

เรื่องที่ 5 การใช้งานโปรแกรม Eclipse เพื่อสร้าง Window Application

ENGCE174 การเขียนโปรแกรมเชิงวัตถุ (Object-oriented programming)

อ.กิตตินันท์ น้อยมณี (mr.kittinan@rmutl.ac.th)

Eclipse WindowBuilder

Eclipse WindowBuilder is composed of Eclipse SWT Designer and Eclipse Swing Designer and makes it very easy to create Java GUI applications without spending a lot of time writing code. Use the WYSIWYG visual designer and layout tools to create simple forms to complex windows; the Java code will be generated for you. Easily add controls using drag-and-drop, add event handlers to your controls, change various properties of controls using a property editor, internationalize your app and much more.

<https://projects.eclipse.org/projects/tools.windowbuilder>

<https://www.eclipse.org/windowbuilder/images/wbdemo.mp4>

Eclipse WindowBuilder

[\[เทรดเดอร์ตั้ม\] Java + GUI ep.1 โปรแกรม Hello World แบบคุลๆ \(blockdit.com\)](#)

Help



Welcome



Help Contents



Search

Show Context Help

Show Active Keybindings... Ctrl+Shift+L



Tip of the Day



Tips and Tricks...

Cheat Sheets...



Eclipse User Storage



Perform Setup Tasks...



Check for Updates



Install New Software...



Eclipse Marketplace...



About Eclipse IDE



Donate



Contribute

Available Software

Check the items that you wish to install.

Work with: --All Available Sites-- Add... Manage...

Search: windowbuilder

Name	Version
<input checked="" type="checkbox"/> General Purpose Tools	
<input checked="" type="checkbox"/> WindowBuilder Core	1.9.7.202106041225
<input checked="" type="checkbox"/> WindowBuilder Core Documentation	1.9.7.202106041225
<input checked="" type="checkbox"/> WindowBuilder Core UI	1.9.7.202106041225
<input checked="" type="checkbox"/> WindowBuilder GroupLayout Support	1.9.7.202106041225
<input checked="" type="checkbox"/> WindowBuilder Java Core	1.9.7.202106041225
<input checked="" type="checkbox"/> WindowBuilder Swing Designer	1.9.7.202106041225
<input checked="" type="checkbox"/> WindowBuilder Swing Designer Documentation	1.9.7.202106041225
<input checked="" type="checkbox"/> WindowBuilder SWT Designer	1.9.7.202106041225

11 items selected

Details

☒ Show only the latest versions of available software

☒ Group items by category

☐ Show only software applicable to target environment

☒ Contact all update sites during install to find required software

☒ Hide items that are already installed

What is [already installed](#)?

Next >

Review Licenses

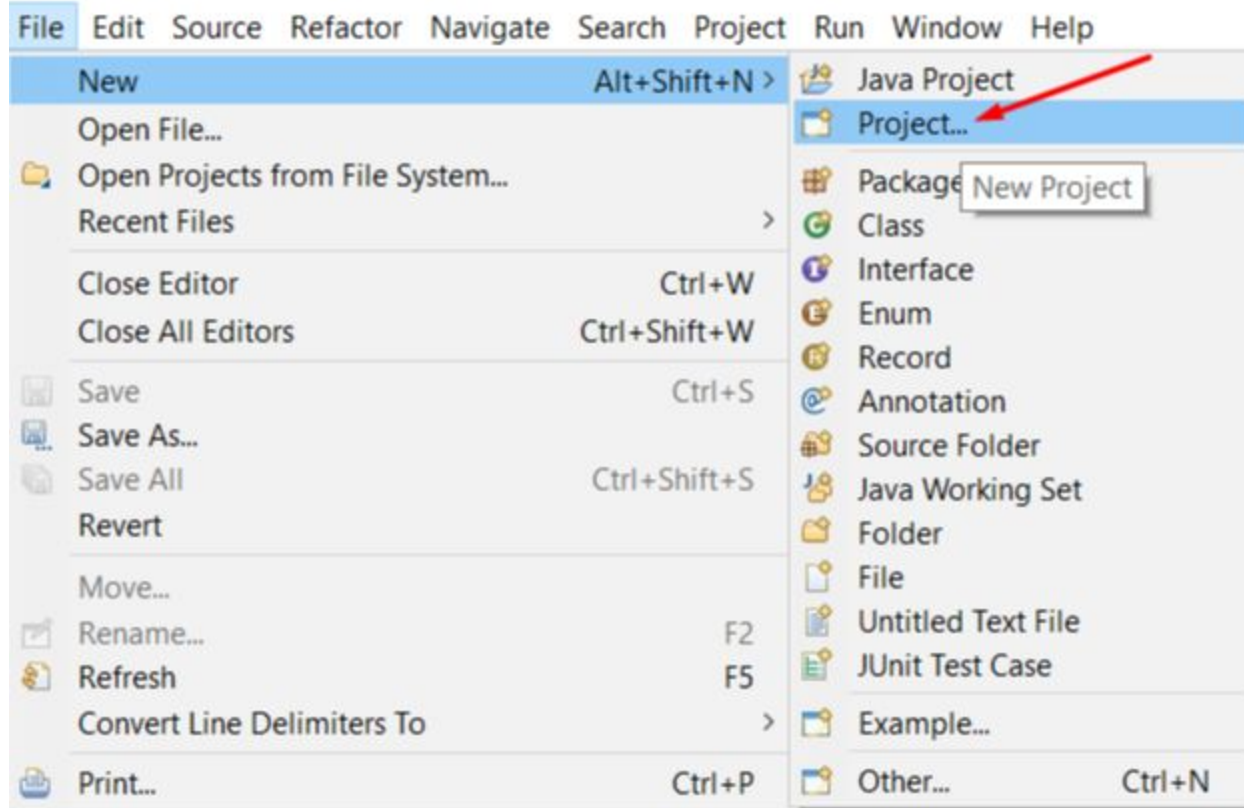
Licenses must be reviewed and accepted before the software can be installed.



