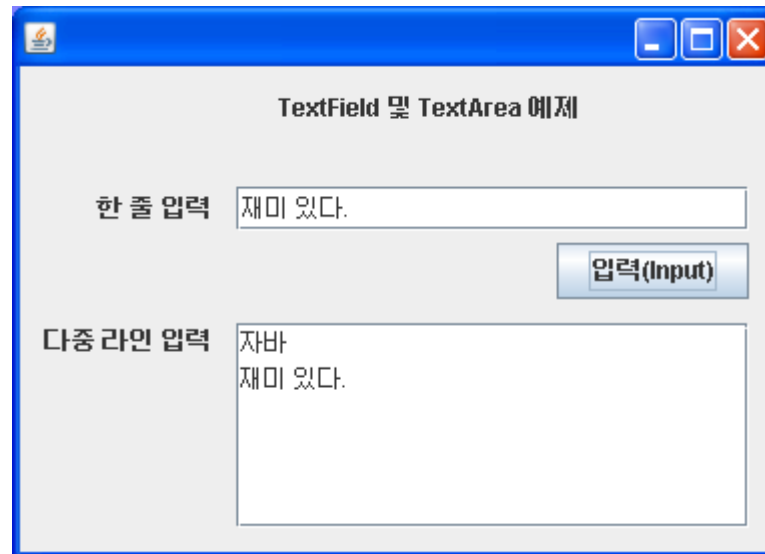




JAVA



GUI 컨트롤 I





학습 목표

■ 이 강의를 마치면 학생들은

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





Label 컨트롤 (1)

■ Label 컨트롤

❖ 기능

◆ 고정 문자열을 표시할 때 사용

❖ 생성자

생성자	기능
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 컨트롤 (2)

❖ Method

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





Button 컨트롤

■ Button 컨트롤

❖ 기능

◆ 버튼을 생성할 때 사용

❖ 생성자

생성자	기능
Button()	제목없는 버튼 생성
Button(String label)	Label로 주어진 문자열 제목의 버튼 생성

❖ Method

Method	기능
String getLabel()	버튼 제목 return
void setLabel(String label)	버튼 제목 변경

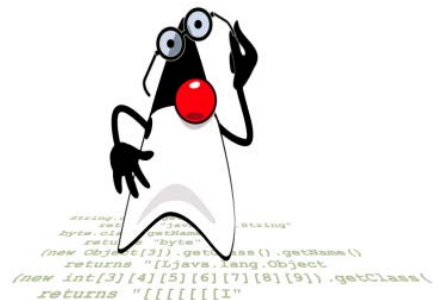
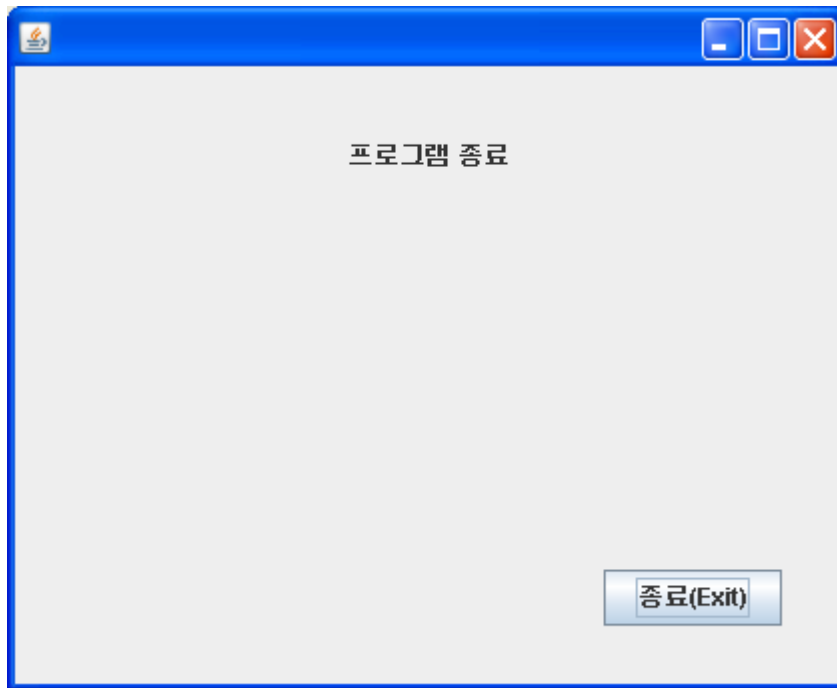




실습 : Label & Button 컨트롤

❖ Project Name: LabelButton_Source (실습시간 : 20분)

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





실습: Label & Button 컨트롤 (1)

Application 생성

The image shows the NetBeans IDE interface with three numbered steps for creating a new application:

- 1. Click**: A red arrow points to the 'New Project...' option in the 'File' menu.
- 2. Click**: A red arrow points to the 'Java Application' option in the 'Projects' list of the 'New Project' dialog.
- 3. Click**: A red arrow points to the 'Next >' button at the bottom of the 'New Project' dialog.

