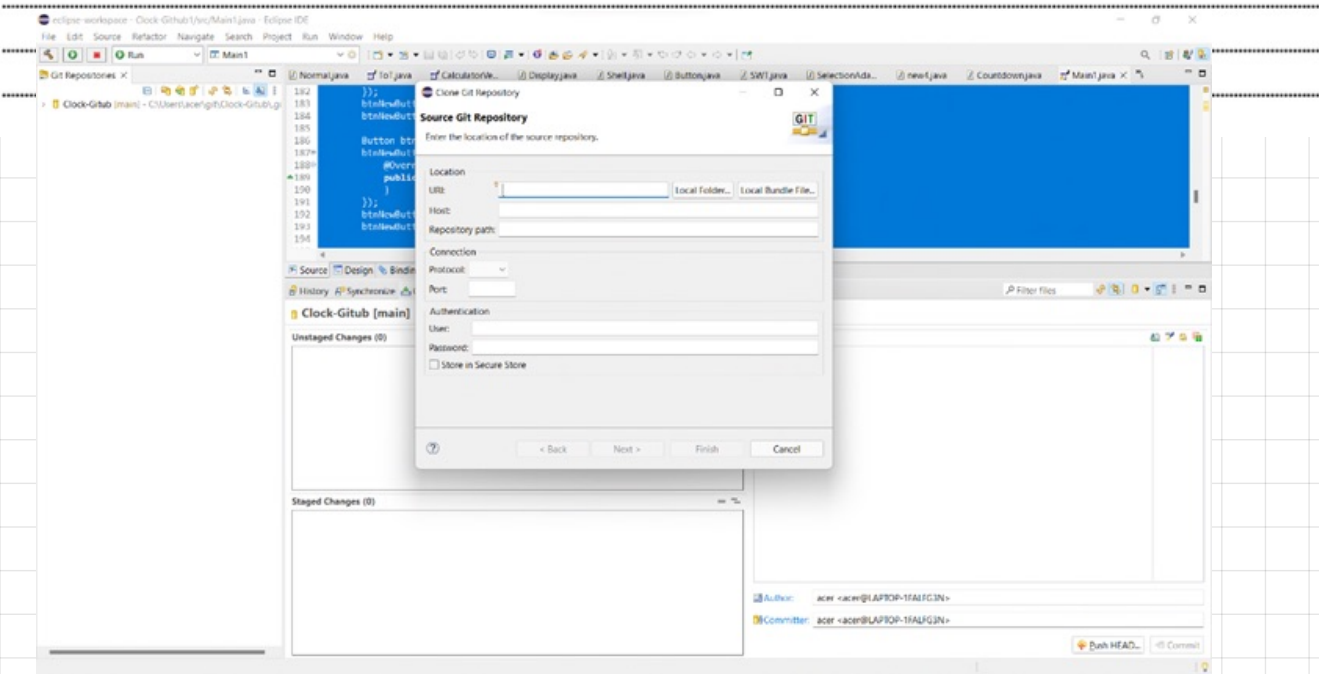
- 3.3. Repository คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

คือการเก็บสำรองข้อมูลและการเปลี่ยนแปลงของ Source Code ทำให้สามารถย้อนกลับไปทีเวอร์ชันใดๆ ก่อนหน้า และดูรายละเอียดการเปลี่ยนแปลงของแต่ละเวอร์ชันได้ นอกจากนั้นยังสามารถดูได้ว่าใครเป็นคนแก้ไข



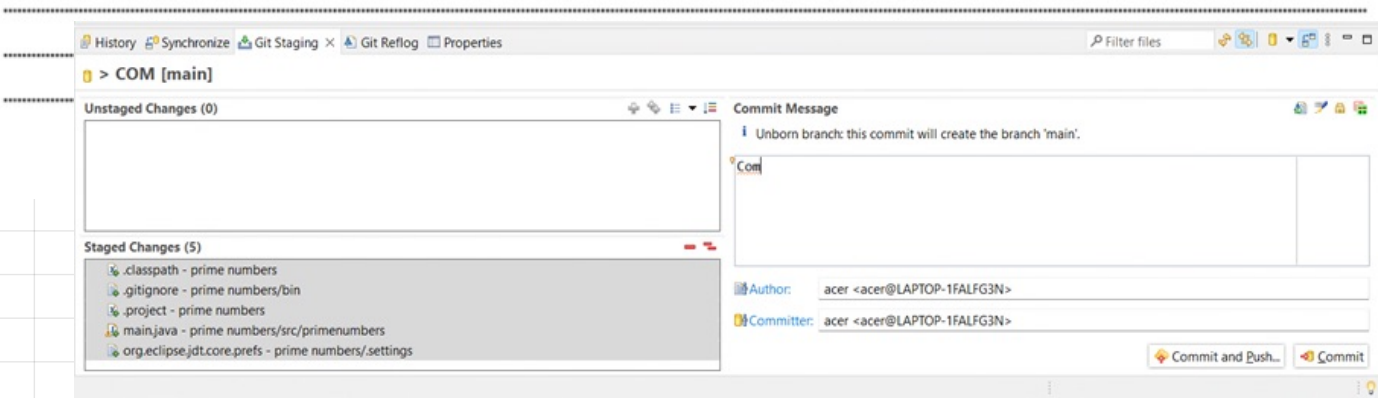
3.4. Clone คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

เวลาที่ผู้อ่านมี Repository อยู่บน Remote ซักแห่งอยู่แล้ว และต้องการ Sync มาลงเครื่องของเรา เราจะต้องทำสิ่งที่เรียกว่า Clone Repository หรือก็คือการก๊อปปี้ Repository จาก Remote



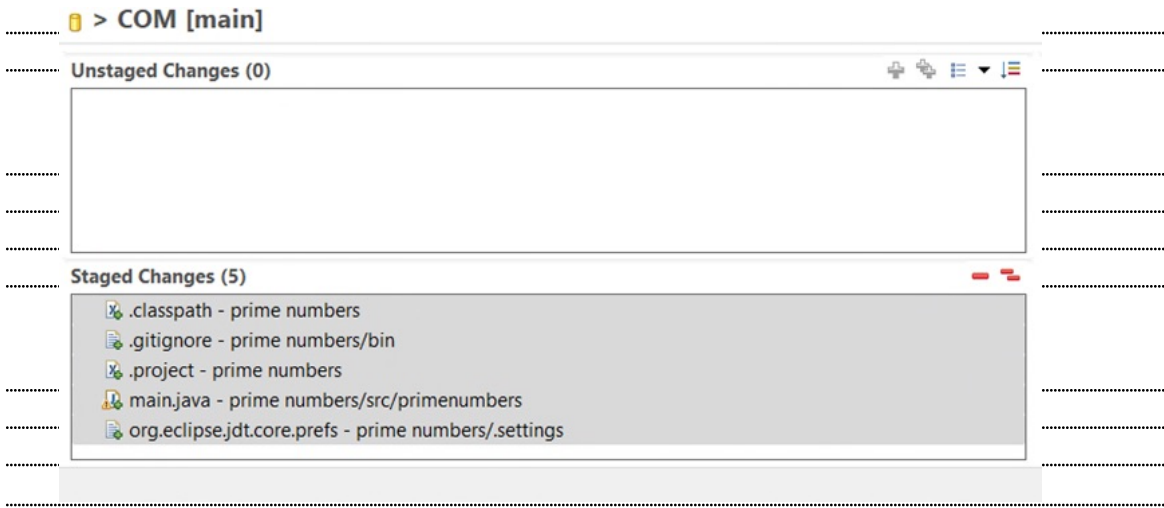
3.5. Commit คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