Licenses:	License text:
<ul style="list-style-type: none">> Eclipse Foundation Software User Agreement> Eclipse Foundation Software User Agreement	<p>Eclipse Foundation Software User Agreement April 9, 2014</p> <p>Usage Of Content</p> <p>THE ECLIPSE FOUNDATION MAKES AVAILABLE SOFTWARE, DOCUMENTATION, INFORMATION AND/OR OTHER MATERIALS FOR OPEN SOURCE PROJECTS (COLLECTIVELY "CONTENT"). USE OF THE CONTENT IS GOVERNED BY THE TERMS AND CONDITIONS OF THIS AGREEMENT AND/OR THE TERMS AND CONDITIONS OF LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW. BY USING THE CONTENT, YOU AGREE THAT YOUR USE OF THE CONTENT IS GOVERNED BY THIS AGREEMENT AND/OR THE TERMS AND CONDITIONS OF ANY APPLICABLE LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW. IF YOU DO NOT AGREE TO THE TERMS AND CONDITIONS OF THIS AGREEMENT AND THE TERMS AND CONDITIONS OF ANY APPLICABLE LICENSE AGREEMENTS OR NOTICES INDICATED OR REFERENCED BELOW, THEN YOU MAY NOT USE THE CONTENT.</p> <p><input checked="" type="radio"/> I accept the terms of the license agreements</p> <p><input type="radio"/> I do not accept the terms of the license agreements</p>

? < Back Next > Finish Cancel

eclipse-workspace - gui1/src/gui1/main_frame.java - Eclipse IDE



Select a wizard


Create a Java project with SWT/JFace support



Wizards:

type filter text



- > General
- > Eclipse Modeling Framework
- > EJB
- > Gradle
- > Java
- > Java EE
- > Maven
- > Plug-in Development
- > Web
- ▼ WindowBuilder
 - ▼ SWT Designer
 -  SWT/JFace Java Project
- > Examples



< Back

Next >

Finish

Cancel

Create a Java Project

Create a Java project in the workspace or in an external location.



Project name: ThisIsMyFirstGUIProject

☒ Use default location

Location: C:\Users\User01\eclipse-workspace\ThisIsMyFirstGUIProject

Browse...

JRE

☒ Use an execution environment JRE:

JavaSE-16

☐ Use a project specific JRE:

jre

☐ Use default JRE 'jre' and workspace compiler preferences

[Configure JREs...](#)

Project layout

☐ Use project folder as root for sources and class files

☒ Create separate folders for sources and class files

[Configure default...](#)

Working sets

☐ Add project to working sets

New...

Working sets:

Select...

Module

☐ Create module-info.java file







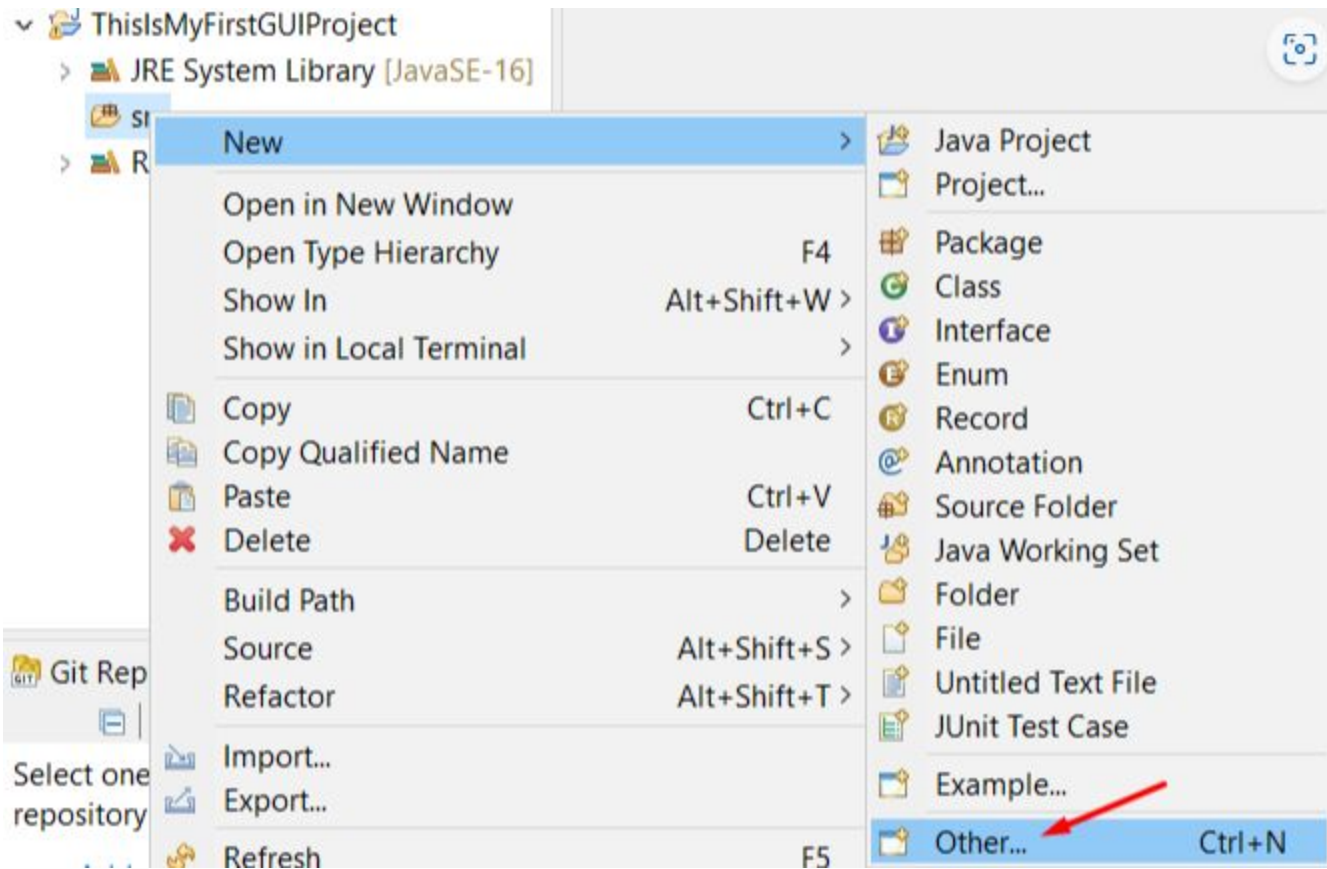
< Back

Next >

Finish

Cancel

- ▼  ThisIsMyFirstGUIProject
 - >  JRE System Library [JavaSE-16]
 -  src ←
 - >  Referenced Libraries



Select a wizard

Create an SWT application window

Wizards:

type filter text

- ▼ WindowBuilder
- Project Palette
- > Swing Designer
- ▼ SWT Designer
 - SWT/JFace Java Project
 - > Databinding
 - > Eclipse 4
 - > Forms
 - > JFace
 - > RCP
 - ▼ SWT
 - Application Window
 - Composite
 - Dialog
 - Shell
- > XML
- > Examples




< Back

Next >

Finish

Cancel

Create SWT Application

 The use of the default package is discouraged.



