



GUI Control I

Key Event 예제

문자 입력

입력 확인

Inha
Technical





학습 목표

■ 이 강의를 마치면 학생들은

- ❖ Label Control 기능에 대하여 설명할 수 있다.
- ❖ Button Control 기능에 대하여 설명할 수 있다.
- ❖ Text Field Control 기능에 대하여 설명할 수 있다.
- ❖ Text Area Control 기능에 대하여 설명할 수 있다.





Label Control (1)

■ Label Control

❖ 기능

◆ 고정 문자열을 표시할 때 사용

❖ 생성자

이것은
가만 안드려

Constructor	Function
Label()	Blank label 생성
Label(String text)	Text가 좌측에 Layout되는 label 생성
Label(String text, int alignment)	Alignment 값에 따라 text가 Layout된 label 생성 Alignment 값 •Label.LEFT •Label.CENTER •Label.RIGHT





Label Control (2)

❖ Method

Method	Function
String getText()	Label의 Text 반환
void setText(String label)	Label에 Text 설정
void setAlignment (int align)	Label의 Text 위치 결정
int getAlignment()	Label의 Text 위치를 반환





Button Control

■ Button Control

❖ 기능

◆ Button을 생성할 때 사용

❖ 생성자

Constructor	Function
Button()	제목없는 Button 생성
Button(String label)	Label로 주어진 문자열 제목의 Button 생성

❖ Method

Method	Function
String getLabel()	Button 제목 return
void setLabel(String label)	Button 제목 변경

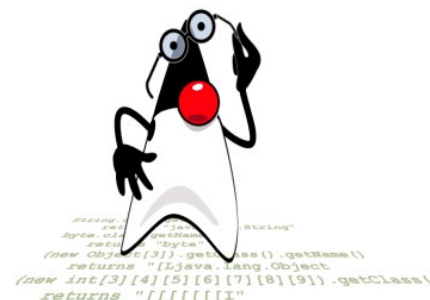
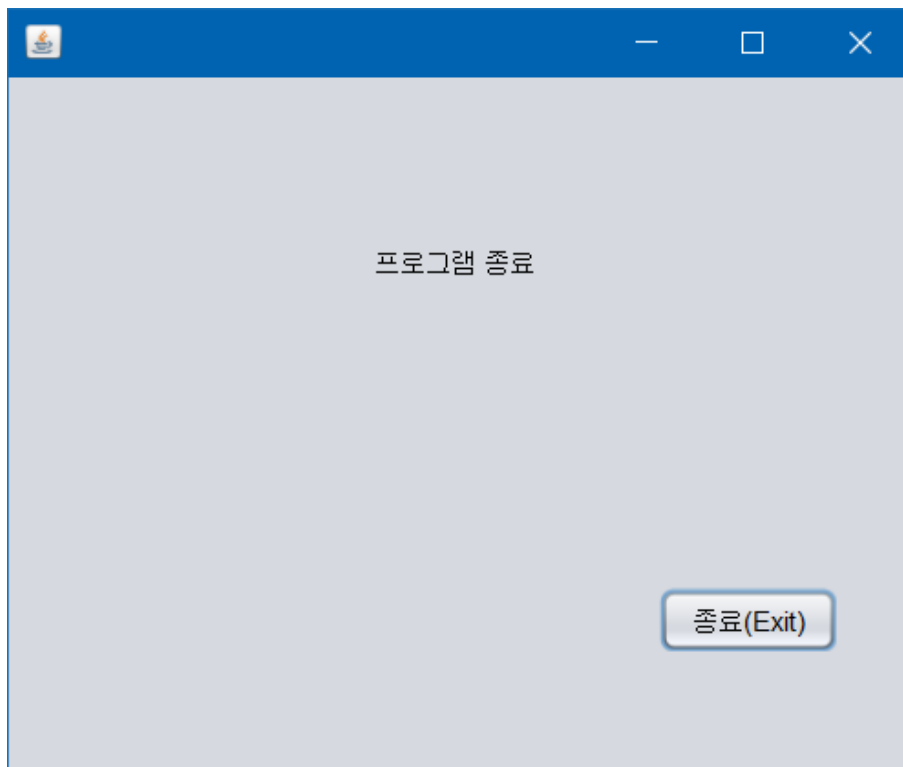




Practice 1 : Label & Button Control

❖ Project Name: LabelButton_Source (Time : 20 min)

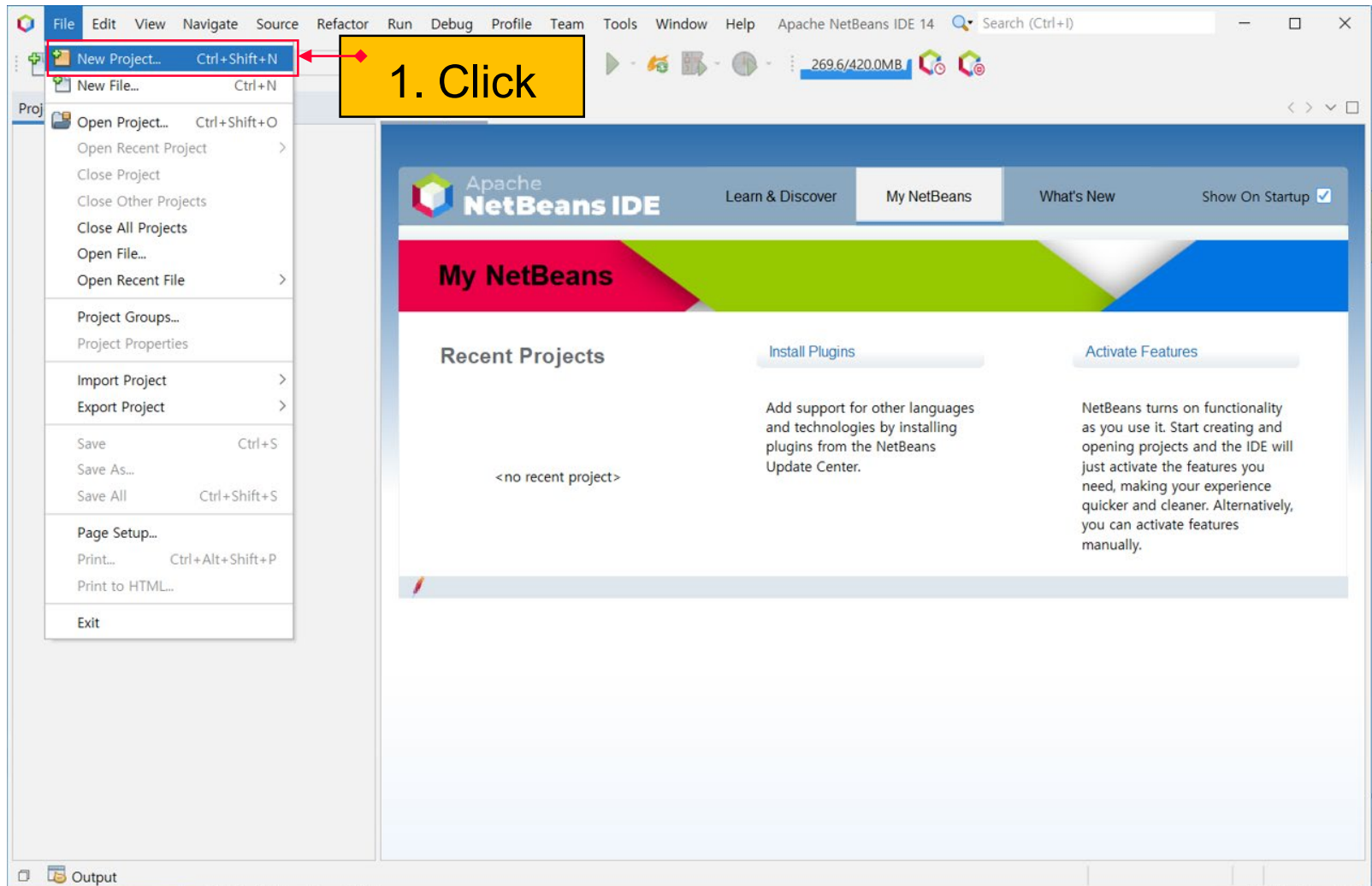
- 종료(Exit) Button을 클릭하면 프로그램이 종료하도록 Coding 하시오.
 - JLabel Control 추가
 - JButton Control 추가





Practice 1 : Label & Button Control (1)

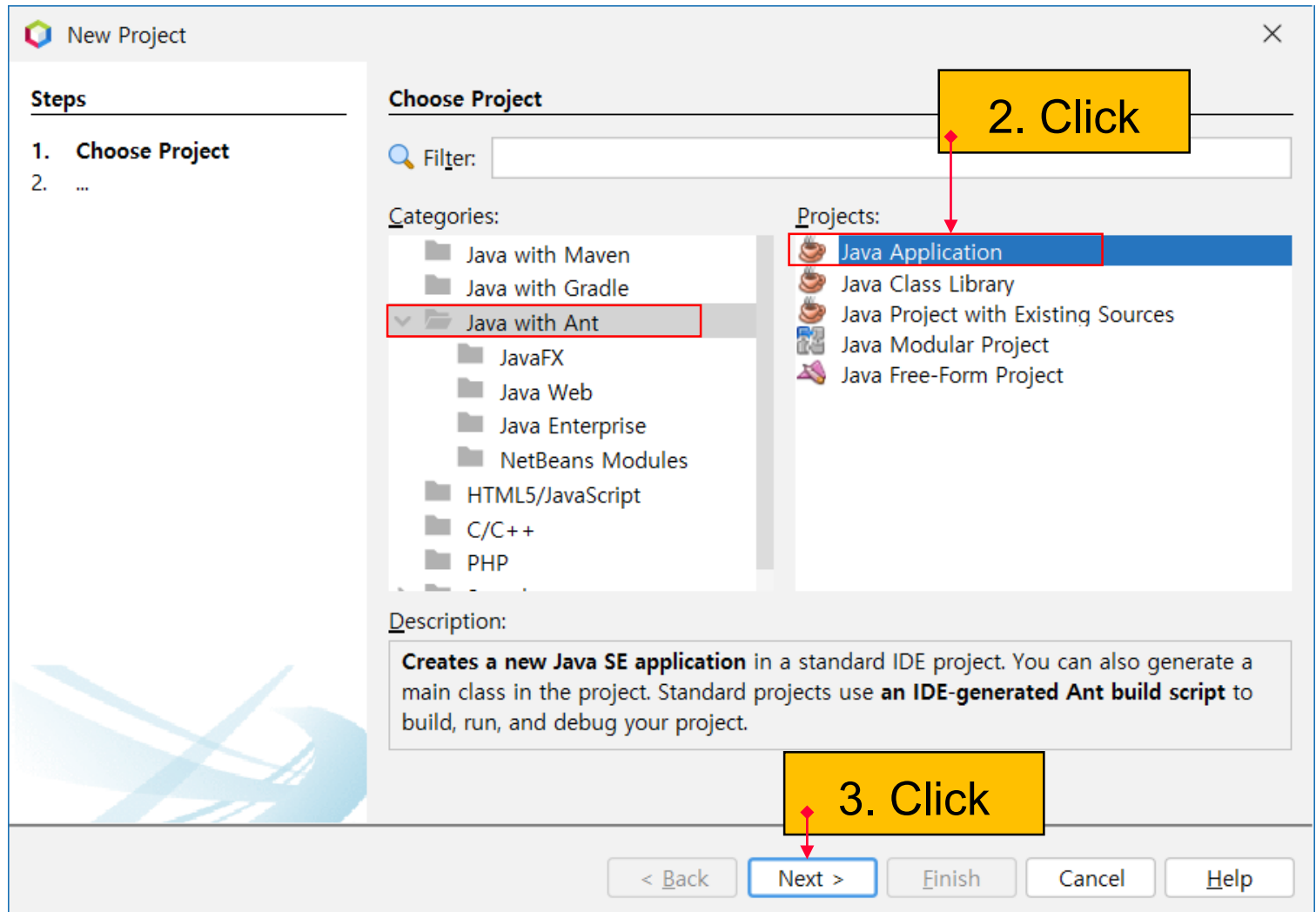
Create Project





Practice 1 : Label & Button Control (2)