เวลาที่มีข้อมูลที่แก้ไขเสร็จแล้ว (โค้ดที่เขียนคำสั่งบางอย่างเสร็จแล้ว) แล้วอยากจะทำการ Backup เก็บไว้ใน VCS จะเรียกกันว่า Commit



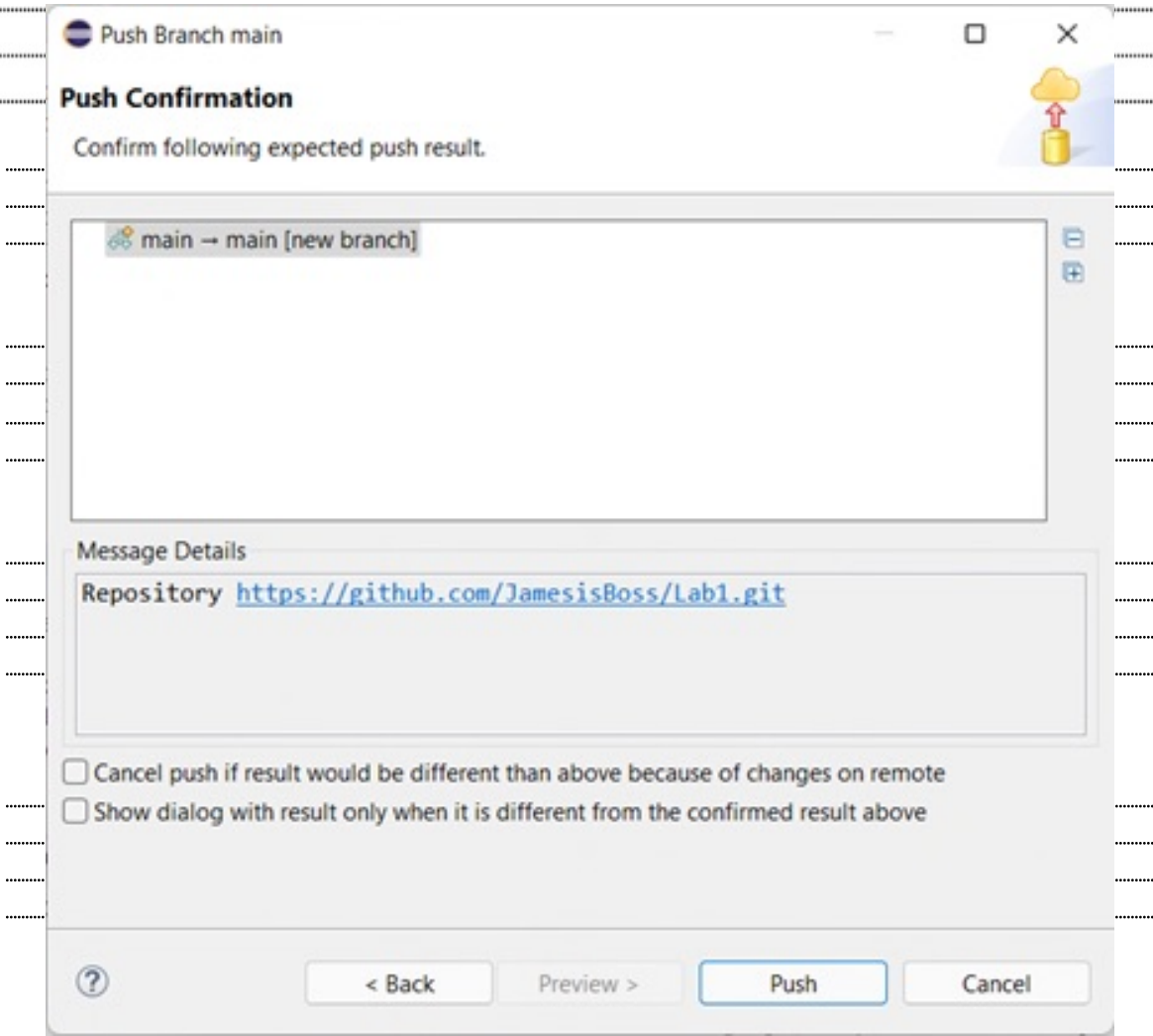
3.6. Staged และ Unstaged คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

เวลาเราแก้ไขโค้ดหรือแก้ไขข้อมูล ไฟล์ที่ถูกแก้ไขจะอยู่ในสถานะ Unstaged และเวลาที่เราทำอะไรเสร็จเรียบร้อยแล้ว แล้วอยากจะ Commit เก็บไว้ จะต้องเลือกไฟล์ที่ต้องการเพื่อย้ายเข้าสู่สถานะ Staged ก่อนถึงจะทำการ Commit ได้



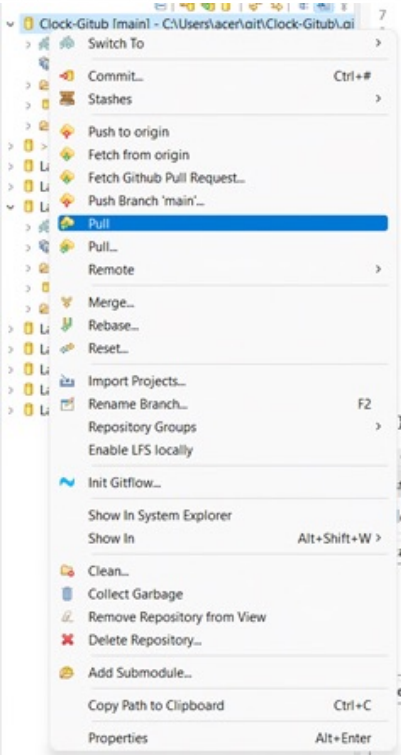
3.7. Push คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

Push คือการนำโค้ดหรือไฟล์เข้าตัวระบบ Git Repository



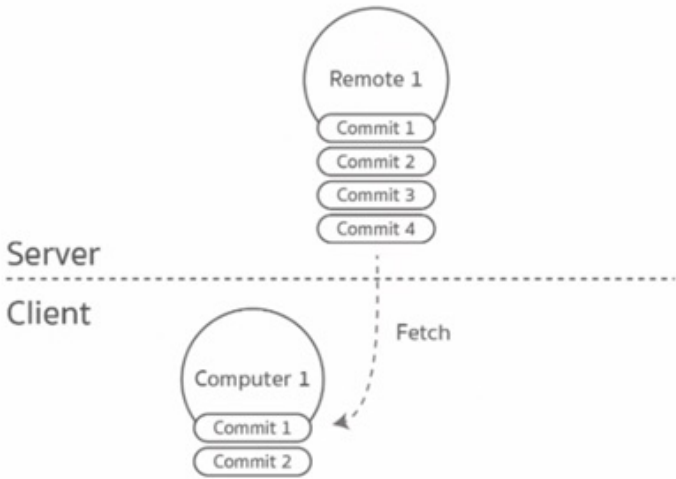
3.8. Pull คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

คือ เวลา Sync จาก Remote เพื่อดึงข้อมูล Commit ใหม่ลงมาเก็บไว้ในเครื่องจะเรียกขั้นตอนนี้ว่า Pull



3.9. Fetch คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

อยากเช็คสถานะของ Remote iewว่ามีใคร Push ข้อมูลใหม่ขึ้นไป Remote หรือป่าว เราเรียกวิธีนี้ว่า Fetch