Source folder: ThisIsMyFirstGUIProject/src

Browse...

Package: (default)

Browse...

Name: FirstGUI

Create contents in:

- ☒ protected createContents() method
☐ public open() method
☐ public static main() method

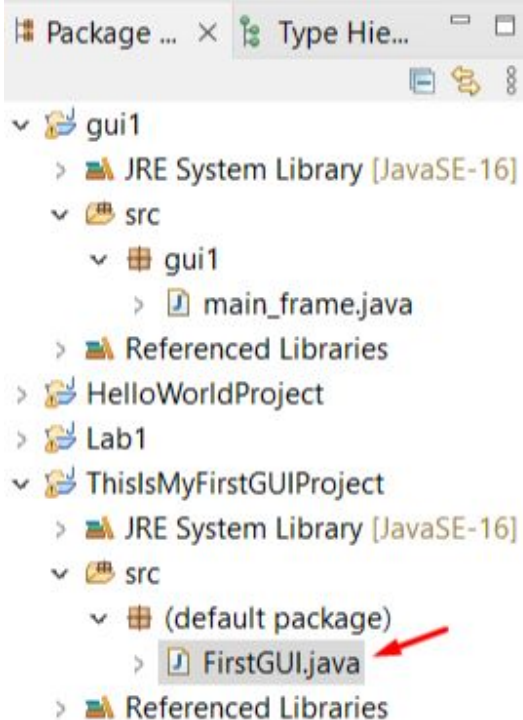


< Back

Next >

Finish

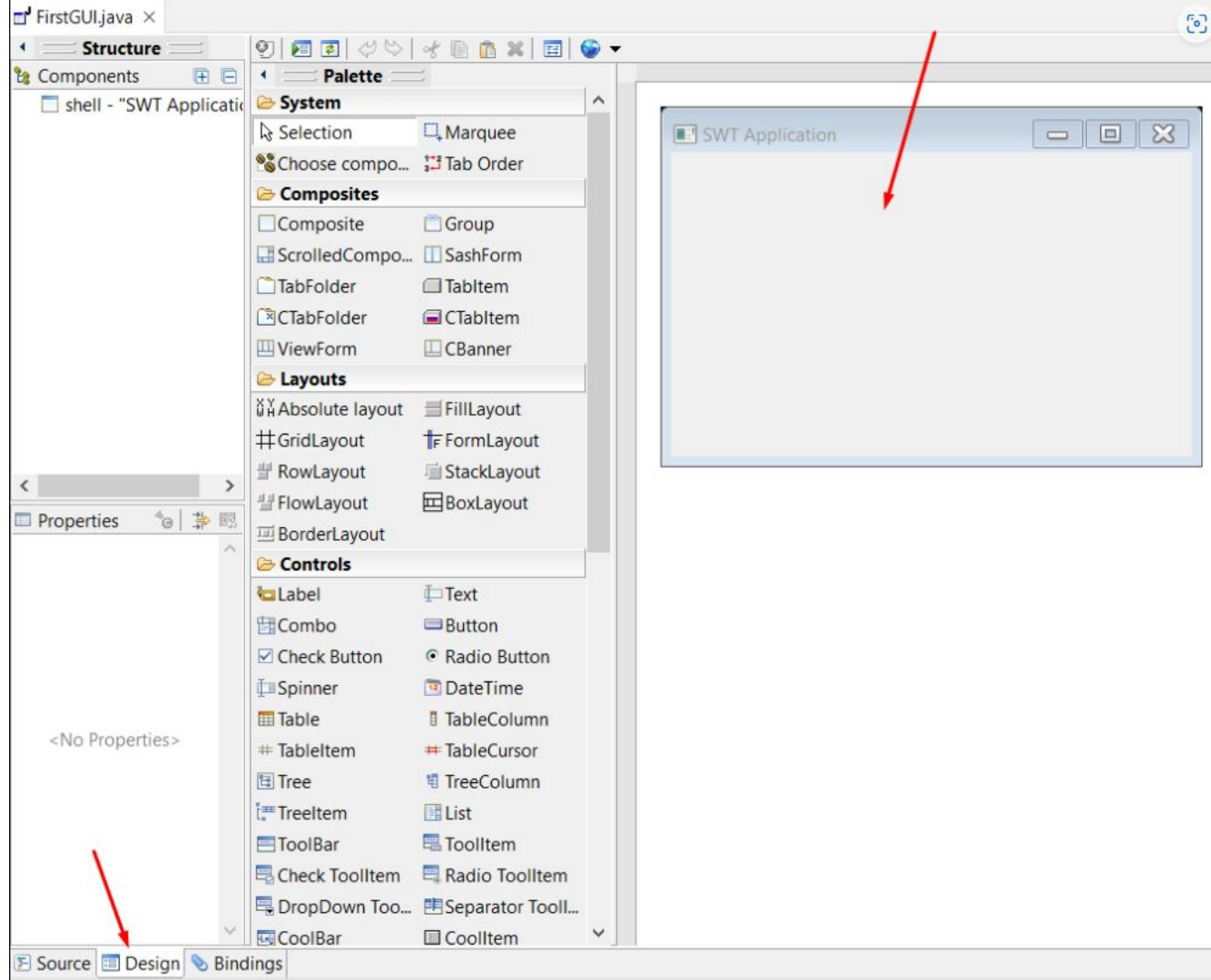
Cancel




FirstGUI.java


```
1 import org.eclipse.swt.widgets.Display;
2
3
4 public class FirstGUI {
5
6     protected Shell shell;
7
8     /**
9      * Launch the application.
10     * @param args
11     */
12     public static void main(String[] args) {
13         try {
14             FirstGUI window = new FirstGUI();
15             window.open();
16         } catch (Exception e) {
17             e.printStackTrace();
18         }
19     }
20
21     /**
22      * Open the window.
23     */
24     public void open() {
25         Display display = Display.getDefault();
26         createContents();
27         shell.open();
```


```
1 import org.eclipse.swt.widgets.Display;
2
3
4 public class FirstGUI {
5
6     protected Shell shell;
7
8     /**
9      * Launch the application.
10     * @param args
11     */
12     public static void main(String[] args) {
13         try {
14             FirstGUI window = new FirstGUI();
15             window.open();
16         } catch (Exception e) {
17             e.printStackTrace();
18         }
19     }
20
21     /**
22     * Open the window.
23     */
24     public void open() {
25         Display display = Display.getDefault();
26         createContents();
27         shell.open();
28         shell.layout();
29     }
30 }
```