Choose Project





Practice 1 : Label & Button Control (3)

■ Project Name and Location

❖ Project name: LabelButton_Source

New Java Application

Steps

1. Choose Project
2. Name and Location

Name and Location

Project Name: LabelButton_Source

Project Location: C:\Java_Project Browse...

Project Folder: C:\Java_Project\LabelButton_Source

☐ Use Dedicated Folder for Storing Libraries

Libraries Folder: Browse...

Different users and projects can share the same compilation libraries (see Help for details).

☐ Create Main Class labelbutton_source.LabelButton_Source

5. Reset check

4. Input "LabelButton_Source"

6. Click

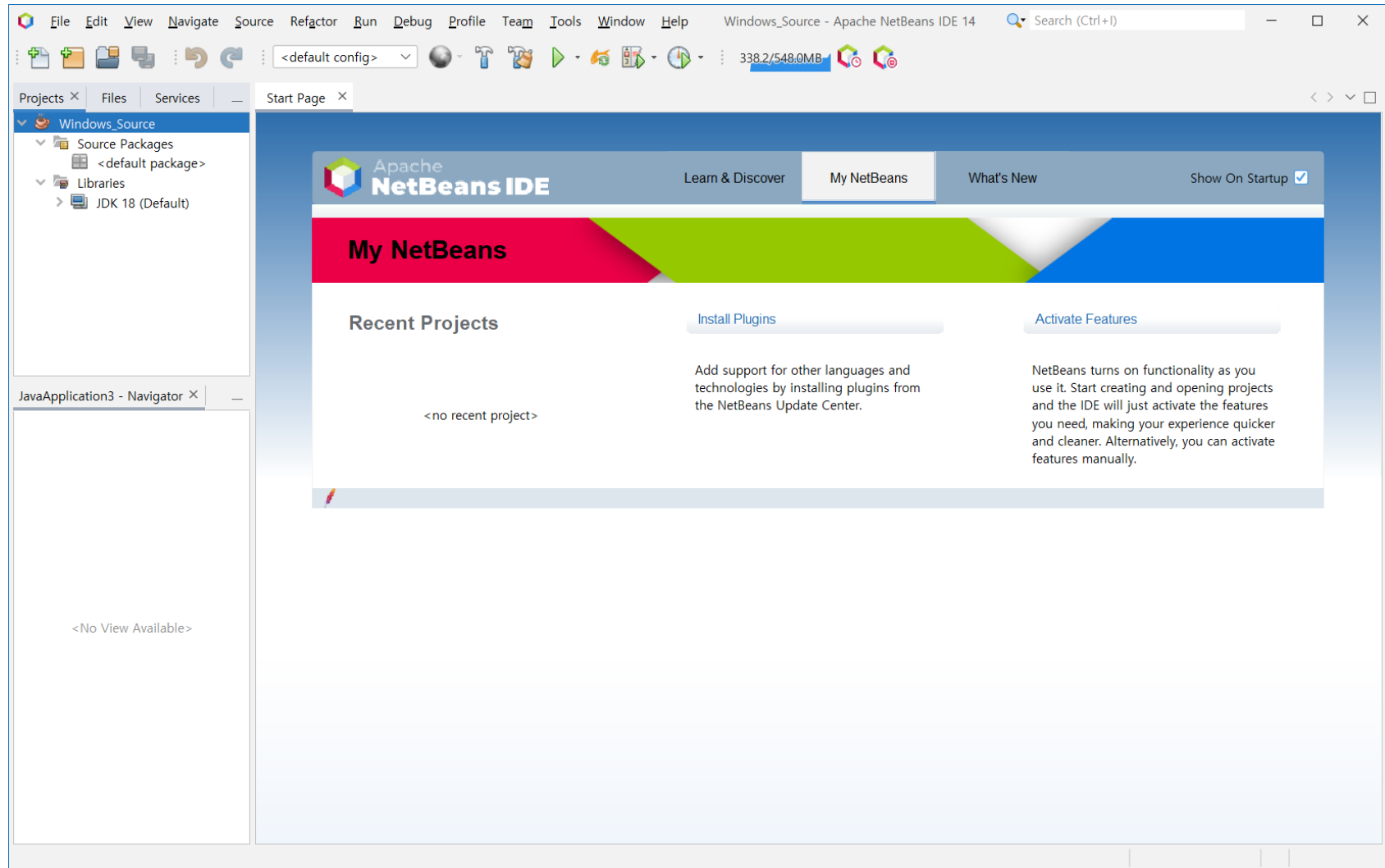
< Back Next > Finish Cancel Help





Practice 1 : Label & Button Control (4)

Create project success

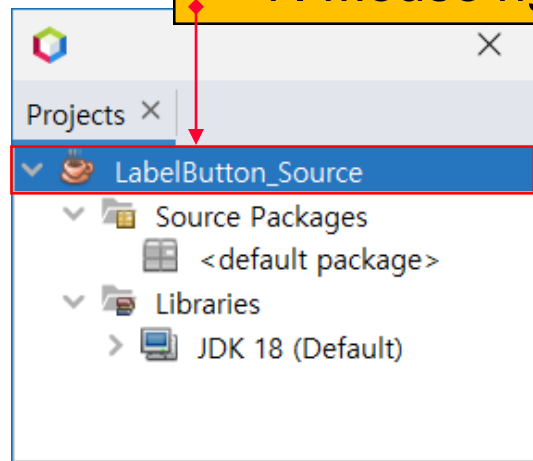




Practice 1 : Label & Button Control (5)

Create JFrame Form

7. Mouse right-button Click



New

Build
Clean and Build
Close

8. Click

Run
Debug
Profile
Test
Set Configuration

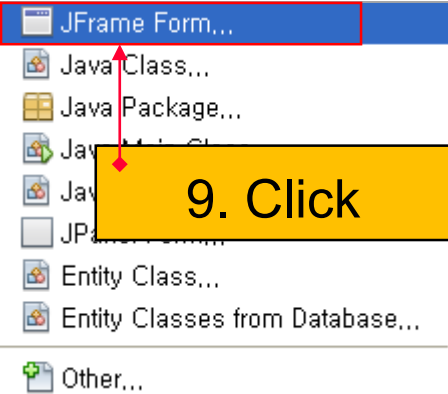
Set as Main Project
Open Required Projects
Close

Rename...
Move...
Copy...
Delete

Find...
Versioning
Local History

Properties

9. Click





Practice 1 : Label & Button Control (6)

■ Setting Class Name

❖ Mainframe.java 생성

New JFrame Form

Steps

1. Choose File Type
2. Name and Location

Name and Location

Class Name: MainFrame

Project: LabelButton_Source

Location: Source Packages

Package:

Created File: C:\Java_Project\LabelButton_Source\src\MainFrame.java

Projects

- LabelButton_Source
 - Source Packages
 - MainFrame.java
- Libraries
 - JDK 18 (Default)

10. Input "MainFrame"

11. Click

Finish



Practice 1 : Label & Button Control (7)

Create JFrame form success

The screenshot displays the Apache NetBeans IDE interface. The main editor window shows a large, empty rectangular area representing the JFrame form. A red double-headed arrow points from a green box labeled "JFrame Form" below to this area. The left sidebar contains the Projects, Files, and Services views. The right sidebar shows the Palette with various Swing controls and menus, and the Properties window for the selected component.

Projects View:

- LabelButton_Source
 - Source Packages
 - <default package>
 - MainFrame.java
 - Libraries
 - JDK 18 (Default)

Properties Window (MainFrame.java):

Properties	
Name	MainFrame
Extension	java
File Size	3228
Modification Time	2022. 8. 9. 오전 11:15:29
All Files	C:\Java_Project\Lab...
Classpaths	
Compile Classpath	
Runtime Classpath	C:\Java_Project\Lab...
Boot Classpath	nbjrtfile/C:/Program...