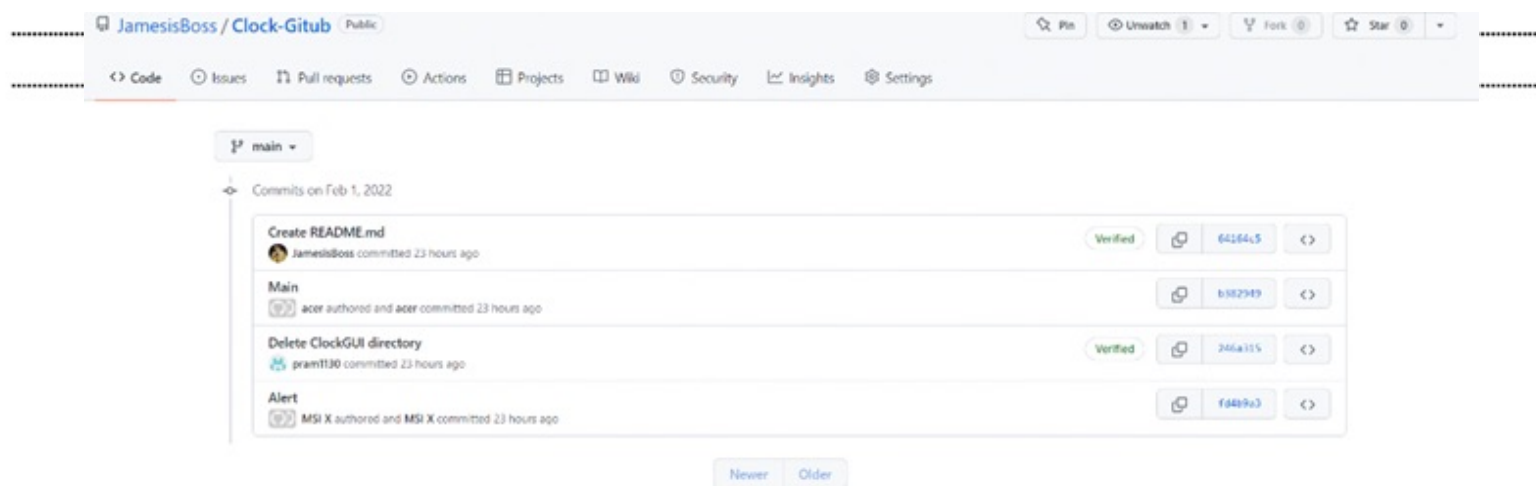


### 3.10. Conflict ใน VSC คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

คือ การเกิดปัญหาการชนของข้อมูลในไฟล์งานที่ทำร่วมกันกับเพื่อนเรา ซึ่งในช่วงที่เราพัฒนาโปรแกรมหรือเขียนโค้ดกับเพื่อนร่วมงานอยู่นั้นเราไม่สามารถรู้ได้เลยว่าเพื่อนเราจะเขียนโค้ดไปในรูปแบบไหน

### 3.11. Merge Commit คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

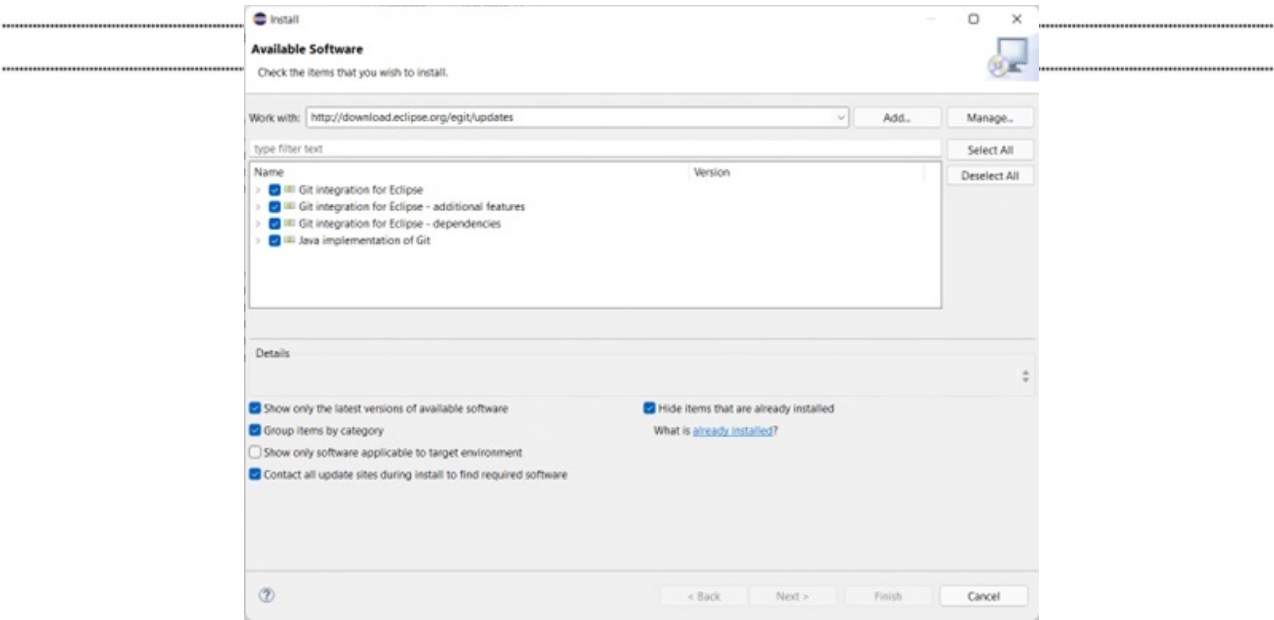
คือการที่มีการแตก branch ออกไป develop แยกกัน โดยที่มีการแก้ไขไฟล์เดียวกันซึ่งโค้ดนั้นอาจมีการทับซ้อน หรืออยู่บรรทัดเดียวกัน เมื่อใครคนใดคนหนึ่งนำโค้ดมา Merge รวมกันนั้นจะเกิดสิ่งที่เรียกว่า Conflict คือโค้ดของทั้งสองคนมีความขัดแย้งกัน



### 3.12. ขั้นตอนที่อยู่ในระหว่าง Development Process ภายใน VSC มีอะไรบ้าง?

3.13. จงบอกและอธิบายขั้นตอนการติดตั้งส่วนขยายใน Eclipse เพื่อให้ใช้งาน Git

- 1. Install Plugin ทำการ Click ไปที่ Help และ Install new software
- 2. จากนั้นก็พิมพ์ <http://download.eclipse.org/egit/updates> ลงในช่อง URL แล้วคลิกที่ Egit



3.หลังจากนั้นกด Next แล้วรออาจใช้เวลาานาน รอนกว่าตัวโปรแกรมขึ้นให้ restart แล้วเปิดโปรแกรมใหม่หลังจากนั้นก็สามารทใช้ส่วนของ Git ได้เลย