◀ **Palette** ▶


 **System**

 **Composites**

 **Layouts**

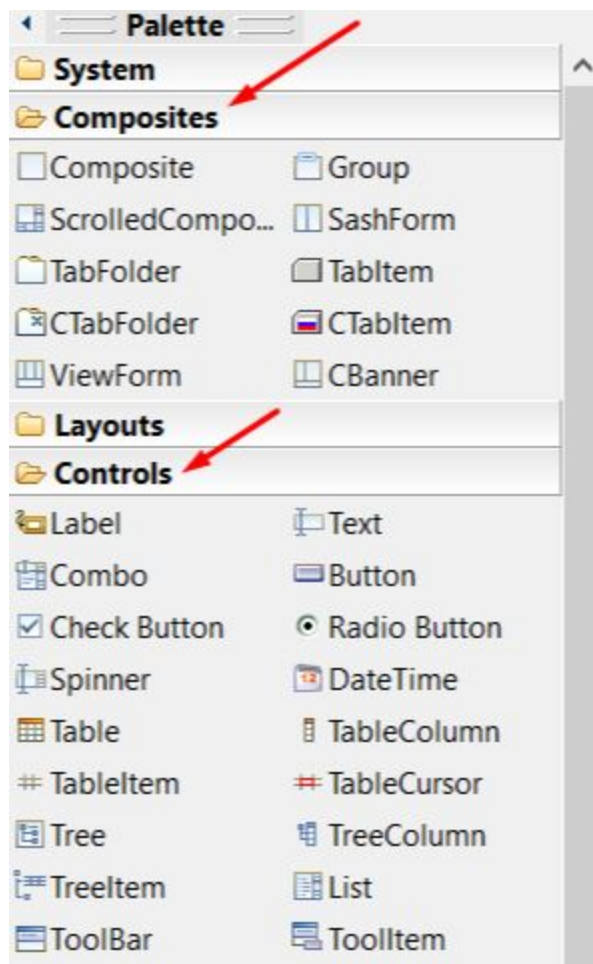
 **Controls**

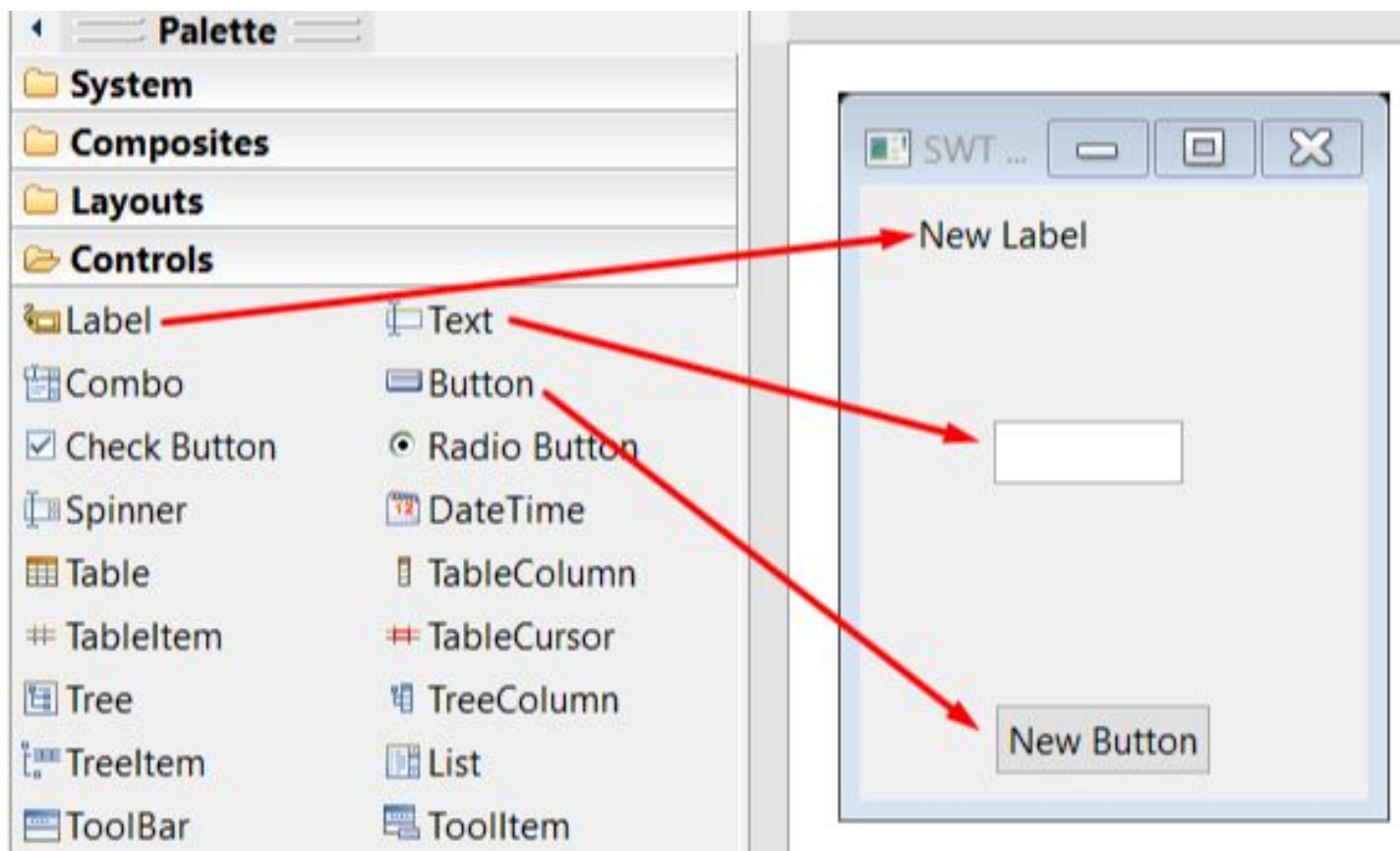