The 'New Project' dialog shows the following details:

- Steps**:
 1. Choose Project
 2. ...
- Choose Project**:
 - Filter: [Empty]
 - Categories: Java, JavaFX, Maven, NetBeans Modules, Samples
 - Projects: Java Application (selected), Java Class Library, Java Project with Existing Sources, Java Free-Form Project
- Description**:

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.
- Buttons**: < Back, Next > (highlighted), Finish, Cancel, Help





실습: Label & Button 컨트롤 (2)

■ Project Name and Location

❖ Project name: LabelButton_Source

4. LabelButton_Source 입력

Steps

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

Name and Location

Project Name: LabelButton_Source

Project Location: C:\Java_Projects\Source

Project Folder: C:\Java_Projects\Source\LabelButton_Source

☐ Use Dedicated Folder for Storing Libraries

Libraries Folder:

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

☐ Create Main Class labelbutton_source,LabelButton_Source

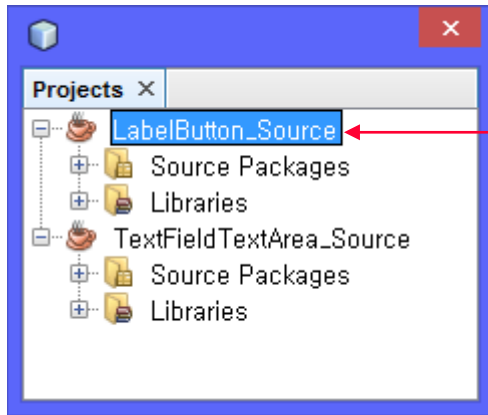
5. Click



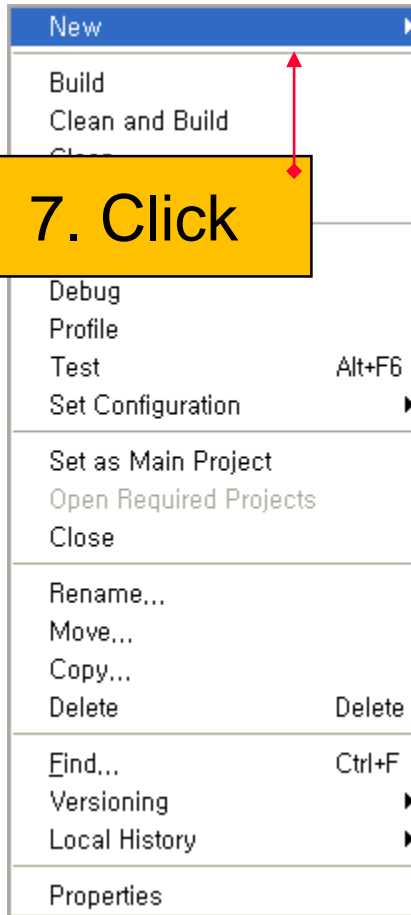


실습: Label & Button 컨트롤 (3)

JFrame Form 생성

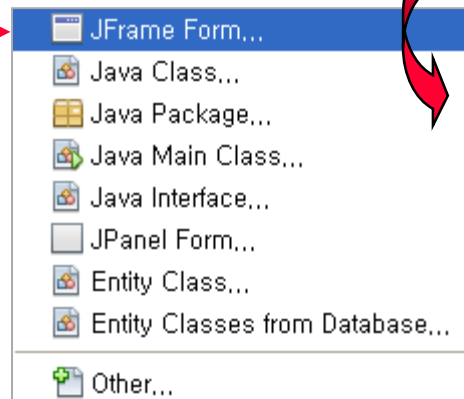


6. 마우스 오른쪽 버튼 Click



7. Click

8. Click





실습: Label & Button 컨트롤 (4)

JFrame Form Name 지정

❖ 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_Projects\Source\LabelButton_Source\src\MainFrame.java