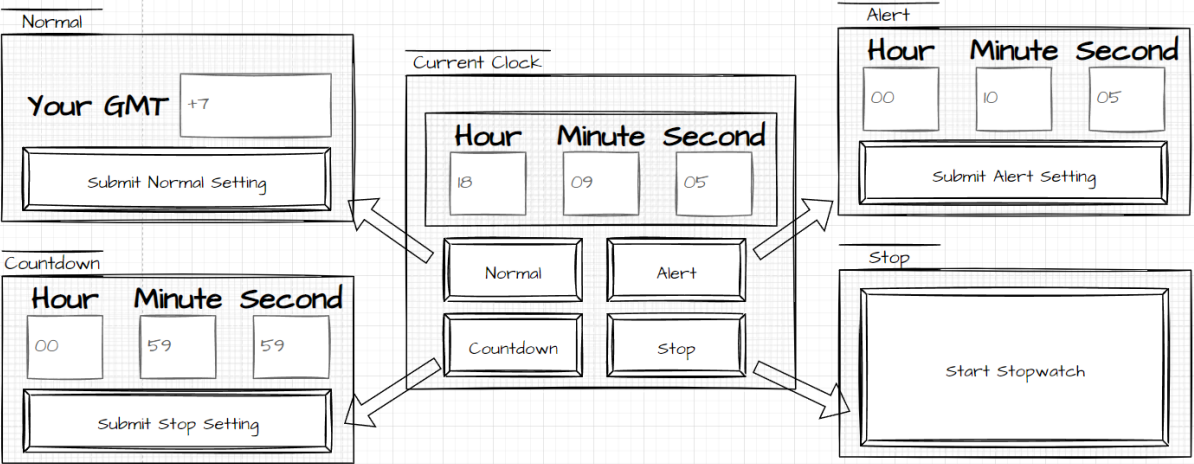


4. ลำดับขั้นการปฏิบัติการ

- 4.1. ลงทะเบียน Github และตกแต่ง Profile ของตนเองให้เรียบร้อย
- 4.2. สร้าง Repository ใน Github
- 4.3. ทำการติดตั้งส่วนเสริมของ Git ลงใน Eclipse เพื่อเตรียมใช้งาน Version Control System ของ Github
- 4.4. การสร้างผลงานโค้ดโปรแกรมใน Github
  - 4.4.1. เชื่อมต่อ Eclipse ของคุณเข้ากับ Github
  - 4.4.2. ทำการ Push โค้ดโปรแกรมตั้งแต่การทดลองที่ 1 ถึง 8 ขึ้นสู่ Remote ใน Github ผ่านโปรแกรม Eclipse

ลิงค์ Github ที่เก็บไฟล์ข้อมูลของการทดลองที่ 1 ถึง 8 ของคุณ	
ลิงค์การทดลองที่ 1 ->	<a href="https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB1/src/LAB1">https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB1/src/LAB1</a>
ลิงค์การทดลองที่ 2 ->	<a href="https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB2/src/LAB2">https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB2/src/LAB2</a>
ลิงค์การทดลองที่ 3 ->	<a href="https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB3/src/Lab3">https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB3/src/Lab3</a>
ลิงค์การทดลองที่ 4 ->	<a href="https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB4/src/Lab4">https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB4/src/Lab4</a>
ลิงค์การทดลองที่ 5 ->	<a href="https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB5/src">https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB5/src</a>
ลิงค์การทดลองที่ 6 ->	<a href="https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB6/src">https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB6/src</a>
ลิงค์การทดลองที่ 7 ->	<a href="https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB7/src/Lab7">https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB7/src/Lab7</a>
ลิงค์การทดลองที่ 8 ->	<a href="https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB8/src/Lab8">https://github.com/benjaphorn/MSI-X-and-MSI-X/tree/main/LAB8/src/Lab8</a>

- 4.5. ทำการ Push โค้ดโปรแกรมตั้งแต่การทดลองที่ 1 ถึง 8 ขึ้นสู่ Remote โดยใช้โปรแกรม Eclipse
- 4.6. สร้างโปรเจคใหม่ใน Eclipse ที่เชื่อมต่อกับ Github ให้เรียบร้อย พร้อมทั้งหาสมาชิกในกลุ่มจำนวน 3-4 คน เพื่อสร้างโปรแกรม “นาฬิกาสารพัดประโยชน์” ที่มีส่วนประกอบของฟิจเจอร์ต่างๆ ดังนี้



- 4.6.1. หน้าต่าง Current Clock เพื่อแสดงนาฬิกาที่จะทำงานตามโหมดต่างๆ ที่ผู้ใช้สั่งตามปุ่มต่างๆ
- 4.6.2. หน้าต่าง Normal จะปรากฏหน้าต่างนี้เมื่อคลิกปุ่ม Normal ที่อยู่ในหน้า Current Clock ซึ่งจะ แสดงส่วนการตั้งค่า GMT ให้กับนาฬิกาหลักหลังจากกดปุ่ม Submit Normal Setting เรียบร้อยแล้ว
- 4.6.3. หน้าต่าง Countdown จะปรากฏหน้าต่างนี้เมื่อคลิกปุ่ม Countdown ที่อยู่ในหน้า Current Clock ซึ่งจะ แสดงส่วนการตั้งค่าการนับเวลาถอยหลัง สามารถปรับค่าได้ในระดับชั่วโมง นาที และวินาที หลังจากกดปุ่ม Submit เรียบร้อย หน้าต่างการตั้งค่าจะหายไป และส่วนการแสดงนาฬิกาใน Current Clock ก็จะทำให้การเริ่มต้นนับถอยหลังไปเรื่อยๆ จนถึงเลข 0 นาฬิกา 0 นาที 0 วินาที
- 4.6.4. หน้าต่าง Alert จะปรากฏหน้าต่างนี้เมื่อคลิกปุ่ม Alert ที่อยู่ในหน้า Current Clock ซึ่งจะ แสดงส่วนการตั้งค่าเวลาปลุกเมื่อเวลาปัจจุบันเดินทางมาถึงเวลาที่กำหนดไว้ สามารถปรับค่าได้ในระดับชั่วโมง นาที และวินาที หลังจากกดปุ่ม Submit เรียบร้อย หน้าต่างการตั้งค่าจะหายไป และส่วนการแสดงนาฬิกาใน Current Clock ก็จะแสดงเวลาตามปกติ แต่เมื่อถึงเวลาที่ตั้งปลุกเอาไว้ ระบบก็จะปรากฏหน้าต่างแจ้งเตือน
- 4.6.5. (หากมีสมาชิกในกลุ่มไม่ถึง 4 คน ไม่ต้องทำฟิจเจอร์นี้) หน้าต่าง Stop จะปรากฏหน้าต่างนี้เมื่อคลิกปุ่ม Stop ที่อยู่ในหน้า Current Clock ซึ่งจะ แสดงส่วนการตั้งค่าการจับเวลา หลังจากกดปุ่ม Start Stopwatch เรียบร้อย หน้าต่างการตั้งค่าจะหายไป และส่วนการแสดงนาฬิกาใน Current Clock ก็จะเริ่มต้นจับเวลา โดยเริ่มตั้งแต่ 0 นาฬิกา 0 นาที 0 วินาที และ