 **JFace**

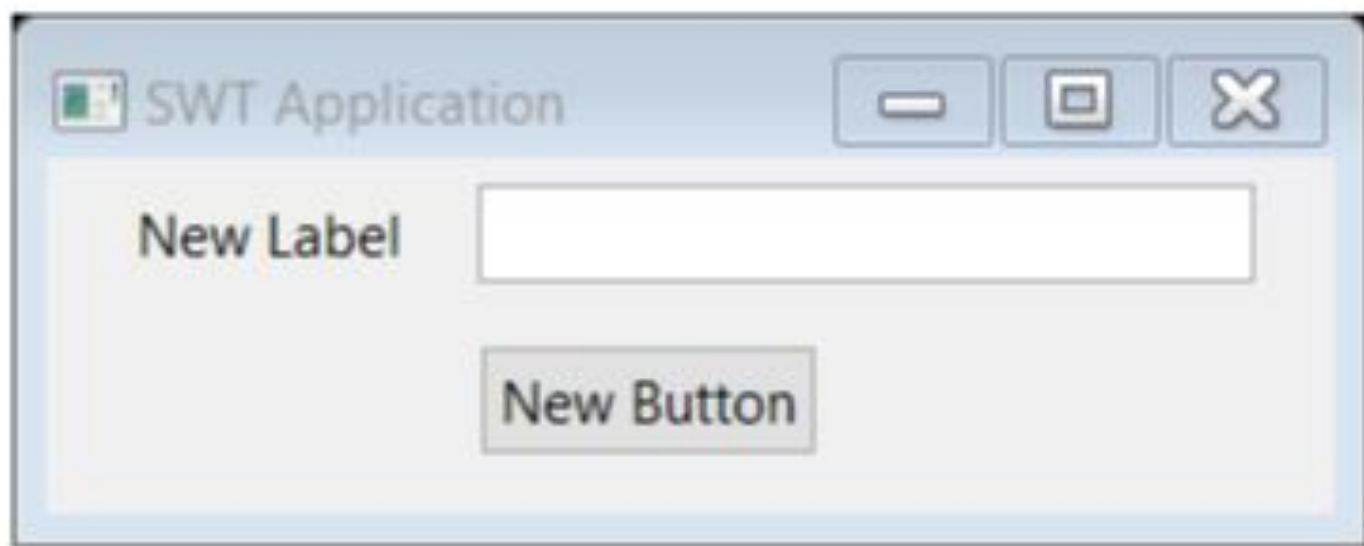
 **Forms API**

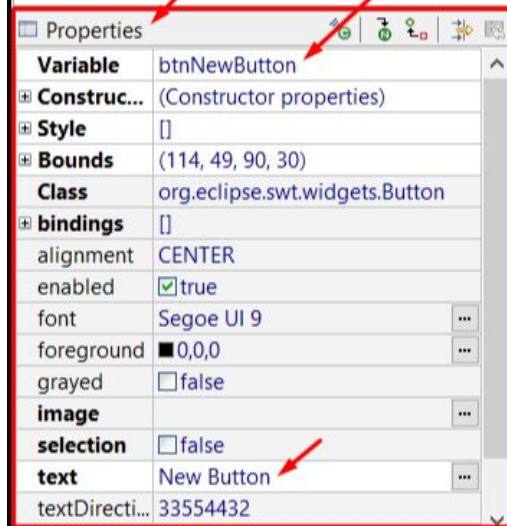
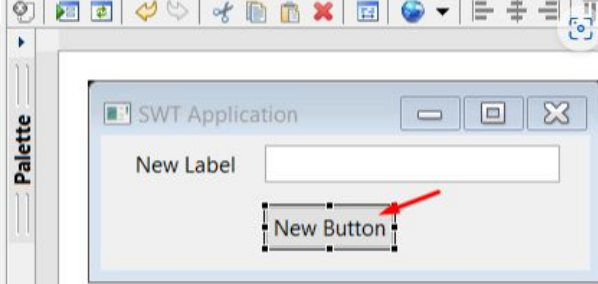
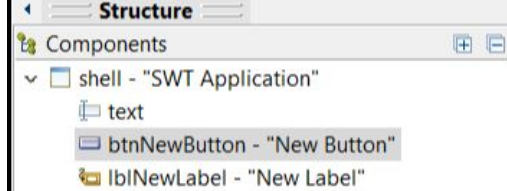
 **Menu**