9. MainFrame 입력

10. Click

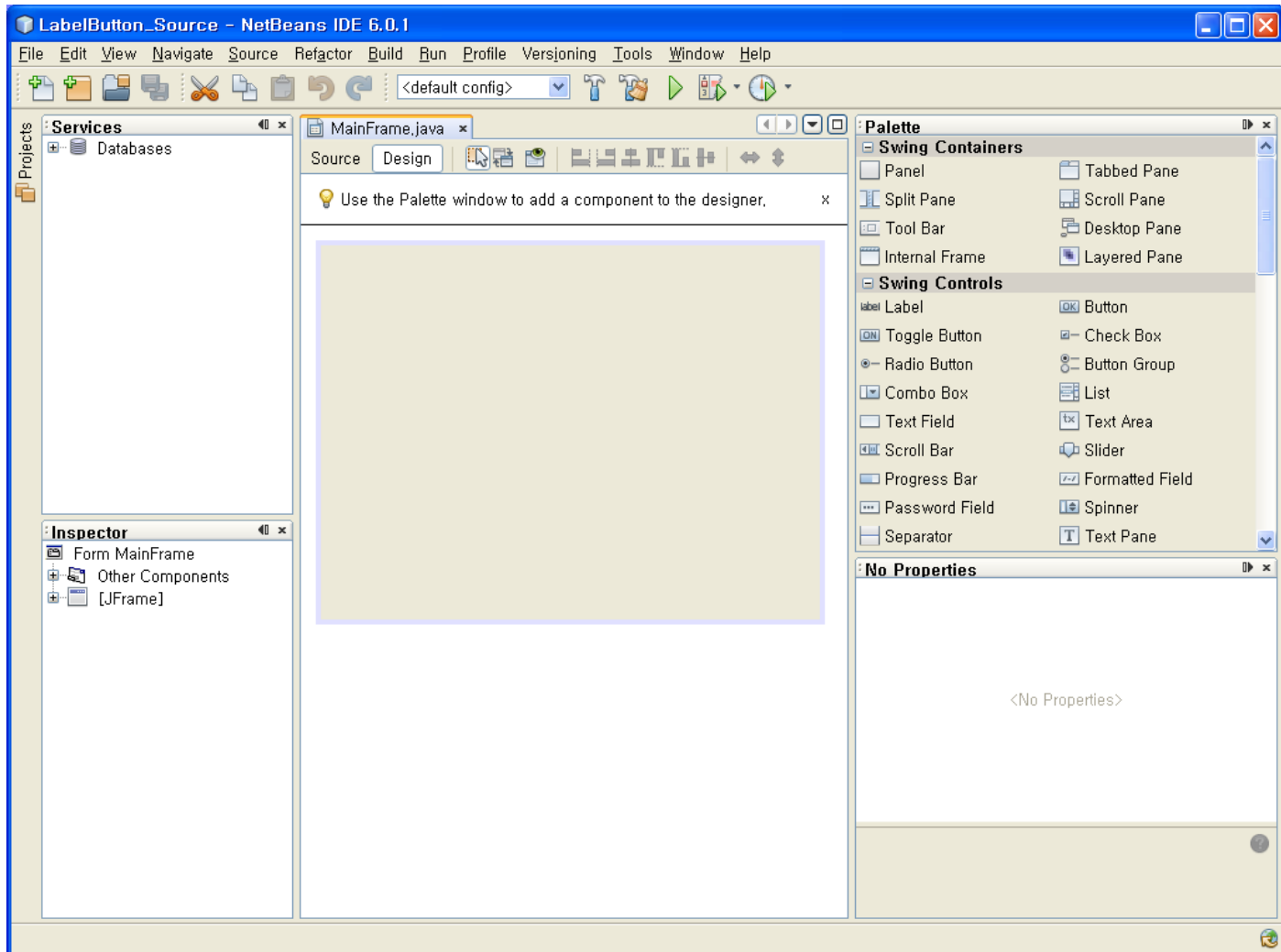
Warning: It is highly recommended that you do not place Java classes in the default package.