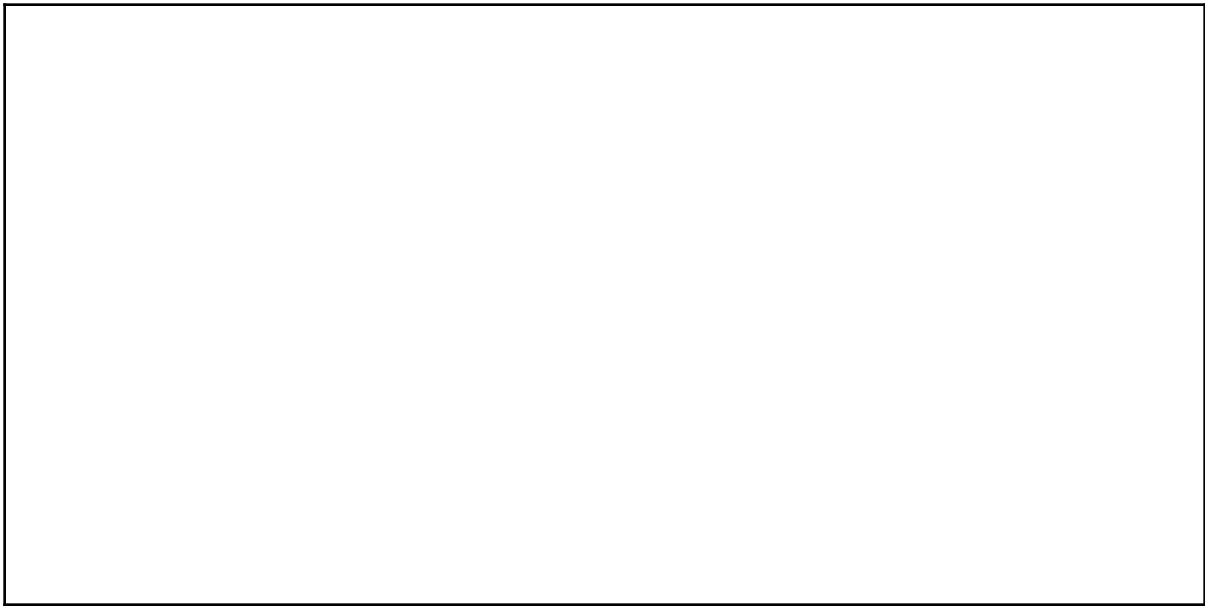
- จำนวนวินาทีจะเริ่มต้นเพิ่มขึ้นไปเรื่อยๆ จนกว่าผู้ใช้งานจะกดปุ่ม Stop อีกครั้ง เพื่อเป็นการหยุดการทำงานของนาฬิกาจับเวลา
- 4.7. จากพีเจเจอร์การทำงานของนาฬิกาข้างต้น ให้นักศึกษาแบ่งหน้าที่ในการกับเพื่อนร่วมงานในกลุ่มเพื่อสร้าง Repository และทำงานร่วมกันภายใน Remote นี้
- 4.7.1. ผู้รับผิดชอบทั้งหมด สร้างและพัฒนาส่วนของ Current Clock
- 4.7.2. ผู้รับผิดชอบคนที่ 1 สร้างและพัฒนาส่วนของ Normal
- 4.7.3. ผู้รับผิดชอบคนที่ 2 สร้างและพัฒนาส่วนของ Countdown
- 4.7.4. ผู้รับผิดชอบคนที่ 3 สร้างและพัฒนาส่วนของ Alert
- 4.7.5. ผู้รับผิดชอบคนที่ 4 (ถ้ามี) สร้างและพัฒนาส่วนของ Stop
- 4.8. นักศึกษาจะต้องทำงานร่วมกัน เพื่อให้เห็นภาพรวมการใช้งาน Eclipse ร่วมกับ Github ให้มองเห็นการทำงานเพื่อการแยก Branch, การ Merge Branch, การจัดการโค้ดโปรแกรมเมื่อเกิด Conflict

รายชื่อสมาชิกภายในกลุ่มของคุณ และหน้าที่รับผิดชอบภายในกลุ่ม			
คนที่ 1	ชื่อ-นามสกุล ..เบญจภรณ์...ไชยเสนา.....	รหัสนักศึกษา ....62543502007-2.....	
	หน้าที่รับผิดชอบ ..... Current Clock,Countdown,Alert.....		

ลิงค์งานกลุ่มของคุณที่อยู่ใน Github
<div>https://github.com/benjaphorn/MSI-X-and-MSI-X-</div>

ผลลัพธ์การทำงานของโปรแกรม
<div><div><div><div>Alert</div><div><div>Hour</div><div>Minute</div><div>Second</div></div><div><div></div><div></div><div></div></div><div>Summit Alert Setting</div></div></div><div><div>Count down</div><div><div>0 ชั่วโมง</div><div>0 นาที</div><div>0 วินาที</div></div><div><div></div><div></div><div></div></div><div>Submit Stop Setting</div></div></div> <div><div><div>Current Clock</div><div><div>Hour</div><div>Minute</div><div>Second</div></div><div><div>20</div><div>3</div><div>32</div></div><div><div>Normal</div><div>Alert</div><div>Countdown</div><div>Stop</div></div></div></div>





โค้ดโปรแกรมภายในหน้าต่าง Current Clock

```
1 import org.eclipse.swt.widgets.Display;
2 import org.eclipse.swt.widgets.Shell;
3 import org.eclipse.swt.widgets.Composite;
4 import org.eclipse.swt.SWT;
5 import org.eclipse.swt.widgets.Button;
6 import org.eclipse.swt.widgets.Label;
7 import org.eclipse.swt.widgets.Text;
8 import org.eclipse.swt.events.SelectionAdapter;
9 import org.eclipse.swt.events.SelectionEvent;
10 import org.eclipse.swt.widgets.DateTime;
11 import org.eclipse.swt.widgets.ResourceManager;
12
13 import javax.swing.*;
14 import java.awt.*;
15 import java.text.SimpleDateFormat;
16 import java.time.LocalDateTime;
17 import java.time.format.DateTimeFormatter;
18 import java.util.Calendar;
19 import java.util.GregorianCalendar;
20 import java.text.SimpleDateFormat;
21 import org.eclipse.ui.forms.widgets.FormToolkit;
22
23 public class Main1 {
24
25     protected Shell shell;
26
27
28     SimpleDateFormat timeFormat;
29
30     private String JH;
31     private String JM;
32     private String JS;
33     private Text Hour;
34     private int sec ;
35     private int minute ;
36     private int hour ;
37     public int Gmt = 0;
38     public int r = 0;
39     public int ah = 0;
40     public int am = 0;
41     public int as = 0;
42     Normal Nm = new Normal();
43     CLKAlertA A1 = new CLKAlertA ();
44     CLKAlertB A2 = new CLKAlertB();
45     Countdown Cd = new Countdown();
46     private Text Min;
47     private Text Sec;
48
49     /**
50      * Launch the application.
51      * @param args
52      */
53     public static void main(String[] args) {
54
55
56         try {
57
58             Main1 window = new Main1();
```

```
59         window.open();
60
61     } catch (Exception e) {
62         e.printStackTrace();
63     }
64 }
65
66 }
67
68 /**
69  * Open the window.
70  */
71
72 public void open() {
73     Display display = Display.getDefault();
74     setTime();
75     createContents();
76     shell.open();
77     shell.layout();
78
79     while (!shell.isDisposed()) {
80         if (!display.readAndDispatch()) {
81             display.sleep();
82         }
83     }
84 }
85
86 }
87
88 }
89
90
91
92 /**
93  * Create contents of the window.
94  */
95
96 protected void createContents() {
97     shell = new Shell();
98     shell.setSize(473, 327);
99     shell.setText("Current Clock");
100
101     //setTime();
102     Composite composite = new Composite(shell, SWT.NONE);
103     composite.setBounds(30, 10, 415, 136);
104
105     Label lblH = new Label(composite, SWT.NONE);
106     lblH.setBounds(72, 10, 59, 14);
107     lblH.setText("Hour");
108
109     Label lblM = new Label(composite, SWT.NONE);
110     lblM.setBounds(189, 10, 59, 14);
111     lblM.setText("Minute");
112
113     Label lblS = new Label(composite, SWT.NONE);
114     lblS.setFont(SWTResourceManager.getFont(".AppleSystemUIFont", 11, SWT.NORMAL));
115     lblS.setBounds(316, 10, 59, 14);
116 }
```

```

117 lblS.setText("Second");
118
119 Hour = new Text(composite, SWT.BORDER);
120 Hour.setFont(SWTResourceManager.getFont(".AppleSystemUIFont", 30, SWT.NORMAL));
121 Hour.setEditable(false);
122 Hour.setBounds(35, 30, 102, 79);
123
124 Min = new Text(composite, SWT.BORDER);
125 Min.setFont(SWTResourceManager.getFont(".AppleSystemUIFont", 30, SWT.NORMAL));
126 Min.setEditable(false);
127 Min.setBounds(161, 30, 102, 79);
128
129 Sec = new Text(composite, SWT.BORDER);
130 Sec.setFont(SWTResourceManager.getFont(".AppleSystemUIFont", 30, SWT.NORMAL));
131 Sec.setEditable(false);
132 Sec.setBounds(288, 30, 102, 79);
133 //formToolkit.adapt(text, true, true);
134
135 //test Fetch
136
137
138
139 Button btnNewButton = new Button(shell, SWT.NONE);
140 btnNewButton.addSelectionListener(new SelectionAdapter() {
141
142     public void widgetSelected(SelectionEvent e) {
143
144         Nm.open();
145
146         Gmt = Nm.getGMT();
147         if(Gmt >=24) {
148             Gmt = Gmt - 24;
149
150         }
151
152
153
154     }
155 });
156 btnNewButton.setBounds(41, 166, 177, 49);
157 btnNewButton.setText("Normal");
158
159 Button btnNewButton_1 = new Button(shell, SWT.NONE);
160 btnNewButton_1.addSelectionListener(new SelectionAdapter() {
161     @Override
162     public void widgetSelected(SelectionEvent e) {
163
164
165         Al.open();
166         setAlert();
167
168     }
169 });
170 btnNewButton_1.setBounds(247, 166, 177, 49);
171 btnNewButton_1.setText("Alert");
172
173 Button btnNewButton_2 = new Button(shell, SWT.NONE);
174 btnNewButton_2.addSelectionListener(new SelectionAdapter() {