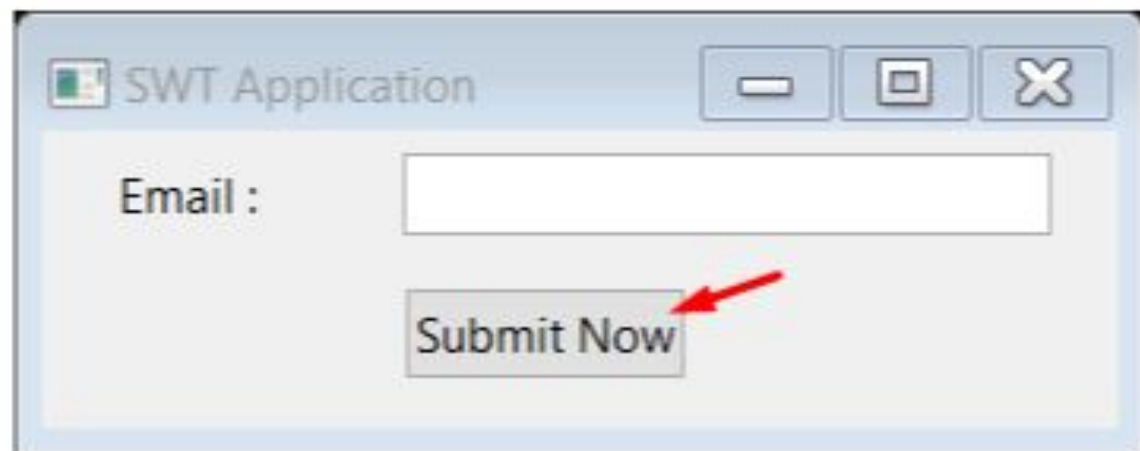
 **SWT_AWT**











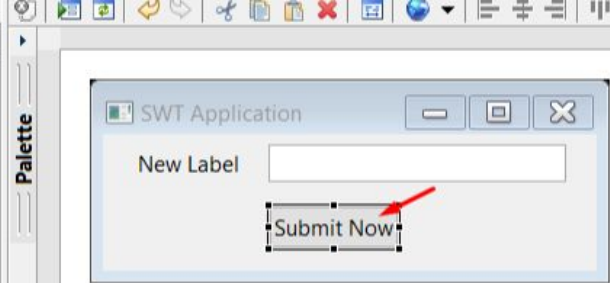
Structure

Components

- shell - "SWT Application"
 - text
 - submit_btn - "Submit Now"
 - IbINewLabel - "New Label"

Properties

Variable	submit_btn
Constructor	(Constructor properties)
Style	[]
Bounds	(114, 49, 90, 30)
Class	org.eclipse.swt.widgets.But...
bindings	[]
alignment	CENTER
enabled	<input checked="" type="checkbox"/> true
font	Segoe UI 9
foreground	■ 0,0,0
grayed	<input type="checkbox"/> false
image	
selection	<input type="checkbox"/> false
text	Submit Now
textDirection	33554432



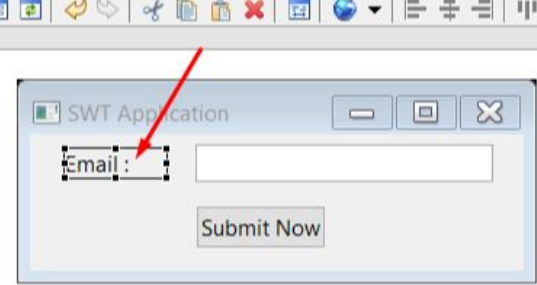
Structure

Components

- shell - "SWT Application"
 - text
 - submit_btn - "Submit Now"
 - name_label - "Email :"

Properties

Variable	name_label
Constructor	(Constructor properties)
Style	[]
Bounds	(24, 10, 70, 20)
Class	org.eclipse.swt.widgets.Lab...
bindings	[]
alignment	LEFT
background	□ 240,240,240
enabled	☑ true
font	Segoe UI 9
foreground	■ 0,0,0
image	
text	Email :
textDirection	33554432
toolTipText	



Structure

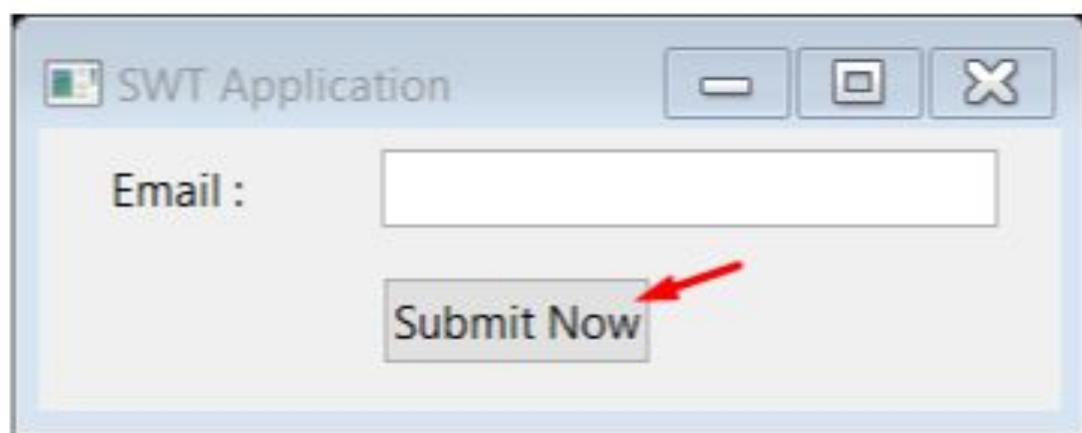
Components

- shell - "SWT Application"
 - text
 - submit_btn - "Submit Now"
 - name_label - "Email :"