< Back Next > Finish Cancel Help



실습: Label & Button 컨트롤 (5)

■ LabelButton_Source Project 생성 완료

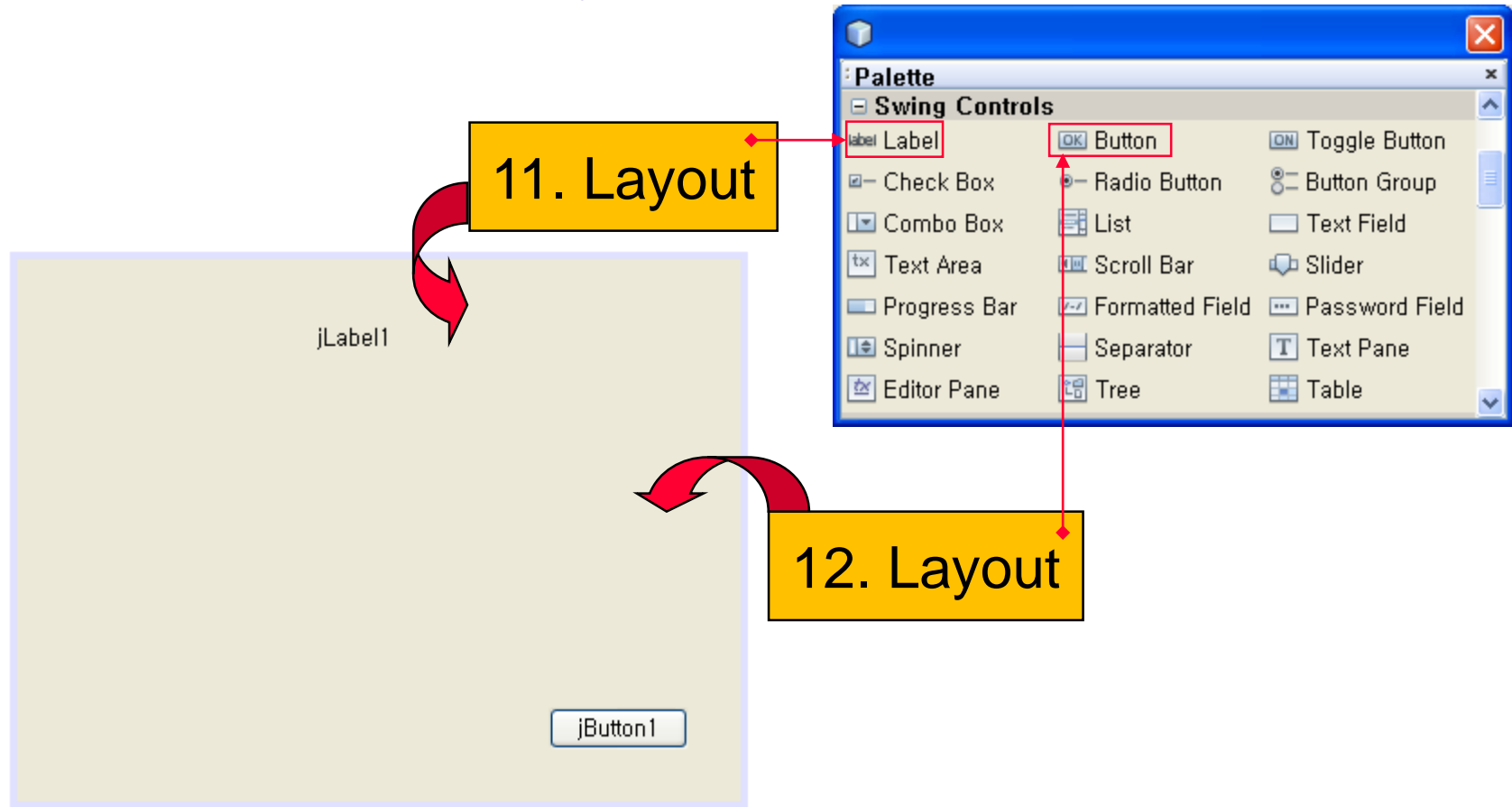




실습: Label & Button 컨트롤 (6)