```

นางสาว เบญจวรรณ ไชยเสนา  
62543502007-2 คอบ.คพ

```

175- @Override
176- public void widgetSelected(SelectionEvent e){
177-
178-     Cd.open();
179-
180- }
181- });
182- btnNewButton_2.setBounds(41, 231, 177, 49);
183- btnNewButton_2.setText("Countdown");
184-
185- Button btnNewButton_3 = new Button(shell, SWT.NONE);
186- btnNewButton_3.addSelectionListener(new SelectionAdapter() {
187-     @Override
188-     public void widgetSelected(SelectionEvent e) {
189-     }
190- });
191- btnNewButton_3.setBounds(247, 231, 177, 49);
192- btnNewButton_3.setText("Stop");
193-
194-
195-
196- }
197-
198-
199- public void setTime() {
200-
201-     new Thread(new Runnable() {
202-         public void run() {
203-             while (true) {
204-                 try { Thread.sleep(1000); } catch (Exception e) { }
205-                 Display.getDefault().asyncExec(new Runnable() {
206-                     public void run() {
207-
208-                         Calendar cal = new GregorianCalendar();
209-                         minute = cal.get(Calendar.MINUTE);
210-                         hour = cal.get(Calendar.HOUR_OF_DAY);
211-                         sec = cal.get(Calendar.SECOND);
212-
213-
214-                         hour = hour+Gmt;
215-
216-
217-                         if(hour >=24) {
218-
219-                             hour = hour- 24;
220-
221-                         }
222-
223-                         if(hour == ah && minute == am && sec == as) {
224-
225-                             Al2.open();

```

นางสาว เบญจกรณ์ ไชยเสนา  
62543502007-2 คอบ.คพ

```

226
227
228     }
229
230     JH = ""+hour;
231     JM = ""+minute;
232     JS = ""+sec;
233     Hour.setText(JH);
234     Min.setText(JM);
235     Sec.setText(JS);
236
237     }
238     });
239     }
240     }
241     }).start();
242
243
244 }
245
246 public void showAlert() {
247
248     ah = Al.h ;
249     am = Al.m ;
250     as = Al.s;
251
252
253 }
254
255 }
256

```

นางสาว เบญจภรณ์ ไชยเสนา  
62543502007-2 คอบ.คพ

โค้ดโปรแกรมภายในหน้าต่าง Normal

```
1 import org.eclipse.swt.widgets.Display;
2 import org.eclipse.swt.widgets.Shell;
3 import org.eclipse.swt.widgets.Label;
4 import org.eclipse.swt.SWT;
5 import org.eclipse.swt.widgets.Label;
6
7 import org.eclipse.swt.widgets.Text;
8 import org.eclipse.swt.widgets.Button;
9 import org.eclipse.swt.events.SelectionAdapter;
10 import org.eclipse.swt.events.SelectionEvent;
11
12 public class Normal {
13
14     protected Shell shell;
15     private Text text;
16     public int GMT = 0;
17
18     /**
19      * Launch the application.
20      * @param args
21      */
22     public static void main(String[] args) {
23         try {
24             Normal window = new Normal();
25             window.open();
26         } catch (Exception e) {
27             e.printStackTrace();
28         }
29     }
30
31     /**
32      * Open the window.
33      */
34     public void open() {
35         Display display = Display.getDefault();
36         createContents();
37         shell.open();
38         shell.layout();
39         while (!shell.isDisposed()) {
40             if (!display.readAndDispatch()) {
41                 display.sleep();
42             }
43         }
44     }
45
46     /**
47      * Create contents of the window.
48      */
49     protected void createContents() {
50         shell = new Shell();
51         shell.setSize(450, 300);
52         shell.setText("Normal");
53         Main1 window1 = new Main1();
54
55         Label lblYourGmt = new Label(shell, SWT.NONE);
56         lblYourGmt.setFont(SWTResourceManager.getFont(".AppleSystemUIFont", 44, SWT.NORMAL));
57         lblYourGmt.setBounds(10, 76, 200, 84);
58         lblYourGmt.setText("Your GMT");
```



```

58
59 text = new Text(shell, SWT.BORDER);
60 text.setBounds(206, 68, 220, 77);
61
62 Button btnNewButton = new Button(shell, SWT.NONE);
63 btnNewButton.addSelectionListener(new SelectionAdapter() {
64     @Override
65     public void widgetSelected(SelectionEvent e) {
66
67
68         GMT = Integer.parseInt(text.getText());
69         window1.r = GMT;
70         shell.close();
71
72
73     }
74 });
75 btnNewButton.setBounds(20, 161, 406, 77);
76 btnNewButton.setText("Summit Normal Setting");
77
78 }
79
80
81 public int getGMT() {
82
83     return this.GMT ;
84 } //end method
85 }
86