Properties

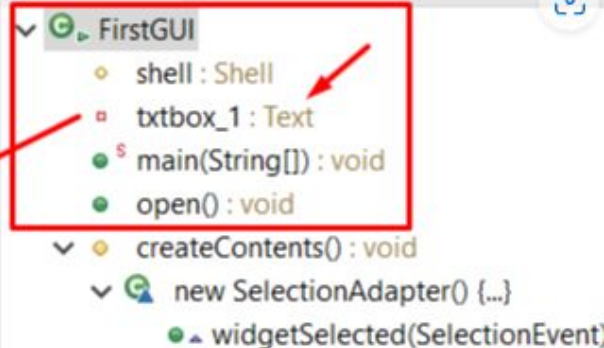
Variable	txtbox_1
Constructor	(Constructor properties)
Style	[BORDER]
Bounds	(114, 7, 205, 26)
Class	org.eclipse.swt.widgets.Text
bindings	[]
background	□ 255,255,255
editable	☑ true
enabled	☑ true
font	Segoe UI 9
foreground	■ 0,0,0
text	
textDirection	33554432
toolTipText	
touchEnabled	☐ false





```
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(358, 141);
49     shell.setText("SWT Application");
50
51     txtbox_1 = new Text(shell, SWT.BORDER);
52     txtbox_1.setBounds(114, 7, 205, 26);
53
54     Button submit_btn = new Button(shell, SWT.NONE);
55     submit_btn.addSelectionListener(new SelectionAdapter() {
56         @Override
57         public void widgetSelected(SelectionEvent e) {
58         }
59     });
60     submit_btn.setBounds(114, 49, 90, 30);
61     submit_btn.setText("Submit Now");
62
63     Label name_label = new Label(shell, SWT.NONE);
64     name_label.setBounds(24, 10, 70, 20);
65     name_label.setText("Email :");
66
67 }
68 }
69
```

```
43  / **
44   * Create contents of the window.
45   */
46  protected void createContents() {
47      shell = new Shell();
48      shell.setSize(358, 141);
49      shell.setText("SWT Application");
50
51      txtbox_1 = new Text(shell, SWT.BORDER);
52      txtbox_1.setBounds(114, 7, 205, 26);
53
54      Button submit_btn = new Button(shell, SWT.NONE);
55      submit_btn.addSelectionListener(new SelectionAdapter() {
56          @Override
57          public void widgetSelected(SelectionEvent e) {
58              txtbox_1.setText( "Hello World" );
59          } //end method
60      });
61      submit_btn.setBounds(114, 49, 90, 30);
62      submit_btn.setText("Submit Now");
63
64      Label name_label = new Label(shell, SWT.NONE);
65      name_label.setBounds(24, 10, 70, 20);
66      name_label.setText("Email :");
67
68  } //end method
69 }
```



The Outline view shows the structure of the FirstGUI class. A red box highlights the class members: shell : Shell, txtbox_1 : Text, main(String[]) : void, and open() : void. Another red box highlights the SelectionAdapter class, which contains the widgetSelected(SelectionEvent) method. Red arrows point from the widgetSelected method in the SelectionAdapter class to the widgetSelected method in the SelectionAdapter class in the code editor.

- FirstGUI
 - shell : Shell
 - txtbox_1 : Text
 - main(String[]) : void
 - open() : void
 - createContents() : void
 - new SelectionAdapter() {...}
 - widgetSelected(SelectionEvent)



SWT Application



Email :

Hello World

Submit Now

FirstGUI.java ×

```
1+ import org.eclipse.swt.widgets.Display;
```

```
9
```

```
10 public class FirstGUI {
```

```
11     protected Shell shell;
```

```
12     private Text txtbox_1;
```

```
13+    public static void main(String[] args) {
```

```
21
```

```
22+    public void open() {
```

```
30
```

```
31+    protected void createContents() {
```

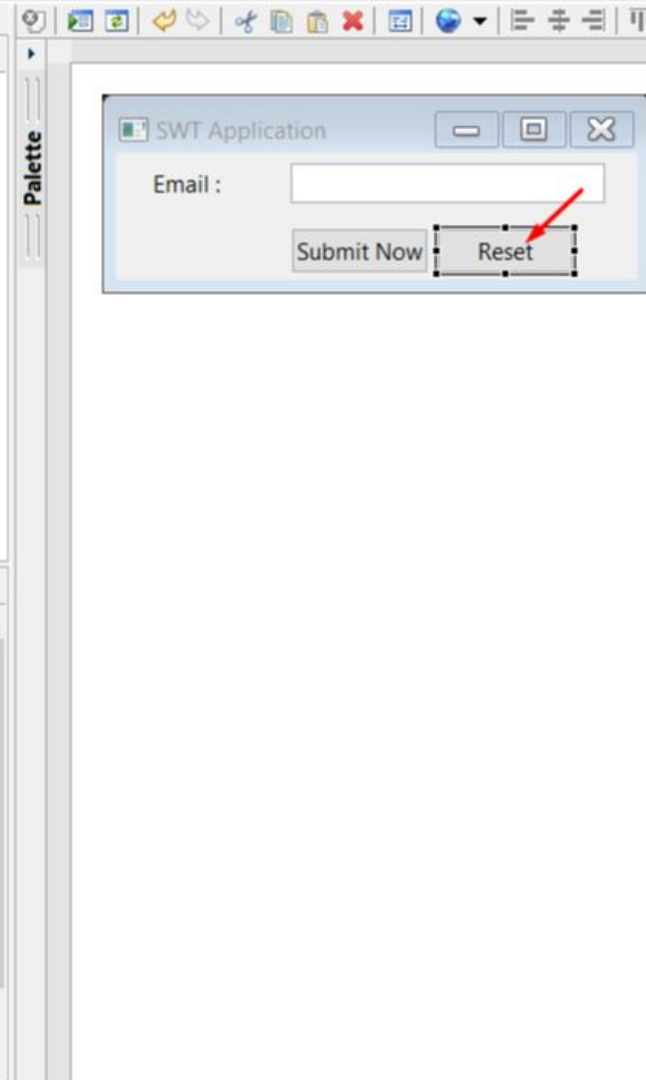
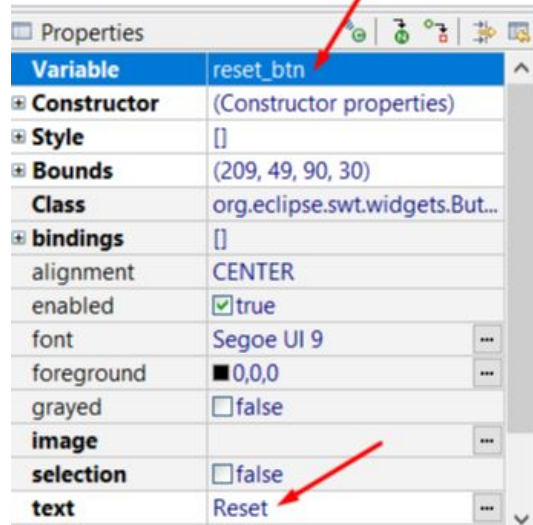
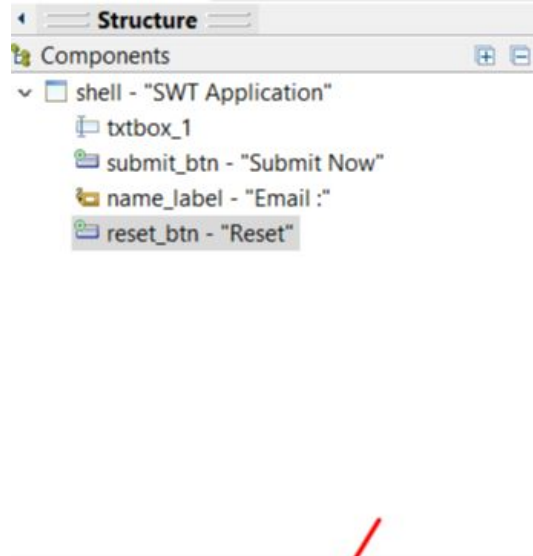
```
54 }//end class
```

```
13⊖    public static void main(String[] args) {  
14        try {  
15            FirstGUI window = new FirstGUI();  
16            window.open();  
17        } catch (Exception e) {  
18            e.printStackTrace();  
19        } //end try catch  
20    } //end method
```

```
22 public void open() {  
23     Display display = Display.getDefault();  
24     createContents();  
25     shell.open();  
26     shell.layout();  
27     while (!shell.isDisposed())  
28         if (!display.readAndDispatch()) display.sleep();  
29 } //end method
```



```
31 protected void createContents() {
32     shell = new Shell();
33     shell.setSize(358, 141);
34     shell.setText("SWT Application");
35
36     txtbox_1 = new Text(shell, SWT.BORDER);
37     txtbox_1.setBounds(114, 7, 205, 26);
38
39     Button submit_btn = new Button(shell, SWT.NONE);
40     submit_btn.addSelectionListener(new SelectionAdapter() {
41         @Override
42         public void widgetSelected(SelectionEvent e) {
43             txtbox_1.setText( "Hello World : " );
44         } //end method
45     });
46     submit_btn.setBounds(114, 49, 90, 30);
47     submit_btn.setText("Submit Now");
48
49     Label name_label = new Label(shell, SWT.NONE);
50     name_label.setBounds(24, 10, 70, 20);
51     name_label.setText("Email :");
52
53     } //end method
```