■ GUI 구현

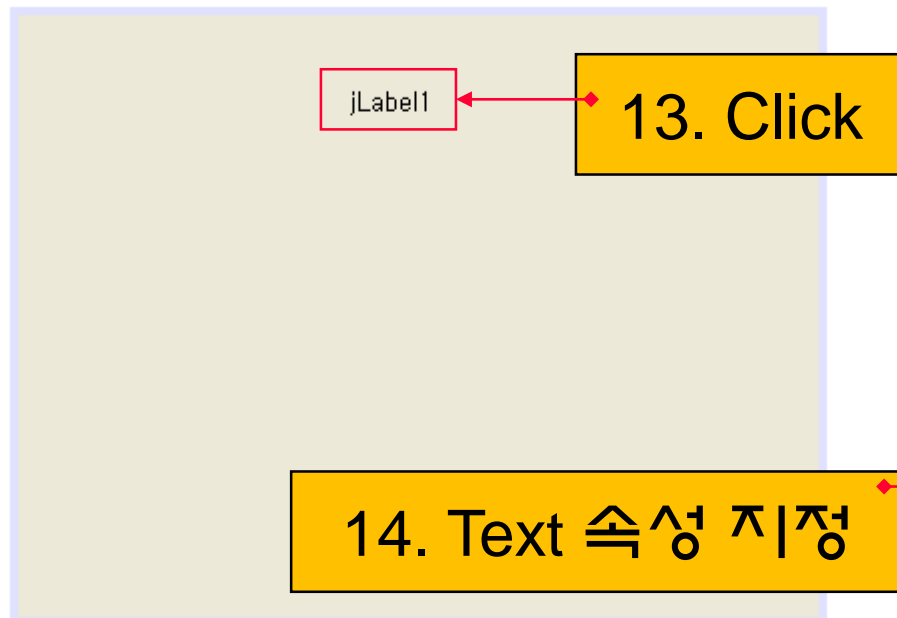
❖ Label 및 Button Layout



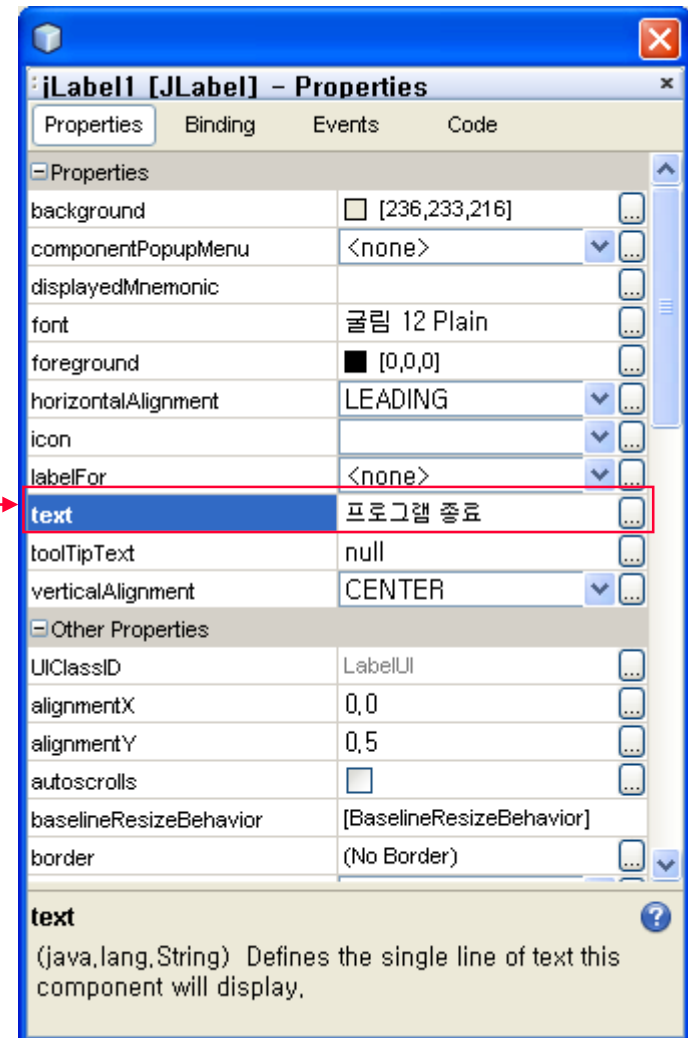


실습: Label & Button 컨트롤 (7)