Practice 1 : Label & Button Control (8)

Control Layout & Property Setting

❖ Layout

◆ Label

◆ Button

12. Drag & Drop

13. Drag & Drop

Palette ×

- Swing Containers
- Swing Controls
 - Label
 - Check Box
 - Combo Box
 - Text Area
 - Progress Bar
 - Spinner
 - Editor Pane
 - OK Button
 - Radio Button
 - List
 - Scroll Bar
 - Formatted Field
 - Separator
 - Tree
 - Toggle Button
 - Button Group
 - Text Field
 - Slider
 - Password Field
 - Text Pane
 - Table





Practice 1 : Label & Button Control (9)

❖ MainFrame.java Source Code

```
MainFrame.java - Editor
MainFrame.java x
Source Design History
1
2 public class MainFrame extends javax.swing.JFrame {
3
4     public MainFrame() {
5         initComponents();
6     }
7
8     @SuppressWarnings("unchecked")
9     Generated Code
47
48 public static void main(String args[]) {
49
50     /* Create and display the form */
51     java.awt.EventQueue.invokeLater(new Runnable() {
52         public void run() {
53             new MainFrame().setVisible(true);
54         }
55     });
56 }
57
58 // Variables declaration - do not modify
59 private javax.swing.JButton jButton1;
60 private javax.swing.JLabel jLabel1;
61 // End of variables declaration
62 }
```

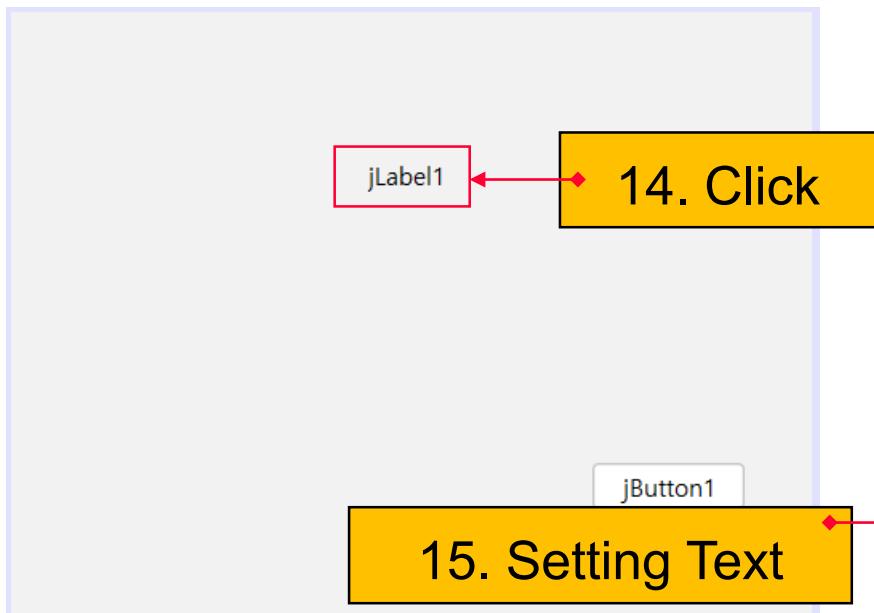
Label , Button 변수 선언





Practice 1 : Label & Button Control (10)

❖ Label 속성 지정



Control	Property Setting
Label	•Text: 프로그램 종료

Swing Properties Window for JLabel1 [JLabel]

Properties | Events | Code

Properties

background	<input type="checkbox"/> [242,242,242]	...
displayedMnemonic		...
font	맑은 고딕 12 Plain	...
foreground	<input checked="" type="checkbox"/> [0,0,0]	...
horizontalAlignment	LEADING	...
icon	<none>	...
labelFor	<none>	...
text	프로그램 종료	...
toolTipText		...
verticalAlignment	CENTER	...

Other Properties

text

(java.lang.String) Defines the single line of text this component will display.



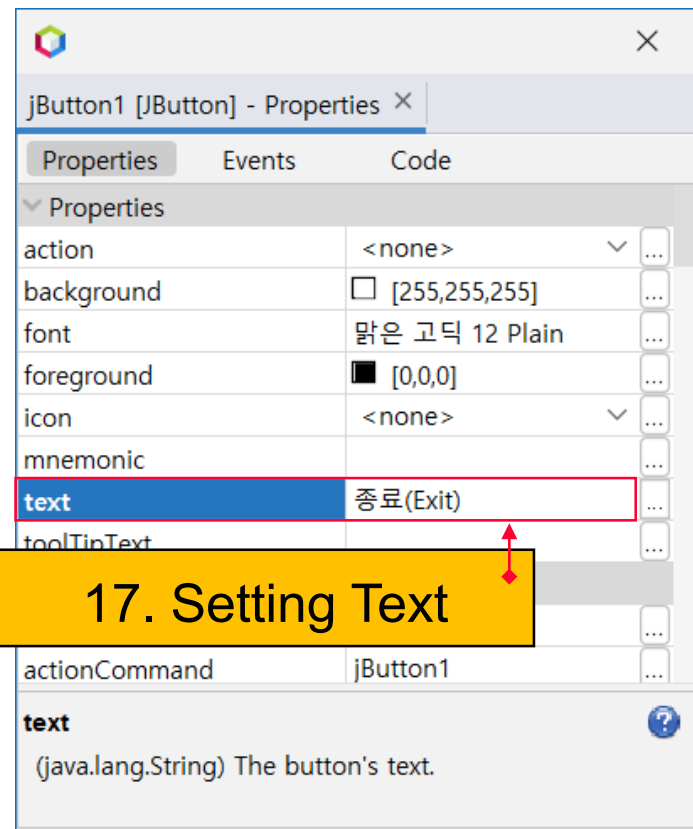


Practice 1 : Label & Button Control (11)

❖ Button 속성 지정



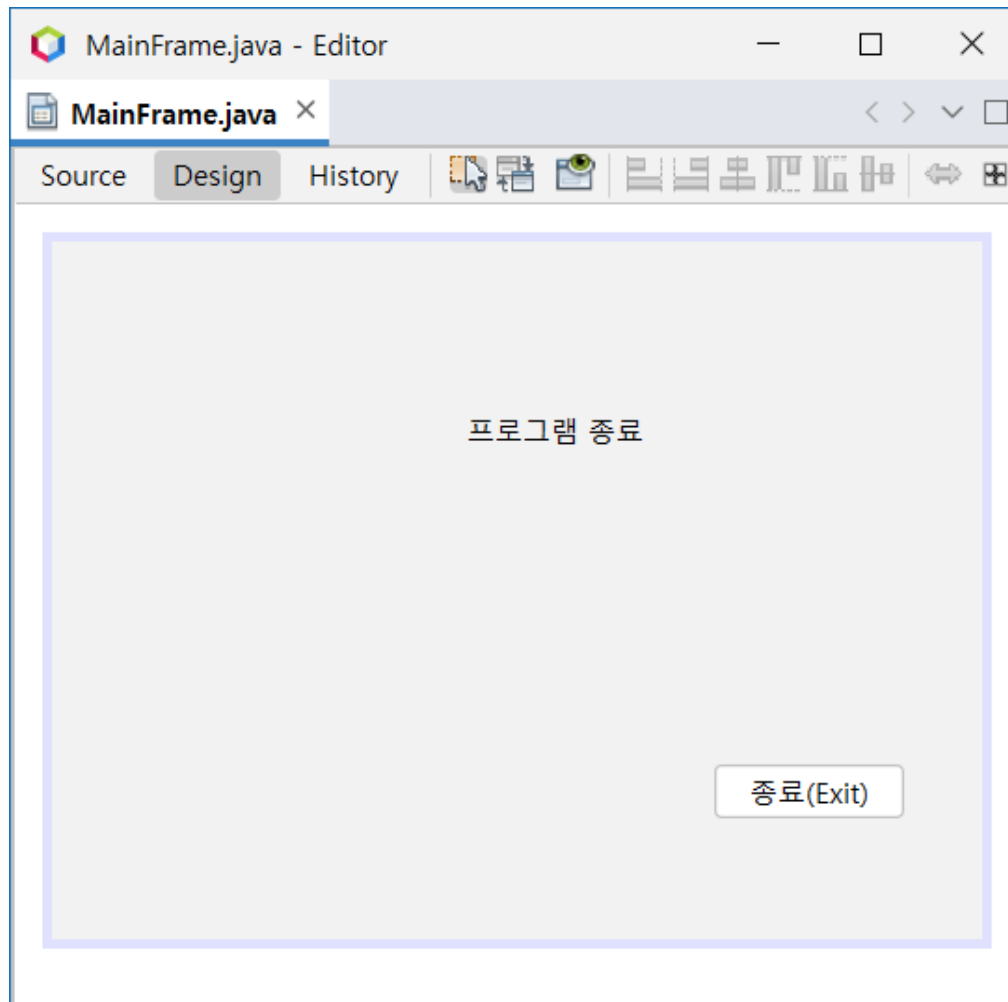
Control	Property Setting
Button	•Text: 종료(Exit)





Practice 1 : Label & Button Control (12)

■ GUI 구현 완료





Practice 1 : Label & Button Control (13)

■ [종료(Exit)] Button Event Handler 구현

```
MainFrame.java - Editor
MainFrame.java x
Source Design History
1 public class MainFrame extends javax.swing.JFrame {
2
3
4 public MainFrame() {...3 lines }
7
8 @SuppressWarnings("unchecked")
9 Generated Code
51
52 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
53 // TODO add your handling code here:
54 System.exit(0); //Application Exit
55 }
56
57 public static void main(String args[]) {...9 lines }
66
67 // Variables declaration - do not modify
68 private javax.swing.JButton jButton1;
69 private javax.swing.JLabel jLabel1;
70 // End of variables declaration
71 }
```

프로그램 종료

18. Double Click

종료(Exit)

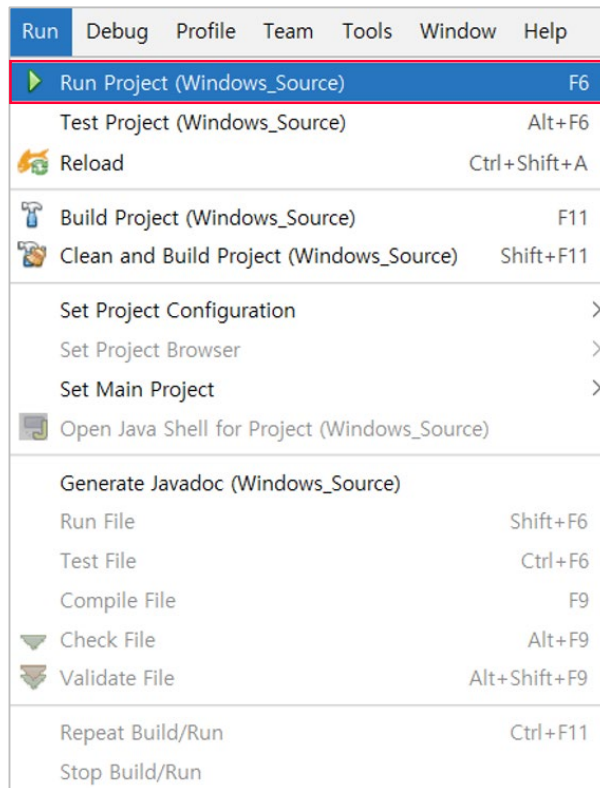
19. Coding



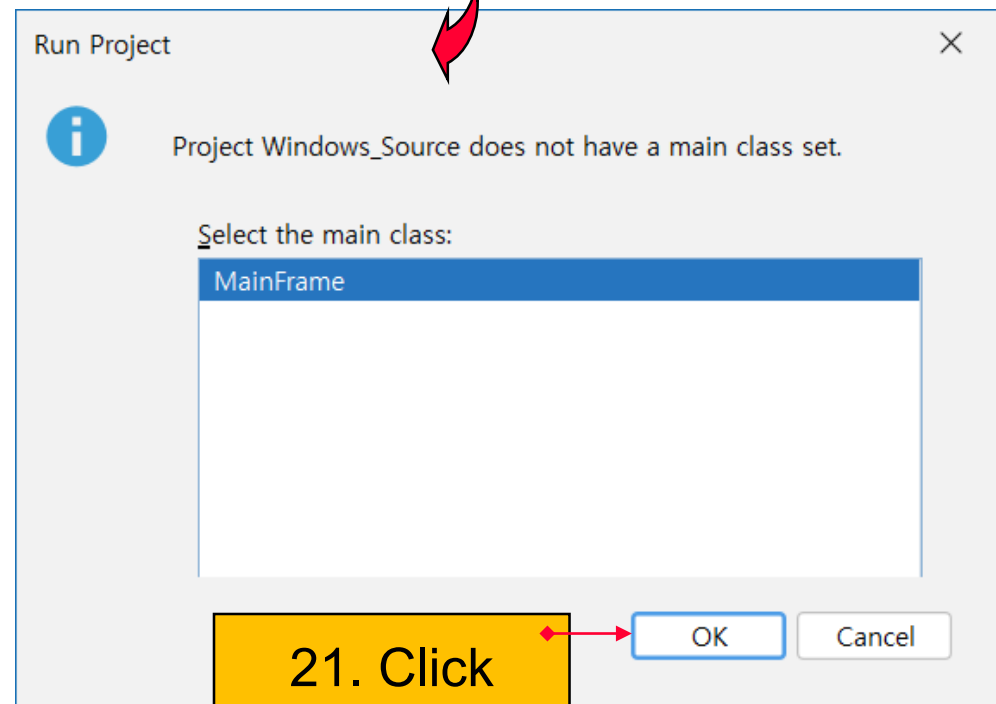


Practice 1 : Label & Button Control (14)