```

นางสาว เบญจภรณ์ ไชยเสนา  
62543502007-2 คอบ.คพ



โค้ดโปรแกรมภายในหน้าต่าง Countdown

```
1 import org.eclipse.swt.widgets.Display;
2 import org.eclipse.swt.widgets.Shell;
3
4 import org.eclipse.swt.SWT;
5 import org.eclipse.swt.widgets.Text;
6 import org.eclipse.swt.widgets.Button;
7 import org.eclipse.swt.events.SelectionAdapter;
8 import org.eclipse.swt.events.SelectionEvent;
9
10 import javax.swing.Timer;
11 import org.eclipse.swt.widgets.Label;
12 import org.eclipse.swt.SWTResourceManager;
13 public class Countdown {
14
15     protected Shell shell;
16     private Text text1;
17     private Text text2;
18     private Text text3;
19
20     Timer timer;
21     int sec,min,hour;
22     int usesec1,usesec2,usesec3;
23     String sec1= "";
24     String min1= "";
25     String hour1= "";
26     private Label la3;
27     private Label la2;
28     private Label la1;
29     private Label lblNewLabel;
30     private Label lblNewLabel_1;
31     private Label lblNewLabel_2;
32     /**
33      * Launch the application.
34      * @param args
35      */
36     public static void main(String[] args) {
37         try {
38             Countdown window = new Countdown();
39             window.open();
40         } catch (Exception e) {
41             e.printStackTrace();
42         }
43     }
44
45     /**
46      * Open the window.
47      */
48     public void open() {
49         Display display = Display.getDefault();
50         createContents();
51         shell.open();
52         shell.layout();
53         while (!shell.isDisposed()) {
54             if (!display.readAndDispatch()) {
55                 display.sleep();
56             }
57         }
58     }
59 }
```

```

59
60@ /**
61  * Create contents of the window.
62  */
63@ protected void createContents() {
64     shell = new Shell();
65     shell.setSize(364, 275);
66     shell.setText("Count down");
67
68     text1 = new Text(shell, SWT.BORDER);
69     text1.setBounds(10, 67, 84, 34);
70
71     text2 = new Text(shell, SWT.BORDER);
72     text2.setBounds(132, 67, 84, 34);
73
74     text3 = new Text(shell, SWT.BORDER);
75     text3.setBounds(254, 67, 84, 34);
76
77     Button button1 = new Button(shell, SWT.NONE);
78@ button1.addSelectionListener(new SelectionAdapter() {
79@     @Override
80     public void widgetSelected(SelectionEvent e) {
81         hour1 = text1.getText();
82         la1.setText(hour1);
83         usesec1 = Integer.parseInt(hour1);
84
85         min1 = text2.getText();
86         la2.setText(min1);
87         usesec2 = Integer.parseInt(min1);
88
89         sec1 = text3.getText();
90         la3.setText(sec1);
91         usesec3 = Integer.parseInt(sec1);
92
93@         new Thread(new Runnable() {
94@             public void run() {
95                 while (true) {
96                     try { Thread.sleep(1000); } catch (Exception e) { }
97@                     Display.getDefault().asyncExec(new Runnable() {
98@                         public void run() {
99                             if(usesec3 < 62){
100                                 usesec3--;
101                                 String usetime3 = Integer.toString(usesec3);
102                                 //OptionPane.showMessageDialog(null,"show number get: "+usetime1);
103                                 la3.setText(""+usetime3);
104                                 if(usesec3 == 0) {
105                                     usesec3 = 60;
106                                     if(usesec2 == 0) {
107                                         usesec2 = 60;
108                                         usesec1--;
109                                         String usetime1 = Integer.toString(usesec1);
110                                         la1.setText(""+usetime1);
111                                     }
112                                     usesec2--;
113                                     String usetime2 = Integer.toString(usesec2);
114                                     la2.setText(""+usetime2);
115

```

นางสาว เบญจกรณ์ ไชยเสนา  
62543502007-2 คอบ.คพ

```

116         }
117     }
118     }); //if(usesec1 == 0) {
119         // break;
120         //}
121
122     }
123 }
124 }).start();
125 }
126 });
127 button1.setBounds(80, 153, 174, 52);
128 button1.setText("Submit Stop Setting");
129
130 la3 = new Label(shell, SWT.NONE);
131 la3.setFont(SWTResourceManager.getFont("Segoe UI", 12, SWT.NORMAL));
132 la3.setBounds(276, 40, 24, 21);
133 la3.setText("0");
134
135 la2 = new Label(shell, SWT.NONE);
136 la2.setFont(SWTResourceManager.getFont("Segoe UI", 12, SWT.NORMAL));
137 la2.setBounds(156, 40, 24, 21);
138 la2.setText("0");
139
140 la1 = new Label(shell, SWT.NONE);
141 la1.setFont(SWTResourceManager.getFont("Segoe UI", 12, SWT.NORMAL));
142 la1.setText("0");
143 la1.setBounds(32, 40, 24, 21);
144
145 lblNewLabel = new Label(shell, SWT.NONE);
146 lblNewLabel.setFont(SWTResourceManager.getFont("Segoe UI", 12, SWT.NORMAL));
147 lblNewLabel.setBounds(307, 40, 44, 21);
148 lblNewLabel.setText("\u0E27\u0E34\u0E19\u0E32\u0E17\u0E35");
149
150 lblNewLabel_1 = new Label(shell, SWT.NONE);
151 lblNewLabel_1.setFont(SWTResourceManager.getFont("Segoe UI", 12, SWT.NORMAL));
152 lblNewLabel_1.setText("\u0E19\u0E32\u0E17\u0E35");
153 lblNewLabel_1.setBounds(188, 40, 39, 21);
154
155 lblNewLabel_2 = new Label(shell, SWT.NONE);
156 lblNewLabel_2.setFont(SWTResourceManager.getFont("Segoe UI", 12, SWT.NORMAL));
157 lblNewLabel_2.setText("\u0E0A\u0E31\u0E48\u0E27\u0E42\u0E21\u0E07");
158 lblNewLabel_2.setBounds(62, 40, 44, 21);
159
160 }
161
162 }
163

```

นางสาว เบญจวรรณ ไชยเสนา  
62543502007-2 คอบ.คพ

โค้ดโปรแกรมภายในหน้าต่าง Alert

```
1 import org.eclipse.swt.widgets.Display;
2 import org.eclipse.swt.widgets.Shell;
3 import org.eclipse.swt.widgets.Label;
4 import org.eclipse.swt.SWT;
5 import org.eclipse.swt.widgets.Text;
6 import org.eclipse.swt.widgets.Button;
7 import org.eclipse.swt.SWTResourceManager;
8 import org.eclipse.swt.events.SelectionAdapter;
9 import org.eclipse.swt.events.SelectionEvent;
10
11 public class CLKAlertA {
12
13     protected Shell shell;
14     private Text Hour;
15     private Text Minute;
16     private Text Second;
17     public int h = 0;
18     public int m = 0;
19     public int s = 0;
20
21     /**
22      * Launch the application.
23      * @param args
24      */
25     public static void main(String[] args) {
26         try {
27             CLKAlertA window = new CLKAlertA();
28             window.open();
29         } catch (Exception e) {
30             e.printStackTrace();
31         }
32     }
33
34     /**
35      * Open the window.
36      */
37     public void open() {
38         Display display = Display.getDefault();
39         createContents();
40         shell.open();
41         shell.layout();
42         while (!shell.isDisposed()) {
43             if (!display.readAndDispatch()) {
44                 display.sleep();
45             }
46         }
47     }
48
49     /**
50      * Create contents of the window.
51      */
52     protected void createContents() {
53         shell = new Shell();
54         shell.setSize(450, 300);
55         shell.setText("Alert");
56         Main1 M1 = new Main1();
57
58         Label lblHour = new Label(shell, SWT.NONE);
```



```

59     lblHour.setAlignment(SWT.CENTER);
60     lblHour.setFont(SWTResourceManager.getFont("AngsanaUPC", 14, SWT.NORMAL));
61     lblHour.setBounds(50, 33, 73, 30);
62     lblHour.setText("Hour");
63
64     Hour = new Text(shell, SWT.BORDER);
65     Hour.setBounds(36, 80, 93, 51);
66
67     Button btnSubmit = new Button(shell, SWT.NONE);
68     btnSubmit.addSelectionListener(new SelectionAdapter() {
69         @Override
70         public void widgetSelected(SelectionEvent e) {
71
72             h = Integer.parseInt(Hour.getText());
73             m = Integer.parseInt(Minute.getText());
74             s = Integer.parseInt(Second.getText());
75
76
77             M1.ah = h;
78             M1.am = m;
79             M1.as = s;
80
81             shell.close();
82
83         }
84     });
85     btnSubmit.setFont(SWTResourceManager.getFont("AngsanaUPC", 22, SWT.NORMAL));
86     btnSubmit.setBounds(31, 158, 375, 70);
87     btnSubmit.setText("Submit Alert Setting");
88
89     Minute = new Text(shell, SWT.BORDER);
90     Minute.setBounds(176, 80, 93, 51);
91
92     Second = new Text(shell, SWT.BORDER);
93     Second.setBounds(313, 80, 93, 51);
94
95     Label lblMinute = new Label(shell, SWT.NONE);
96     lblMinute.setAlignment(SWT.CENTER);
97     lblMinute.setText("Minute");
98     lblMinute.setFont(SWTResourceManager.getFont("AngsanaUPC", 16, SWT.NORMAL));
99     lblMinute.setBounds(176, 33, 93, 30);
100
101     Label lblSecond = new Label(shell, SWT.NONE);
102     lblSecond.setAlignment(SWT.CENTER);
103     lblSecond.setText("Second");
104     lblSecond.setFont(SWTResourceManager.getFont("AngsanaUPC", 16, SWT.NORMAL));
105     lblSecond.setBounds(306, 33, 110, 30);
106
107 }
108 }
109