❖ Label 속성 지정



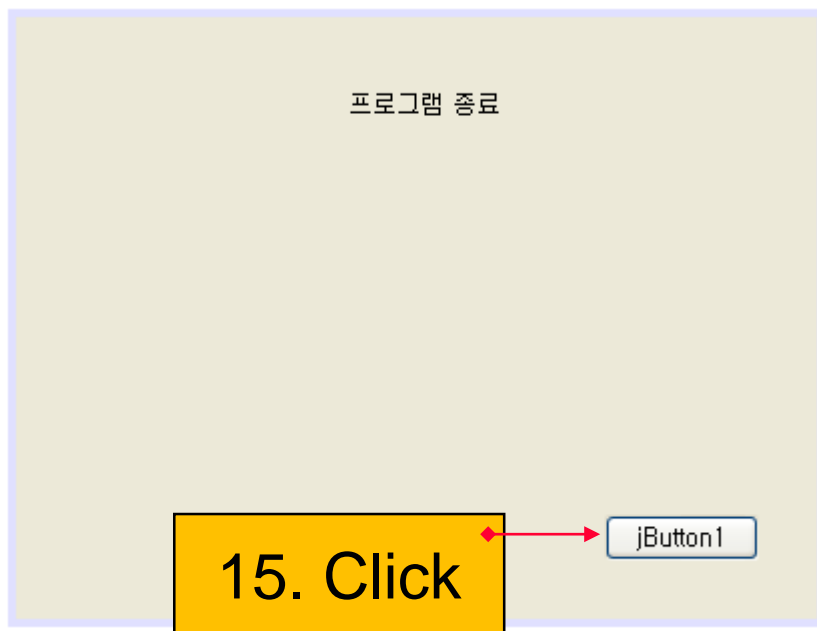
컨트롤	속성지정
Label	•Text: 프로그램 종료



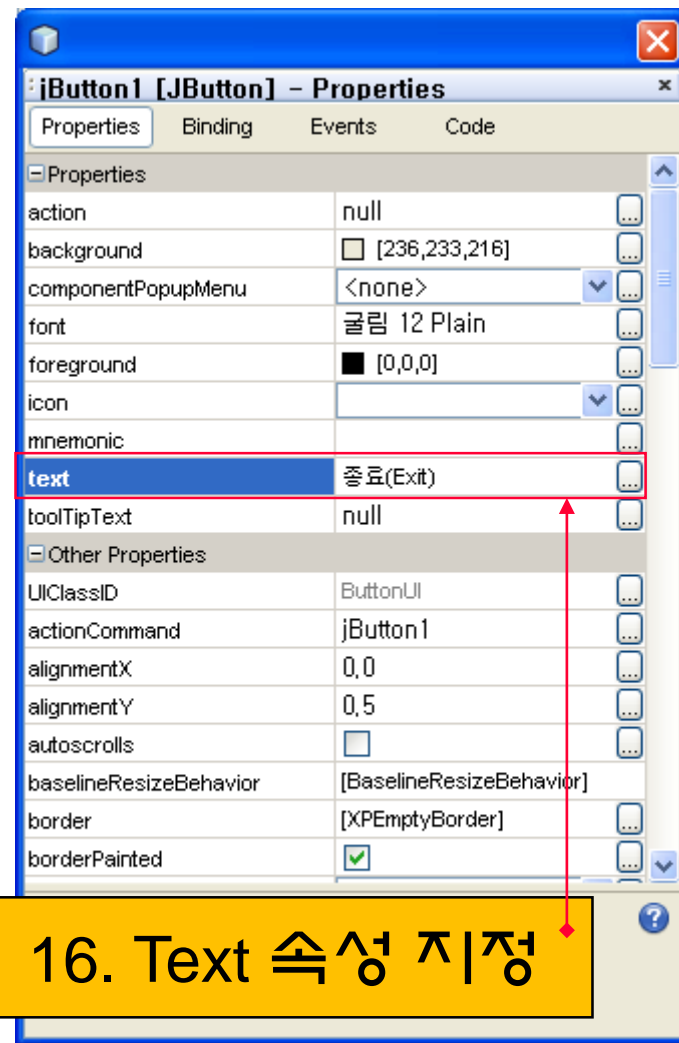


실습: Label & Button 컨트롤 (8)

❖ Button 속성 지정



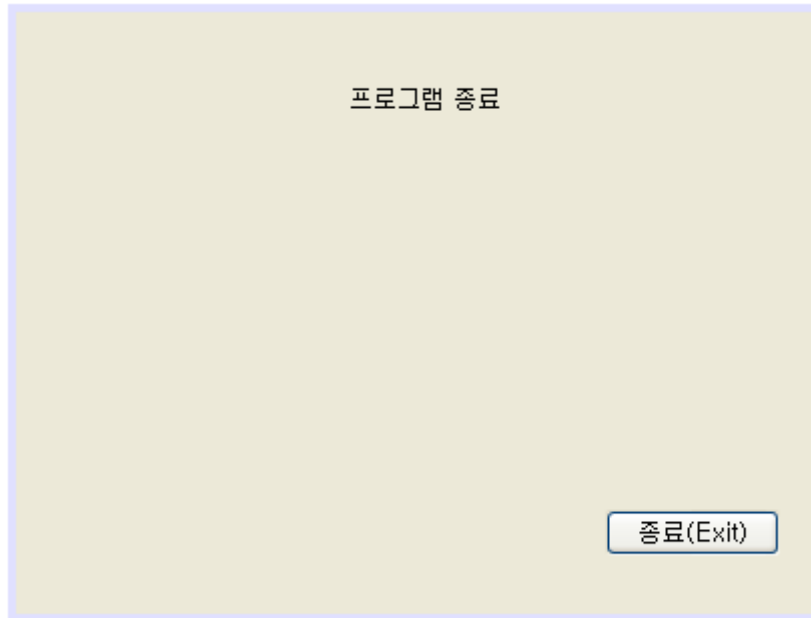
컨트롤	속성지정
Button	•Text: 종료(Exit)