Run



20. Click





Practice 1 : Label & Button Control (15)

❖ 종료(Exit) Button Click

◆ 프로그램 종료





Text Field Control

■ TextField Control

❖ 기능

◆ 단일 문자열 입력

❖ 형식

```
TextField strInput = new TextField(str, size)
```

str	Textfield에 표시될 기본 문자열
size	Textfield 크기

❖ Method

Method	Function
public String getText()	입력된 문자열 반환
void setText(String str)	문자열 str을 TextField에 지정
public String getSelectedText()	선택된 문자열 반환
public void setEditable(boolean b)	TextField에 입력 여부 결정(true/false)





Text Area Control

■ TextArea Control

❖ 기능

◆ 다중 문자열 입력

❖ 형식

```
TextArea strInput = new TextArea(str, rows, cols)
```

str	TextArea에 표시될 기본 문자열
rows	Row
cols	Column

❖ Method

Method	Function
void append(String str)	기존의 문자열에 str 문자열 추가
int getColumns	Column 수 반환
int getRows	Row 수 반환





Practice 2 : Text Field & Text Area Control

❖ Project Name: TextFieldTextArea_Source (Time: 20 min)

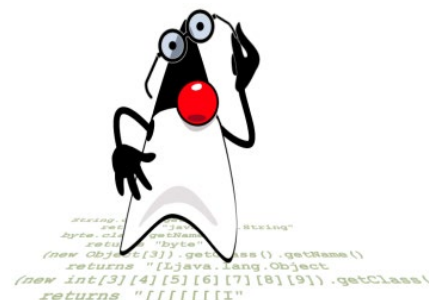
- 한 줄 입력 TextField에 문자열 입력
- 입력(Input) Button Click
- 다중 라인 입력 TextArea의 문자열에 한 줄 입력 TextField의 문자열을 추가 한다.

TextField 및 TextArea 예제

한 줄 입력 재미있다.

입력 (Input)

다중 라인 입력 Java
재미있다.





Practice 2 : Text Field & Text Area Control (1)

Create Project

The screenshot illustrates the process of creating a new project in the Apache NetBeans IDE. The interface includes a menu bar at the top with options like File, Edit, View, and Run. A red box highlights the 'New Project...' option in the File menu, with a yellow callout box labeled '1. Click' pointing to it. Below the menu, the 'New Project' dialog is open. It features a 'Steps' section on the left with '1. Choose Project' and '2. ...'. The 'Choose Project' section has a 'Filter' text field and two lists: 'Categories' and 'Projects'. In the 'Categories' list, 'Java with Ant' is selected and highlighted with a red box. In the 'Projects' list, 'Java Application' is selected and highlighted with a red box. A yellow callout box labeled '2. Click' points to the 'Java Application' selection. At the bottom of the dialog, there is a 'Description' section and a 'Next >' button. A yellow callout box labeled '3. Click' points to the 'Next >' button. The bottom of the IDE window shows an 'Output' tab.

1. Click

2. Click

3. Click





Practice 2 : Text Field & Text Area Control (2)

■ Project Name and Location

❖ Project name: TextFieldTextArea_Source

New Java Application

Steps

1. Choose Project
2. **Name and Location**

Name and Location

Project Name: TextFieldTextArea_Source

Project Location: C:\Java_Project Browse...

Project Folder: C:\Java_Project\TextFieldTextArea_Source

☐ Use Dedicated Folder for Storing Libraries

Libraries Folder: Browse...

Different users and projects can share the same compilation libraries (see Help for details).

☐ Create Main Class textfieldtextarea_source.TextFieldTextArea_Source

4. Input "TextFieldTextArea_Source"

5. Reset check

6. Click

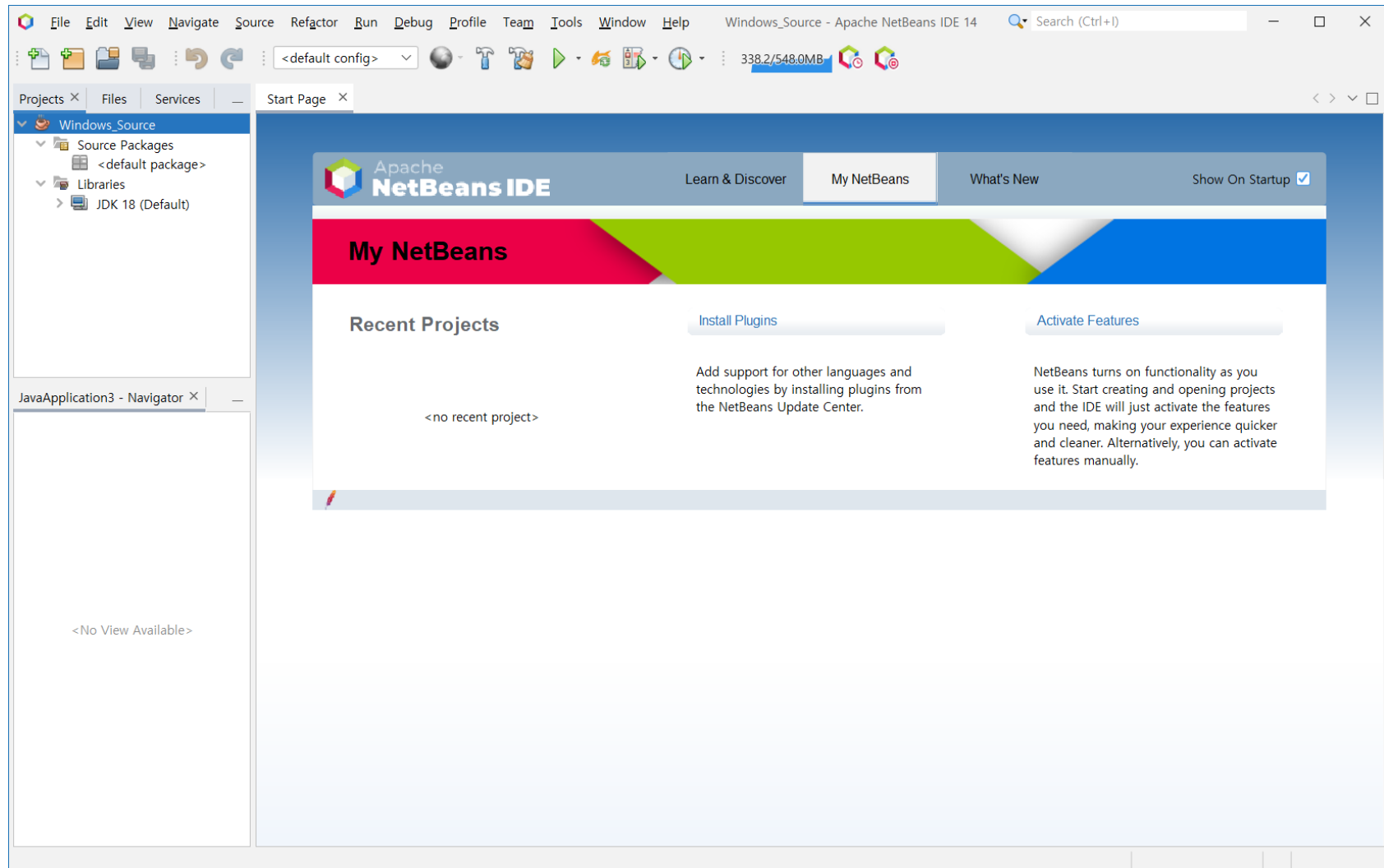
< Back Next > Finish Cancel Help





Practice 2 : Text Field & Text Area Control (3)

Create project success





Practice 2 : Text Field & Text Area Control (4)

■ Create JFrame Form

7. Mouse right-button Click

8. Click

9. Click

Projects X

- TextFieldTextArea_Source
 - Source Packages
 - <default package>
 - Libraries
 - JDK 18 (Default)

New

- Build
- Clean and Build
- Close
- Run
- Debug
- Profile
- Test
- Set Configuration
- Set as Main Project
- Open Required Projects
- Close
- Rename...
- Move...
- Copy...
- Delete
- Find...
- Versioning
- Local History
- Properties

JFrame Form...

- Java Class...
- Java Package...
- Java Module...
- Java Project...
- JSP...
- Entity Class...
- Entity Classes from Database...
- Other...





Practice 2 : Text Field & Text Area Control (5)

■ Setting Class Name

❖ Mainframe.java 생성

New JFrame Form

Steps

1. Choose File Type
2. Name and Location

Name and Location

Class Name: MainFrame

Project: TextFieldTextArea_Source

Location: Source Packages

Package:

Created File: C:\Java_Project\TextFieldTextArea_Source\src\MainFrame.java

Projects

- LabelButton_Source
 - Source Packages
 - <default package>
 - MainFrame.java
- Libraries
 - JDK 18 (Default)

10. Input "MainFrame"

11. Click

Warning: It is highly recommended that you do not place Java classes in the default package

< Back Next > Finish Cancel Help



Practice 2 : Text Field & Text Area Control (6)

Create JFrame form success

The screenshot displays the Apache NetBeans IDE interface. The central design view shows a large, empty rectangular frame representing the JFrame. A green box with the text "JFrame Form" and a red double-headed arrow points to this frame. The left sidebar shows the project structure with "MainFrame.java" selected. The bottom-left pane shows the class hierarchy for "MainFrame : JFrame". The right sidebar contains the "Swing Controls" palette and the "Properties" window for "MainFrame.java".