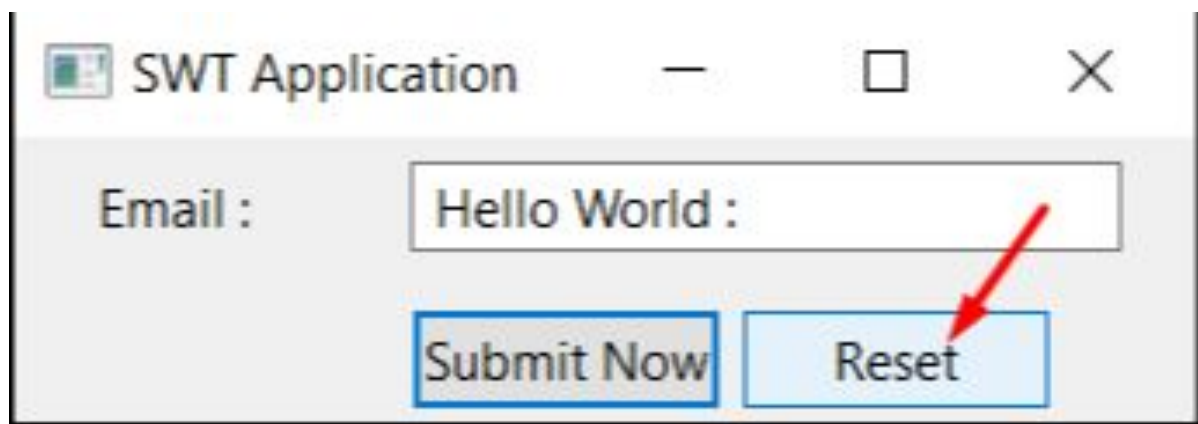
```
protected void createContents() {
    shell = new Shell();
    shell.setSize(358, 130);
    shell.setText("SWT Application");

    txtbox_1 = new Text(shell, SWT.BORDER);
    txtbox_1.setBounds(114, 7, 205, 26);

    Button submit_btn = new Button(shell, SWT.NONE);
    submit_btn.addSelectionListener(new SelectionAdapter() {
        @Override
        public void widgetSelected(SelectionEvent e) {
            txtbox_1.setText( "Hello World : " );
        } //end method
    });
    submit_btn.setBounds(114, 49, 90, 30);
    submit_btn.setText("Submit Now");

    Label name_label = new Label(shell, SWT.NONE);
    name_label.setBounds(24, 10, 70, 20);
    name_label.setText("Email :");

    reset_btn = new Button(shell, SWT.NONE);
    reset_btn.addSelectionListener(new SelectionAdapter() {
        @Override
        public void widgetSelected(SelectionEvent e) {
            txtbox_1.setText( "" );
        } //end method
    });
    reset_btn.setBounds(209, 49, 90, 30);
    reset_btn.setText("Reset");
} //end method
```



```
10 public class FirstGUI {  
11     protected Shell shell;  
12     private Text txtbox_1;  
13     private Button reset_btn;  
14  
15     public int i = 0 ;  
16  
17 public static void main(String[] args) {  
18     try {  
19         FirstGUI window = new FirstGUI();  
20         window.open();  
21     } catch (Exception e) {  
22         e.printStackTrace();  
23     } //end try catch  
24 } //end method  
25
```



```
Button submit_btn = new Button(shell, SWT.NONE);
submit_btn.addSelectionListener(new SelectionAdapter() {
    @Override
    public void widgetSelected(SelectionEvent e) {
        i++ ;
        txtbox_1.setText( "Hello World : " + i ) ;
    } //end method
});
submit_btn.setBounds(114, 49, 90, 30);
submit_btn.setText("Submit Now");
```



```
reset_btn = new Button(shell, SWT.NONE);
reset_btn.addSelectionListener(new SelectionAdapter() {
    @Override
    public void widgetSelected(SelectionEvent e) {
        i = 0 ;
        txtbox_1.setText( "" ) ;
    } //end method
});
reset_btn.setBounds(209, 49, 90, 30);
reset_btn.setText("Reset");
```



SWT Application



Email :

Hello World : 9



Submit Now



Reset