실습: Label & Button 컨트롤 (9)

■ GUI 구현 완료





실습: Label & Button 컨트롤 (10)

❖ Label 및 Button Layout에 따른 Source Coding

```
72  /**
73   * @param args the command line arguments
74   */
75  public static void main(String args[]) {
76      java.awt.EventQueue.invokeLater(new Runnable() {
77          public void run() {
78              new MainFrame().setVisible(true);
79          }
80      });
81  }
82
83  // Variables declaration - do not modify
84  private javax.swing.JButton jButton1;
85  private javax.swing.JLabel jLabel1;
86  // End of variables declaration
87
88
89  }
```

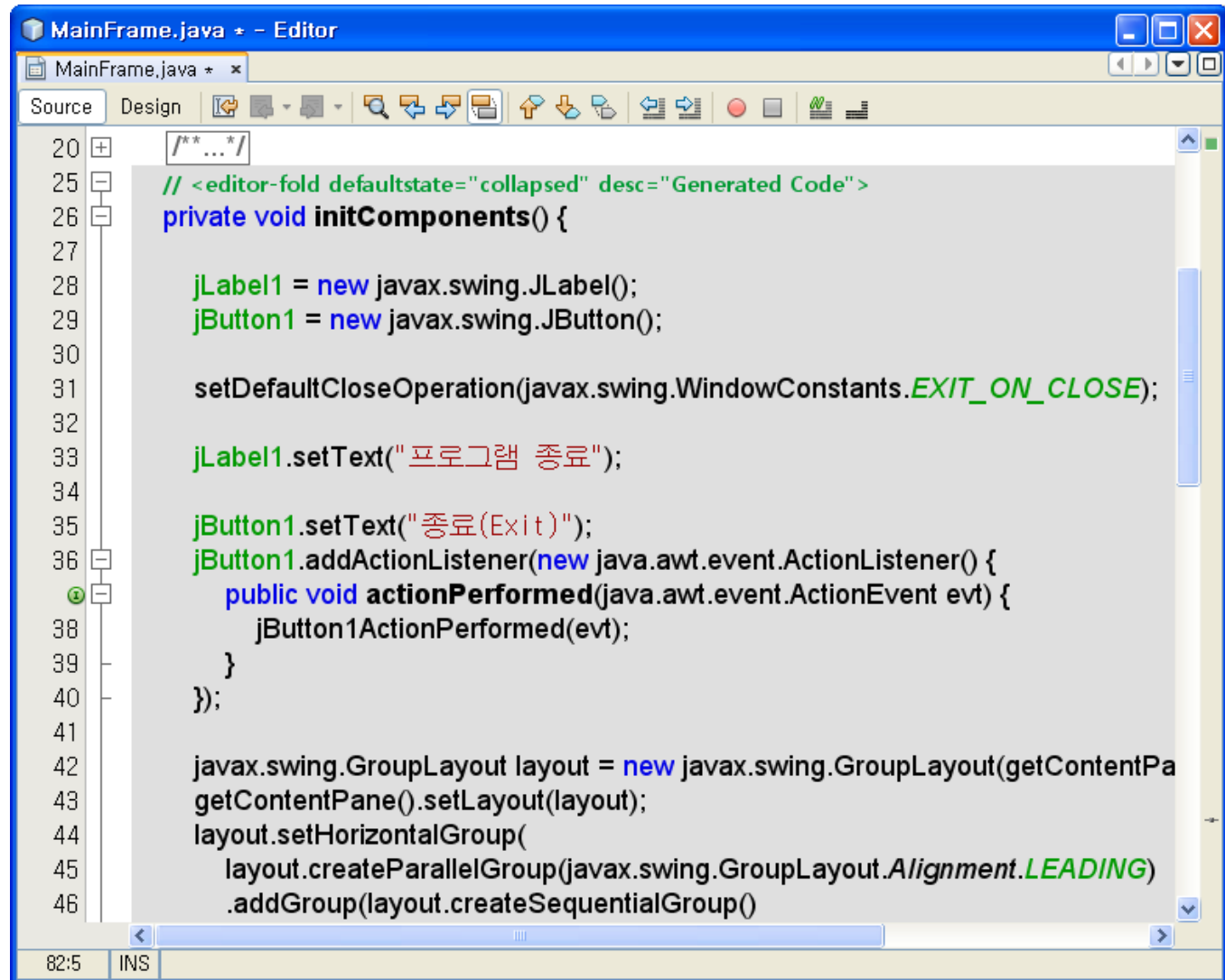
Label , Button 컨트롤 변수 선언





실습: Label & Button 컨트롤 (11)

❖ Label 및 Button 속성 지정에 따른 Source Coding



```
20  /** ... */
25  // <editor-fold defaultstate="collapsed" desc="Generated Code">
26  private void initComponents() {
27
28      jLabel1 = new javax.swing.JLabel();
29      jButton1 = new javax.swing.JButton();
30
31      setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
32
33      jLabel1.setText("프로그램 종료");
34
35      jButton1.setText("종료(Exit)");
36      jButton1.addActionListener(new java.awt.event.ActionListener() {
37          public void actionPerformed(java.awt.event.ActionEvent evt) {
38              jButton1ActionPerformed(evt);
39          }
40      });
41