Swing Controls

- Label
- Toggle Button
- Radio Button
- Combo Box
- Text Field
- Scroll Bar
- Progress Bar
- Password Field
- Separator
- Editor Pane
- Table
- Button
- Check Box
- Button Group
- List
- Text Area
- Slider
- Formatted Field
- Spinner
- Text Pane
- Tree

Swing Menus

- Menu Bar
- Menu
- Menu Item
- Menu Item / CheckBox
- Menu Item / RadioButton
- Popup Menu
- Separator

MainFrame.java - Properties

Properties	
Name	MainFrame
Extension	java
File Size	3228
Modification Time	2022. 8. 9. 오후 12:22:15
All Files	C:\Java_Project\WText...
Classpaths	
Compile Classpath	...
Runtime Classpath	C:\Java_Project\WText...
Boot Classpath	nbjrtfile/C:/Program...

MainFrame.java





Practice 2 : Text Field & Text Area Control (7)

Control Layout & Setting Property

TextField 및 TextArea 예제

한 줄 입력

다중 라인 입력

입력(Input)

Control	Properties Setting
jLabel1	• Text: TextField 및 TextArea 예제
jLabel2	• Text: 한 줄 입력
jLabel3	• Text: 다중 입력
jTextField1	• Text:
jTextArea1	• Text:
jButton1	• Text: 입력(Input)





Practice 2 : Text Field & Text Area Control (8)

❖ 입력(Input) Button Event Handler

```
MainFrame.java - Editor
MainFrame.java x
Source Design History
1 public class MainFrame extends javax.swing.JFrame {
2
3     public MainFrame() {
4         initComponents();
5     }
6
7     @SuppressWarnings("unchecked")
8     Generated Code
84
86     private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
87         String strCRLF = "\n"; //CR + LF(\n)
88         String strData = null; //String variable
89         strData = jTextArea1.getText(); //TextArea -> strData
90         strData += jTextField1.getText() + strCRLF; //TextField -> strData
91         jTextArea1.setText(strData); //sreData -> TextArea
92
93     }
94
95     public static void main(String args[]) {...8 lines }
103
```

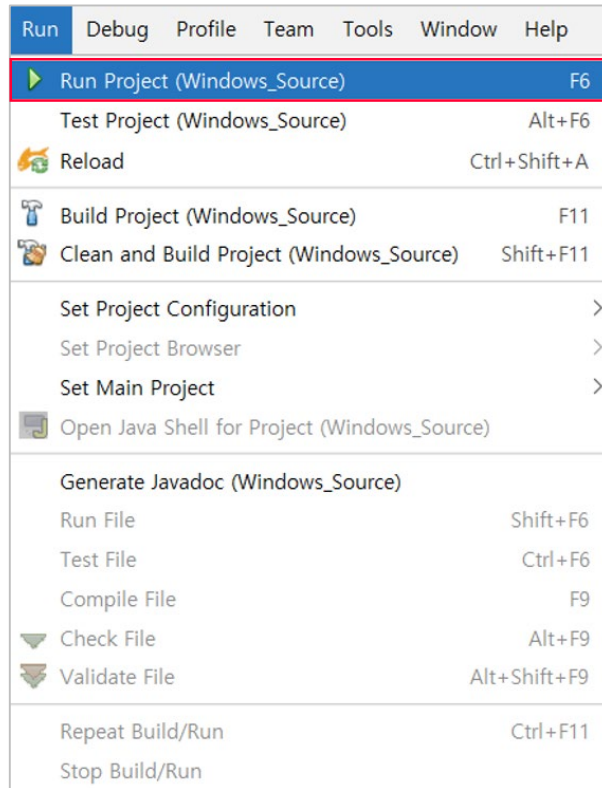
12. Coding



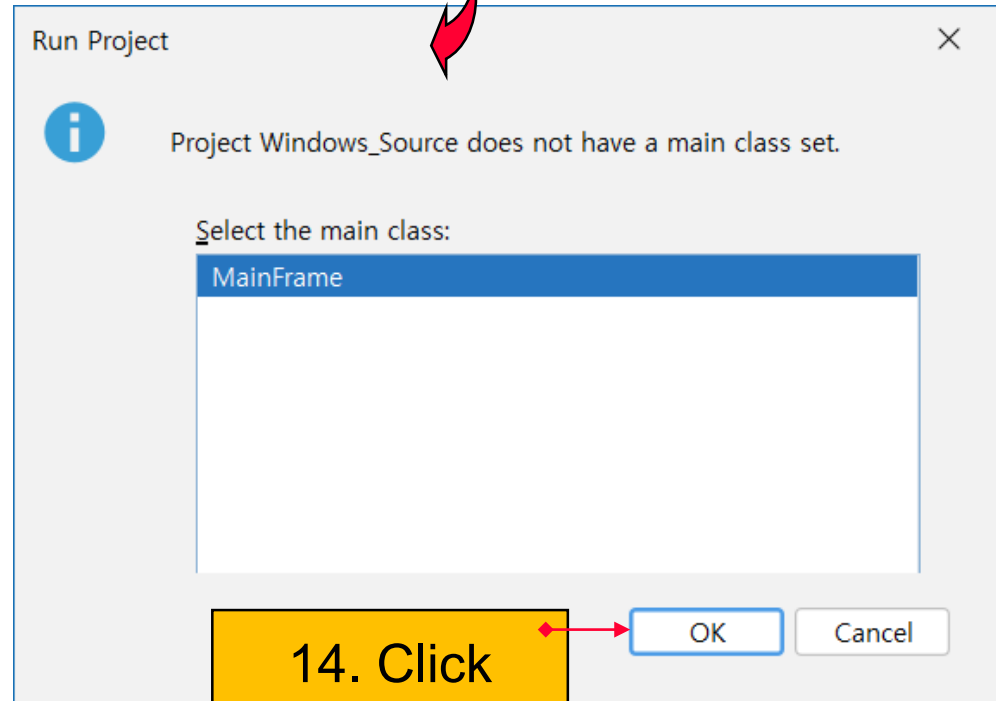


Practice 2 : Text Field & Text Area Control (9)

Run



13. Click





Practice 2 : Text Field & Text Area Control (10)

■ 실행 결과

❖ 입력(Input) Button Click

◆ 한 줄 입력 문자열 -> 다중 라인 입력 문자열에 추가

TextField 및 TextArea 예제

한 줄 입력

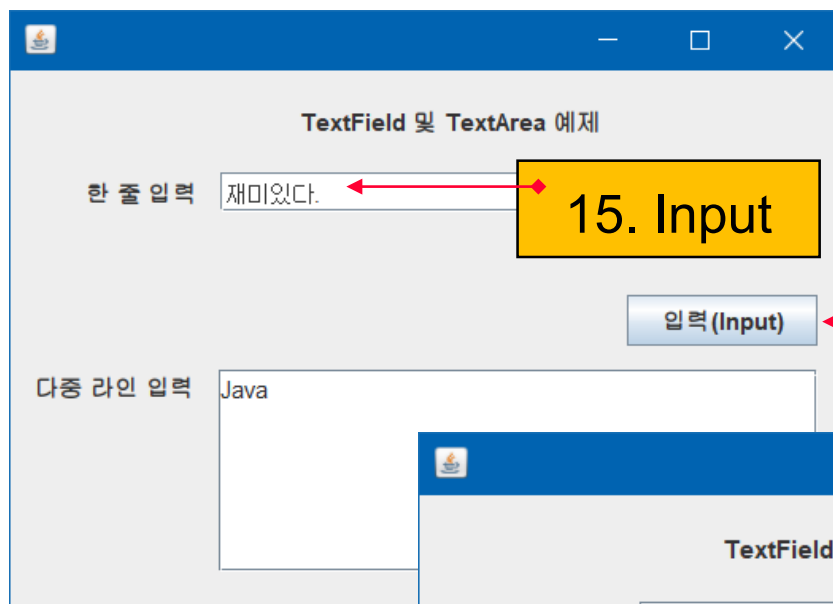
다중 라인 입력

입력 (Input)

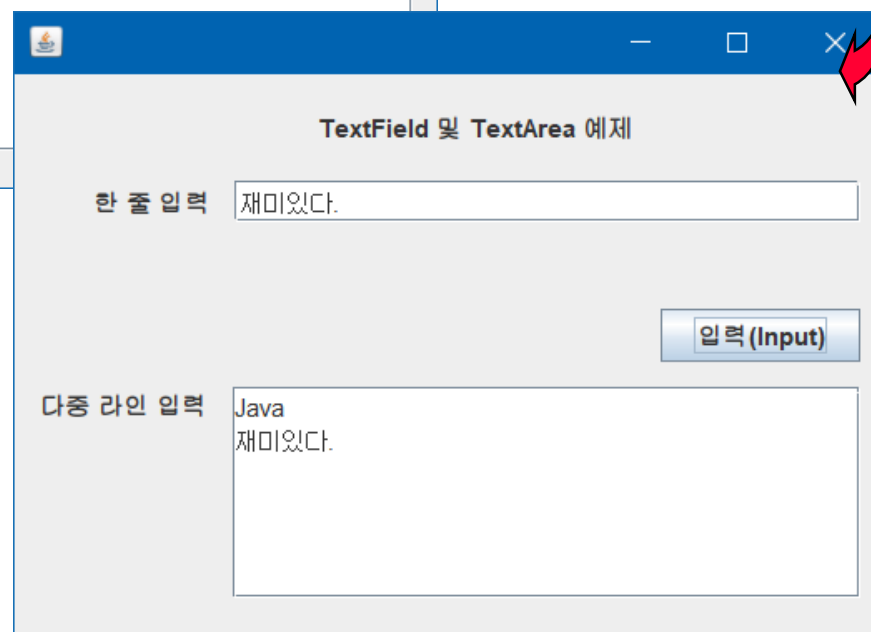




Practice 2 : Text Field & Text Area Control (11)



16. Click





Practice 3 : Text Field & Text Area Control

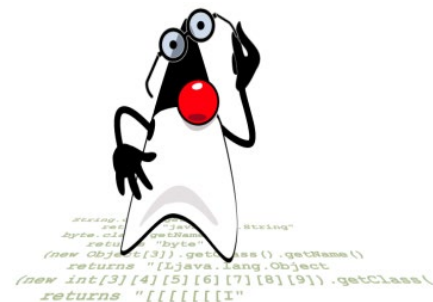
❖ Project Name: TextAreaAppend_Source (Time: 20 min)

- 다중 라인 입력 TextArea의 문자열에 한 줄 입력 TextField의 문자열을 추가 한다.
 - append() method를 이용하여 동일한 기능을 하도록 구현하시오.