```

นางสาว เบญจวรรณ ไชยเสนา  
62543502007-2 คอบ.คพ

```

1 import org.eclipse.swt.widgets.Display;
2 import org.eclipse.swt.widgets.Shell;
3 import org.eclipse.swt.widgets.Label;
4 import org.eclipse.swt.SWT;
5 import org.eclipse.swt.widgets.Composite;
6 import org.eclipse.swt.widgets.Button;
7 import org.eclipse.swt.events.SelectionAdapter;
8 import org.eclipse.swt.events.SelectionEvent;
9
10
11 public class CLKAlertB {
12
13     protected Shell shell;
14
15     /**
16      * Launch the application.
17      * @param args
18      */
19     public static void main(String[] args) {
20         try {
21             CLKAlertB window = new CLKAlertB();
22             window.open();
23         } catch (Exception e) {
24             e.printStackTrace();
25         }
26     }
27
28     /**
29      * Open the window.
30      */
31     public void open() {
32         Display display = Display.getDefault();
33         createContents();
34         shell.open();
35         shell.layout();
36         while (!shell.isDisposed()) {
37             if (!display.readAndDispatch()) {
38                 display.sleep();
39             }
40         }
41     }
42
43     /**
44      * Create contents of the window.
45      */
46     protected void createContents() {
47         shell = new Shell();
48         shell.setSize(450, 300);
49         shell.setText("Alert!");
50
51         Composite composite = new Composite(shell, SWT.NONE);
52         composite.setBackground(SWTResourceManager.getColor(SWT.COLOR_WHITE));
53         composite.setBounds(46, 28, 348, 106);
54
55         Label lblAlert = new Label(composite, SWT.NONE);
56         lblAlert.setBackground(SWTResourceManager.getColor(255, 0, 0));
57         lblAlert.setAlignment(SWT.CENTER);
58
59         lblAlert.setForeground(SWTResourceManager.getColor(SWT.COLOR_WHITE));
60         lblAlert.setBounds(10, 10, 328, 86);
61         lblAlert.setFont(SWTResourceManager.getFont("BrowalliaUPC", 32, SWT.BOLD));
62         lblAlert.setText("It's Time to Alert!!!!!!");
63
64         Button btnNewButton = new Button(shell, SWT.NONE);
65         btnNewButton.addSelectionListener(new SelectionAdapter() {
66             @Override
67             public void widgetSelected(SelectionEvent e) {
68                 shell.close();
69             }
70         });
71         btnNewButton.setFont(SWTResourceManager.getFont(".AppleSystemUIFont", 17, SWT.NORMAL));
72         btnNewButton.setBounds(86, 152, 265, 76);
73         btnNewButton.setText("Close");
74     }
75 }
76
77 }
78

```

นางสาว เบญจวรรณ ไชยเสนา  
62543502007-2 คอบ.คพ

โค้ดโปรแกรมภายในหน้าต่าง Stop

```
142 public void widgetSelected(SelectionEvent e) {
143
144     Nm.open();
145
146     Gmt = Nm.getGMT();
147     if(Gmt >=24) {
148         Gmt = Gmt - 24;
149     }
150
151
152
153
154 }
155
156 btnNewButton.setBounds(41, 166, 177, 49);
157 btnNewButton.setText("Normal");
158
159 Button btnNewButton_1 = new Button(shell, SWT.NONE);
160 btnNewButton_1.addSelectionListener(new SelectionAdapter() {
161     @Override
162     public void widgetSelected(SelectionEvent e) {
163
164
165         Al.open();
166         setAlert();
167
168     }
169 });
170 btnNewButton_1.setBounds(247, 166, 177, 49);
171 btnNewButton_1.setText("Alert");
172
173 Button btnNewButton_2 = new Button(shell, SWT.NONE);
174 btnNewButton_2.addSelectionListener(new SelectionAdapter() {
175     @Override
176     public void widgetSelected(SelectionEvent e){
177
178         Cd.open();
179
180     }
181 });
182 btnNewButton_2.setBounds(41, 231, 177, 49);
183 btnNewButton_2.setText("Countdown");
184
185 Button btnNewButton_3 = new Button(shell, SWT.NONE);
186 btnNewButton_3.addSelectionListener(new SelectionAdapter() {
187     @Override
188     public void widgetSelected(SelectionEvent e) {
189     }
190 });
191 btnNewButton_3.setBounds(247, 231, 177, 49);
192 btnNewButton_3.setText("Stop");
193
194
195
196 }
```

5. สรุปผลการปฏิบัติการ

สามารถใช้งาน Commit จาก eclipse ไปยัง Github แต่ติดปัญหาเรื่อง Merge งานเพื่อนขึ้นมابนเครื่องเราแล้ว  
ติดปัญหานิดหน่อย แต่ก็สามารถ Merge มาทำได้

.....

.....

.....

.....



6. คำถามท้ายการทดลอง

6.1. ควร Commit อย่างไร เพื่อหลีกเลี่ยงการเกิด Conflict ให้เหมาะสมที่สุด

ทำส่วนของ Project หรือ ตัวไฟล์งานไว้เลย อย่างรวมไฟล์แล้ว commit ทีเดียว เพราะอาจทำให้การ commit.....  
นั้นเกิด การ conflict และอาจทำให้ Pull code มีปัญหาได้.....

6.2. ควรมีหลักเกณฑ์ในการ Push ขึ้นไปบน Remote เมื่อใดจึงจะเหมาะสมที่สุด

เลือกไฟล์ที่ต้องอัปเดตขึ้น git แล้วหลังจากนั้นค่อย Share project เสร็จเช็คก่อนที่จะไป staged changes.....  
ขึ้นต่อไปทำการ commit ก่อนแล้วค่อยไป push and commit.....

6.3. เมื่อใดจึงควรใช้คำสั่ง Fetch

เมื่อต้องการเช็คข้อมูลว่าใครที่ push เข้ามาทำแล้วบ้างเราไม่จำเป็นต้อง pull เข้าเครื่อง.....  
fetch ยังสามารถเช็ค history ทั้งหมดได้ด้วย.....

6.4. เราควรแยก Branch เมื่อใด? และควร Merge Branch เมื่อใด?

เมื่อเราจะอัปเดตไฟล์ไปใน Git Branch เพราะเราต่างคนต่างทำ โค้ดอาจทำให้เวลาทำ Feature อาจไม่รู้ว่าเป็นของหรืออาจ.....  
ทำให้ Source Code รวมอยู่ในไฟล์เดียวกันได้ เราควร Merge Branch เมื่อมีการจะ push โค้ดเข้าไปในตัวของ git hub.....