TextField 및 TextArea 예제

한 줄 입력

다중 라인 입력





Practice 3 : Text Field & Text Area Control

❖ 입력(Input) Button Event Handler

```
MainFrame.java - Editor
MainFrame.java x
Source Design History
1 public class MainFrame extends javax.swing.JFrame {
2
3     public MainFrame() {
4         initComponents();
5     }
6
7     @SuppressWarnings("unchecked")
8     Generated Code
84
86     private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
87         String strCRLF = "\n"; //CR + LF(\n)
88         String strData = null; //String variable
89         strData = jTextField1.getText() + strCRLF; //TextField -> strData
90         jTextArea1.append(strData); //sreData -> TextArea
91
92     }
93
94     public static void main(String args[]) { ...8 lines }
102
```

Coding

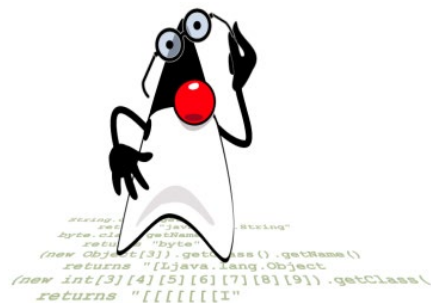




Practice 4 : Key Event Handler

❖ Project Name: KeyEvent_Source(Time: 20 min)

- 문자 입력에 하나의 문자 입력
 - 입력 문자 -> TextArea에 추가
- 초기화(Reset) Button Click
 - TextField 문자열 초기화
 - TextArea 문자열 초기화



Key Event 예제

문자 입력

입력 확인

Inha
Technical



Practice 4 : Key Event Handler (1)

Create Project

The screenshot shows the Apache NetBeans IDE interface. The 'File' menu is open, and 'New Project...' is highlighted. A yellow box labeled '1. Click' points to this menu item. The 'New Project' dialog is open, showing the 'Choose Project' tab. The 'Projects' list on the right contains 'Java Application', which is highlighted. A yellow box labeled '2. Click' points to this item. At the bottom of the dialog, the 'Next >' button is highlighted, with a yellow box labeled '3. Click' pointing to it. The 'Description' section at the bottom of the dialog states: 'Creates a new Java SE application in a standard IDE project. You can also generate a main class in the project. Standard projects use an IDE-generated Ant build script to build, run, and debug your project.'

1. Click

2. Click

3. Click





Practice 4 : Key Event Handler (2)

■ Project Name and Location

❖ Project name: KeyEvent_Source

New Java Application

Steps

1. Choose Project
2. **Name and Location**

Name and Location

Project Name: KeyEvent_Source

Project Location: C:\Java_Project Browse...

Project Folder: C:\Java_Project\KeyEvent_Source

☐ Use Dedicated Folder for Storing Libraries

Libraries Folder: Browse...

Different users and projects can share the same compilation libraries (see Help for details).

☒ Create Main Class keyevent_source.KeyEvent_Source

< Back Next > **Finish** Cancel Help

4. Input "KeyEvent_Source"

5. Reset check

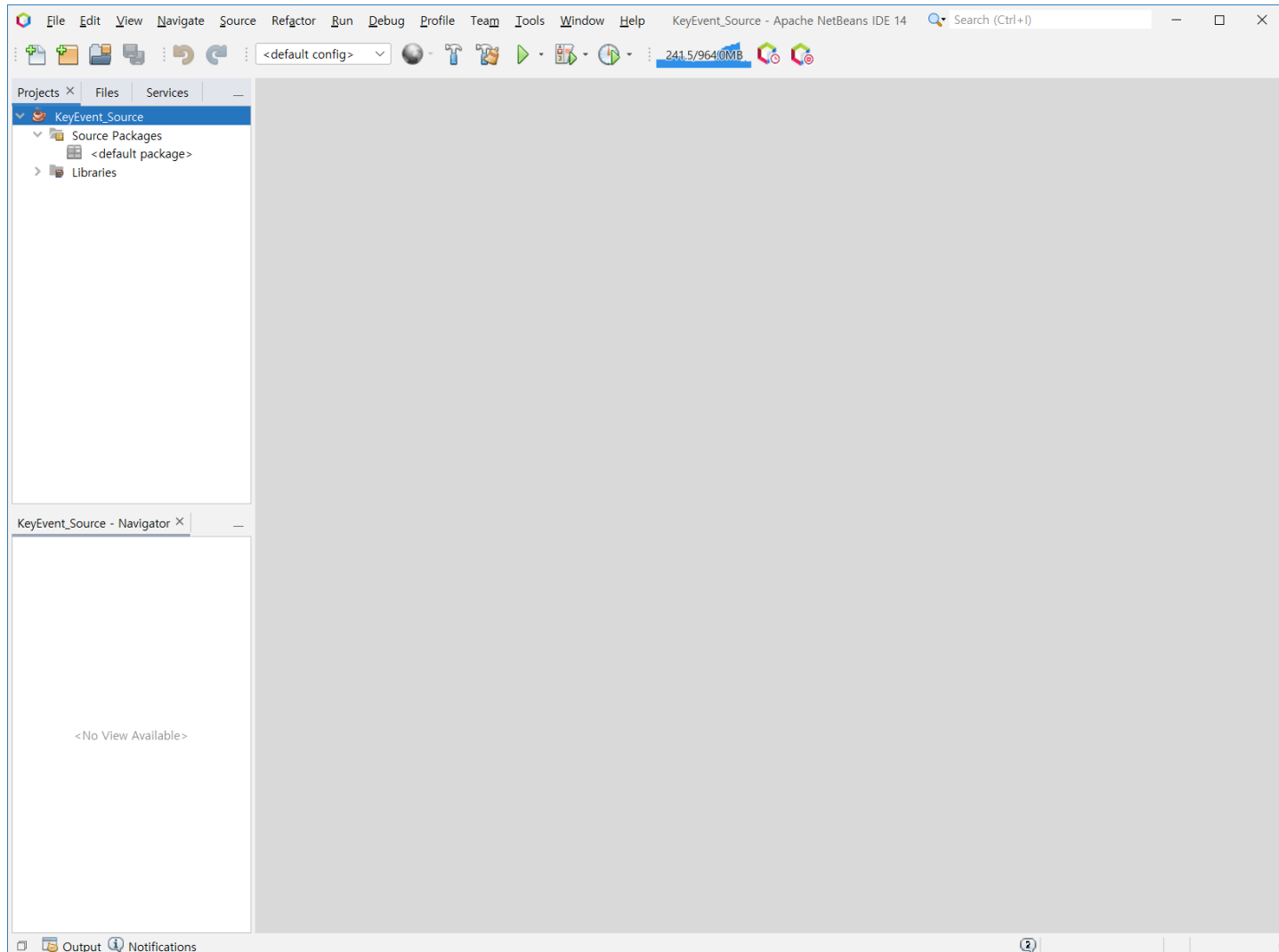
6. Click





Practice 4 : Key Event Handler (3)

Create project success





Practice 4 : Key Event Handler (4)

Create JFrame Form

7. Mouse right-button Click

8. Click

9. Click

Projects X

- KeyEvent_Source
 - Source Packages
 - <default package>
 - Libraries

New

- Build
- Clean and Build
- Close All
- Run
- Debug
- Profile
- Test Alt+F6
- Set Configuration
- Set as Main Project
- Open Required Projects
- Close
- Rename...
- Move...
- Copy...
- Delete Delete
- Find... Ctrl+F
- Versioning
- Local History
- Properties

JFrame Form...

- Java Class...
- Java Package...
- Java Main Class...
- Java Class from Database...
- JPA Entity Class...
- Entity Class...
- Entity Classes from Database...
- Other...





Practice 4 : Key Event Handler (5)

■ Setting Class Name

❖ Mainframe.java 생성

New JFrame Form

Steps

1. Choose File Type
2. Name and Location

Name and Location

Class Name: MainFrame

Project: KeyEvent_Source

Location: Source Packages

Package:

Created File: C:\Java_Project\KeyEvent_Source\src\MainFrame.java

Projects

- KeyEvent_Source
 - Source Packages
 - MainFrame.java
 - <default package>
 - Libraries

10. Input "MainFrame"

11. Click

It is highly recommended that you do not place Java classes in the default package

< Back Next > Finish Cancel Help



Practice 4 : Key Event Handler (6)

Create JFrame form success

The screenshot displays the Apache NetBeans IDE interface for a project named 'KeyEvent_Source'. The central editor window shows 'MainFrame.java' in the 'Design' tab, featuring a large, empty rectangular box representing the JFrame form. A red double-headed arrow points from this box to a green rectangular label with the text 'JFrame Form'. The left sidebar contains the 'Projects' and 'Files' views, with 'MainFrame.java' selected. The bottom-left 'Navigator' view shows the class structure: 'MainFrame : JFrame' with methods 'MainFrame()', 'initComponents()', and 'main(String[] args)'. The right sidebar includes the 'Palette' with 'Swing Containers' and 'Swing Controls' (such as Label, Button, Text Field, etc.) and the 'Properties' view for 'MainFrame.java', which lists attributes like Name, Extension, File Size, and Classpaths.





Practice 4 : Key Event Handler (7)

Control Layout & Property Setting

Key Event 예제

문자 입력

입력 확인

Control	Properties Setting
jLabel1	• Text: Key Event 예제
jLabel2	• Text: 문자 입력
jLabel3	• Text: 입력 확인
jTextField1	• Text:
jTextArea1	• Text:
jButton1	• Text: 초기화(Reset)

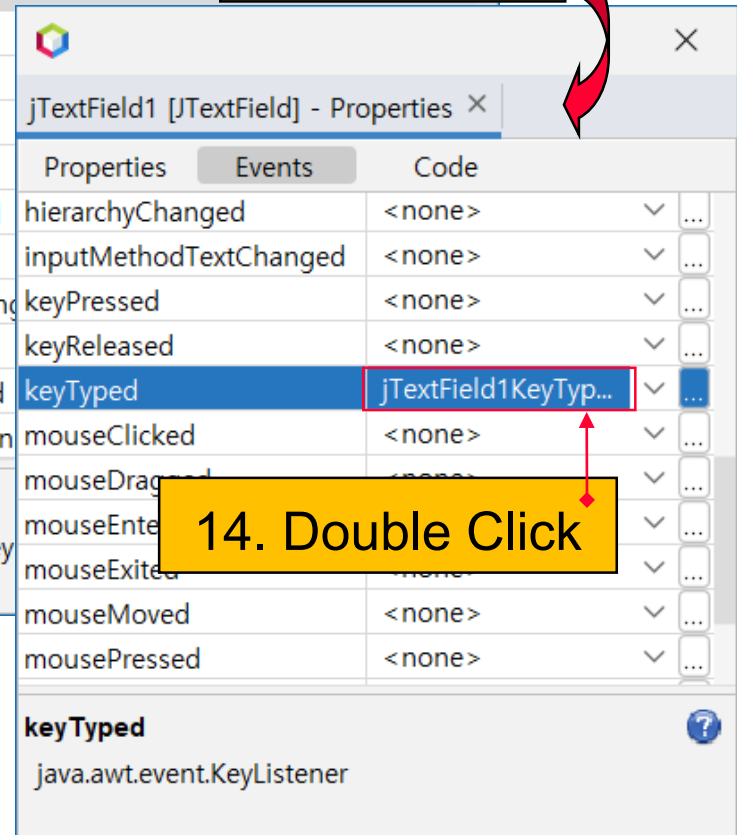
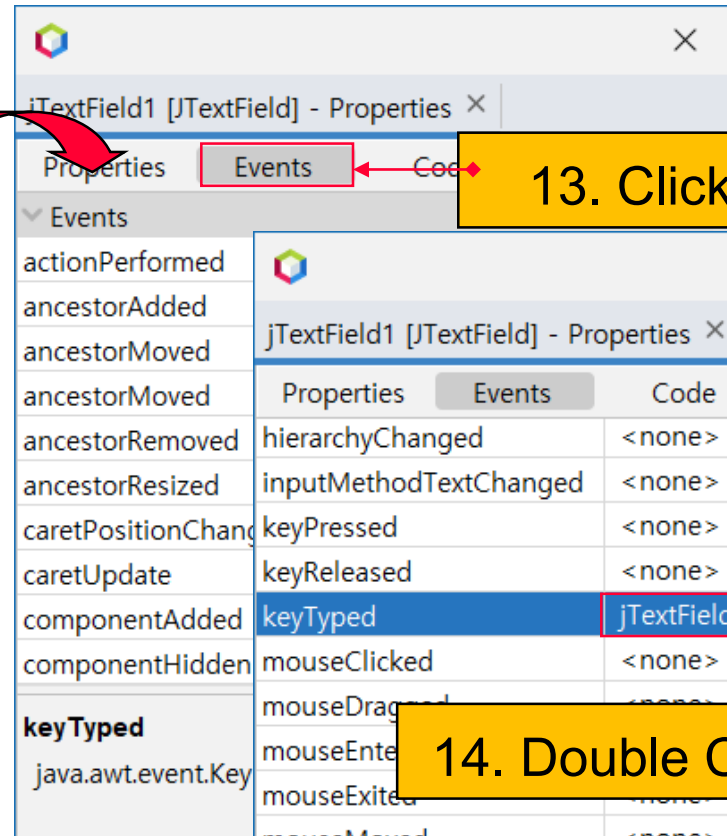
Control	Event
jTextField1	• ActionPerformed(Event evt) • KeyTyped(Event evt)
jButton1	• ActionPerformed(Event evt)





Practice 4 : Key Event Handler (8)

■ jTextField1 Event 지정





Practice 4 : Key Event Handler (9)

❖ KeyTyped Event Handler

```
MainFrame.java - Editor
MainFrame.java x
Source Design History
1
2 import java.awt.event.KeyEvent; //VK_BACK_SPACE, VK_ENTER Key
3
4 public class MainFrame extends javax.swing.JFrame {
5
6     public MainFrame() {
7         initComponents();
8     }
9
10    @SuppressWarnings("unchecked")
11    Generated Code
12
13    private void jTextField1KeyTyped(java.awt.event.KeyEvent evt) {
14        if ( evt.getKeyChar() != KeyEvent.VK_BACK_SPACE){
15            //Add input Character
16            jTextField1.setText(jTextField1.getText() + evt.getKeyChar());
17        } else {
18            if ( evt.getKeyChar() == KeyEvent.VK_ENTER) //Add newline
19                jTextField1.setText(jTextField1.getText() + evt.getKeyChar() + "\n");
20        }
21    }
22
23    public static void main(String args[]) {...8 lines }
```

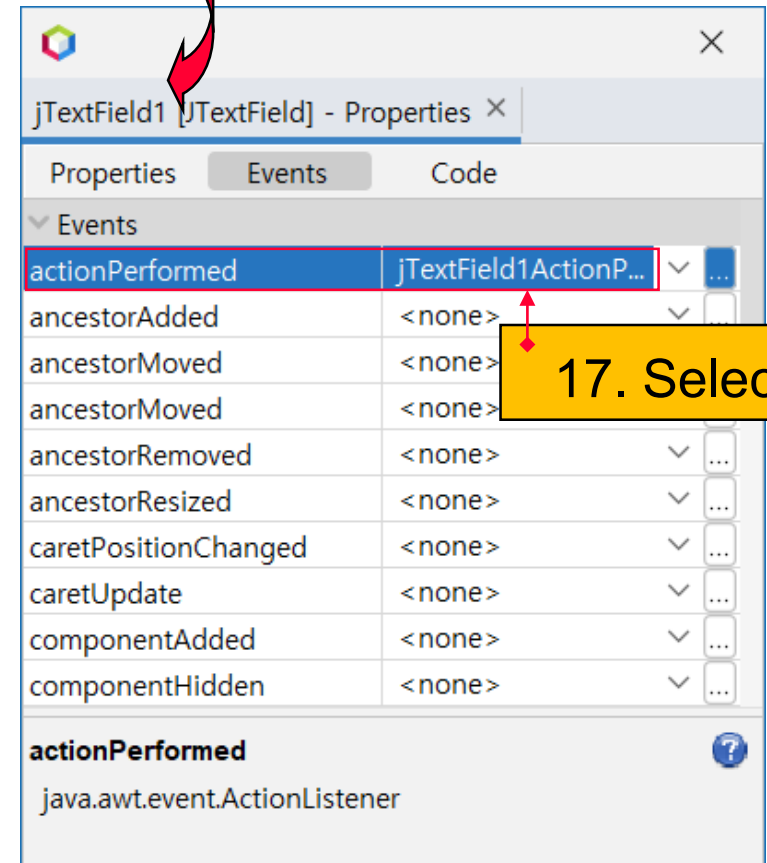
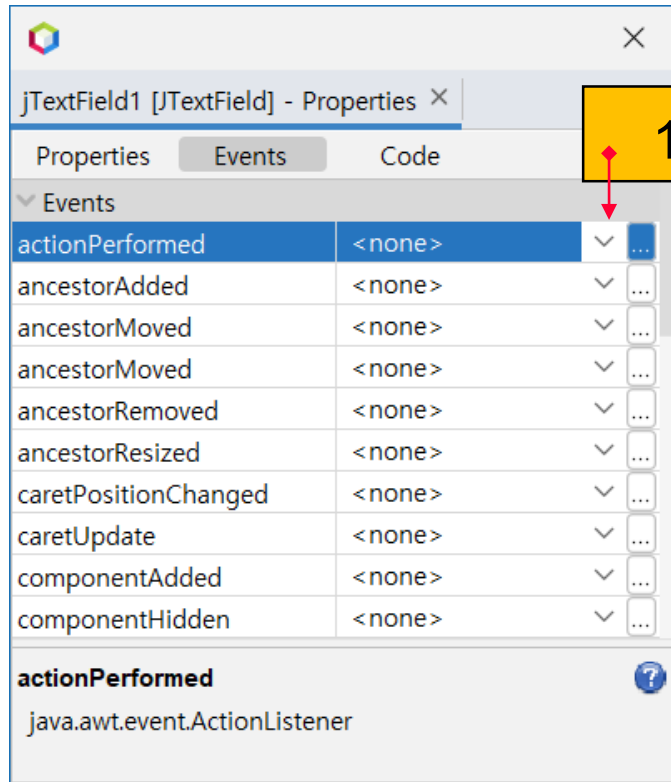
15. Coding





Practice 4 : Key Event Handler (10)

■ jTextField1 ActionPerformed Event 지정





Practice 4 : Key Event Handler (11)

❖ jTextField1 ActionPerformed Event Handler

```
1  import java.awt.event.KeyEvent; //VK_BACK_SPACE, VK_ENTER Key
2
3
4  public class MainFrame extends javax.swing.JFrame {
5
6      public MainFrame() {
7          initComponents();
8      }
9
10     @SuppressWarnings("unchecked")
11     Generated Code
12
13     private void jTextField1KeyTyped(java.awt.event.KeyEvent evt) {...9 lines }
14
15     private void jTextField1ActionPerformed(java.awt.event.ActionEvent evt) {
16         // TODO add your handling code here:
17         jTextField1.setText(null); //Input initialize
18     }
19
20     public static void main(String args[]) {...8 lines }
```





Practice 4 : Key Event Handler (12)

■ jButton1 ActionPerformed Event 처리

Key Event 예제

문자 입력

입력 확인

초기화(Reset)

19. Double Click





Practice 4 : Key Event Handler (13)

❖ JButton1 ActionPerformed Event Handler

```
MainFrame.java - Editor
MainFrame.java x
Source Design History
10 @SuppressWarnings("unchecked")
11 Generated Code
97
98 private void jTextField1KeyTyped(java.awt.event.KeyEvent evt) {
99     if ( evt.getKeyChar() != KeyEvent.VK_BACK_SPACE){
100         //Add input Character
101         jTextField1.setText(jTextField1.getText() + evt.getKeyChar());
102     } else {
103         if ( evt.getKeyChar() == KeyEvent.VK_ENTER) //Add newline
104             jTextField1.setText(jTextField1.getText() + evt.getKeyChar() + "\n");
105     }
106 }
107
108 private void jTextField1ActionPerformed(java.awt.event.ActionEvent evt) {
109     // TODO add your handling code here:
110     jTextField1.setText(null); //Input initialize
111 }
112
113 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
114     // TODO add your handling code here:
115     jTextField1.setText(null); //jTextField1 initialize
116     jTextField2.setText(null); //jTextField2 initialize
117 }
118
```

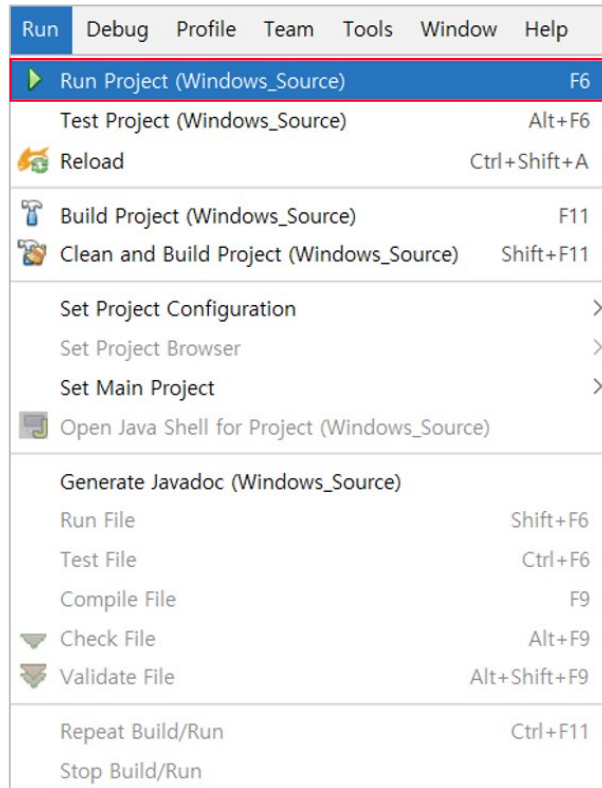
20. Coding



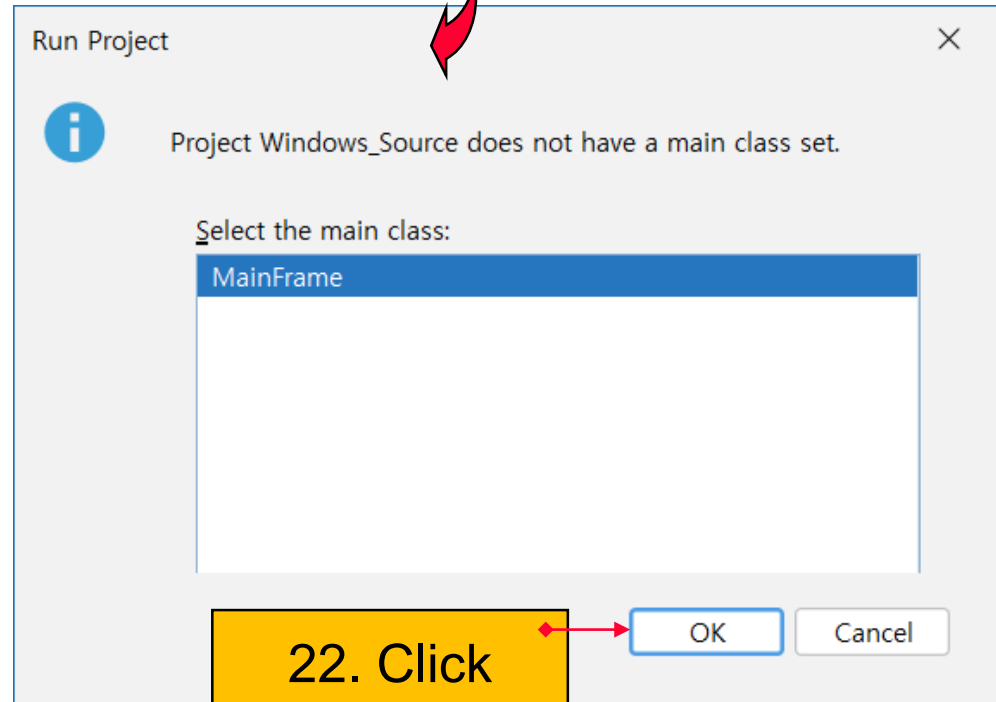


Practice 4 : Key Event Handler (14)

Run



21. Click



22. Click





Practice 4 : Key Event Handler (15)

■ Run

❖ 문자 입력

- ◆ Inha
- ◆ Technical
- ◆ College

❖ 초기화(Reset) Button Click

- ◆ TextField 초기화
- ◆ Text Are 초기화

Key Event 예제

문자 입력 College

입력 확인 Inha
Technical
College

초기화(Reset)

Key Event 예제

문자 입력

입력 확인

초기화(Reset)

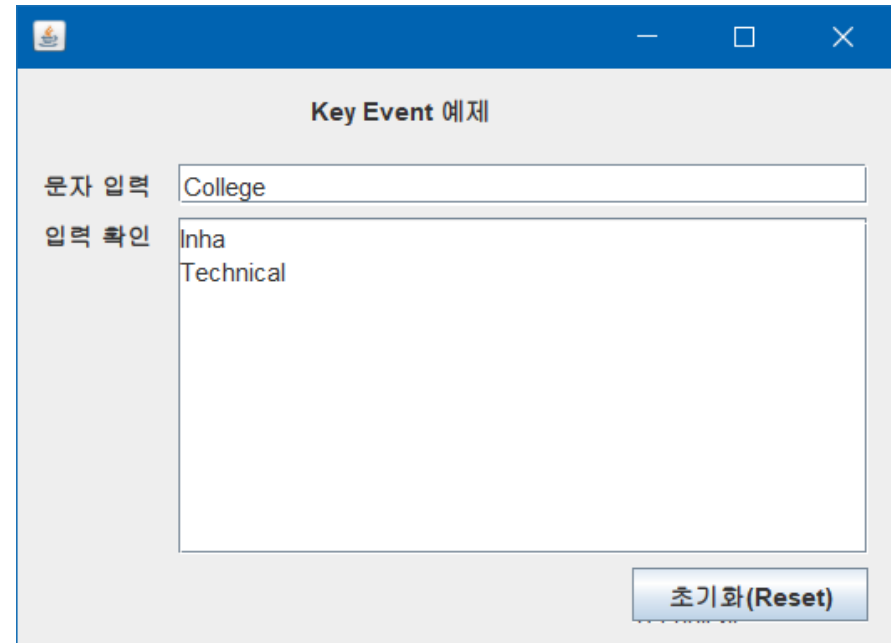
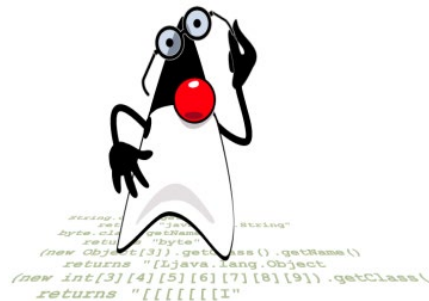




Practice 5 : Key Event Handler

❖ Project Name: KeyEvnet2_Source(Time: 20 min)

- 문자 입력에 문자열 입력 후에 [Enter] Key 누름
 - 입력 문자열 -> TextArea에 추가
- 초기화(Reset) Button Click
 - TextField 문자열 초기화
 - TextArea 문자열 초기화





Practice 5 : Key Event Handler

❖ KeyTyped Event Handler

MainFrame.java - Editor

MainFrame.java x

Source Design History

Generated Code

Coding

```
10 @SuppressWarnings("unchecked")
11
97
98 private void jTextField1KeyTyped(java.awt.event.KeyEvent evt) {
99
100     if ( evt.getKeyChar() == KeyEvent.VK_ENTER){ //Add newline
101         1
102     }
103 }
104
105
106 private void jTextField1ActionPerformed(java.awt.event.ActionEvent evt) {
107     // TODO add your handling code here:
108     //jTextField1.setText(null); //Input initialize
109 }
110
111 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
112     // TODO add your handling code here:
113     jTextField1.setText(null); //jTextField1 initialize
114     jTextArea1.setText(null); //jTextArea1 initialize
115 }
116
```

127:41/1:8 INS





Homework # 1

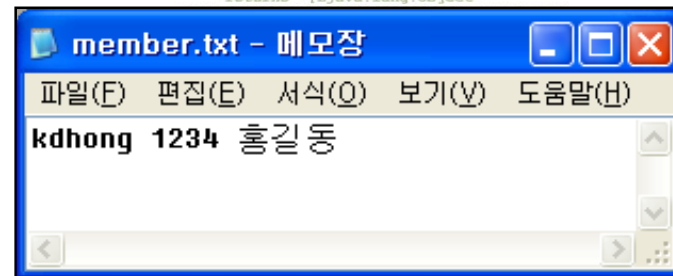
❖ 회원 가입 모듈 구현

- Project 명: Member_ID
- [중복 확인] Button Click
 - member.txt 파일에서 회원정보를 read하여 중복되는 "아이디"가 존재하는지를 검색하여 결과를 출력한다.
- [회원 가입] Button Click
 - 비밀번호가 일치하면 아이디, 비밀번호, 이름을 member.txt 파일에 저장한다.

회원 가입

아이디	kdhong	중복 확인
비밀번호	****	
비밀번호 확인	****	
성명	홍길동	

회원 가입



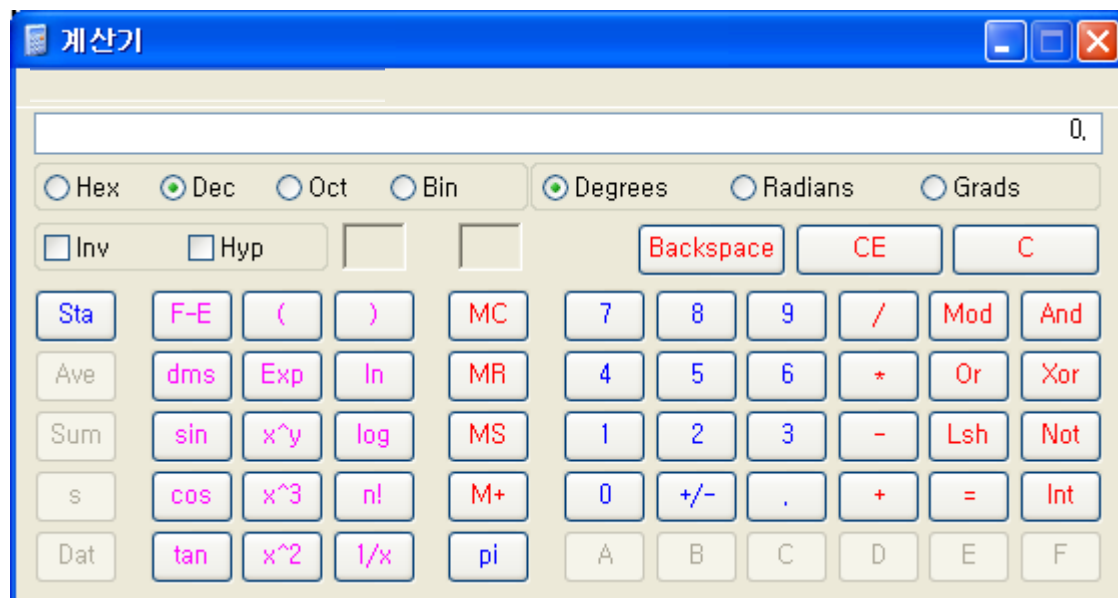
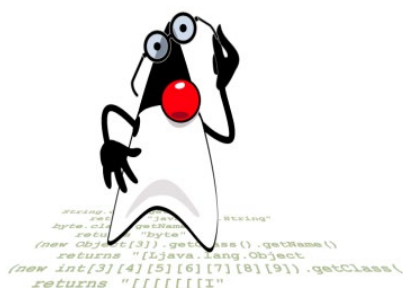


Homework # 2

❖ 계산기 구현

■ Project 명: Calculator_Source

- 일반용
- 공학용





학습 요약

- ❖ Label Control
- ❖ Button Control
- ❖ Text Field Control
- ❖ Text Area Control