42      javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
43      getContentPane().setLayout(layout);
44      layout.setHorizontalGroup(
45          layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
46              .addGroup(layout.createSequentialGroup()
                  .addContainerGap()
                  .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                      .addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT_SIZE, 150, true)
                      .addComponent(jButton1, javax.swing.GroupLayout.DEFAULT_SIZE, 100, true))
                  .addContainerGap(150, true))
      );
47  }
```





실습: Label & Button 컨트롤 (12)

■ [종료(Exit)] Button Event Handler 구현

```
MainFrame.java - Editor
MainFrame.java *
Source Design
67
68 private void jButton1ActionPerformed(java.awt.event.ActionEvent
69     System.exit(0); //프로그램 종료
70 }
71
72 /** ... */
75 public static void main(String args[]) {
76     java.awt.EventQueue.invokeLater(new Runnable() {
77         public void run() {
78             new MainFrame().setVisible(true);
79         }
80     });
81 }
82 // Variables declaration - do not modify
83 private javax.swing.JButton jButton1;
84 private javax.swing.JLabel jLabel1;
85 // End of variables declaration
86
87 }
```



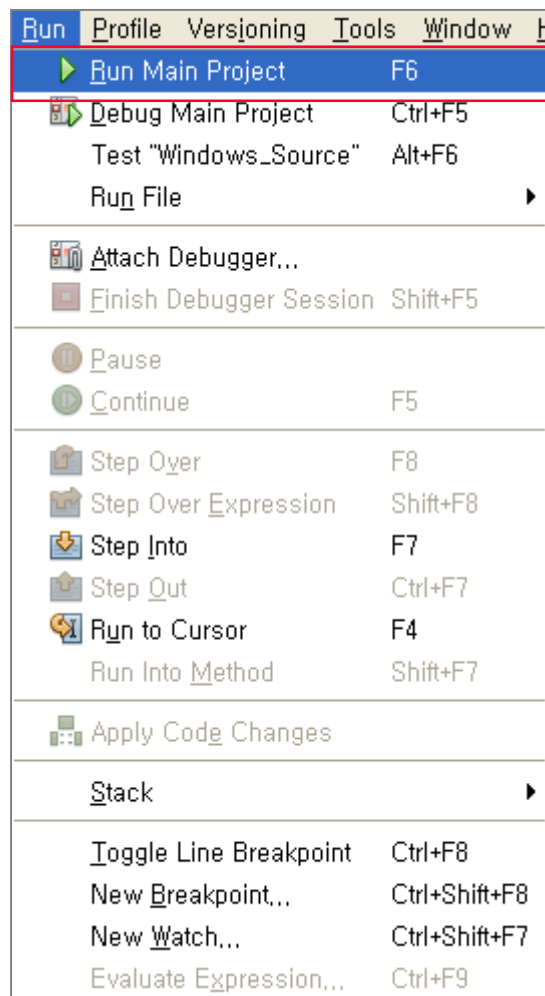
18. Coding



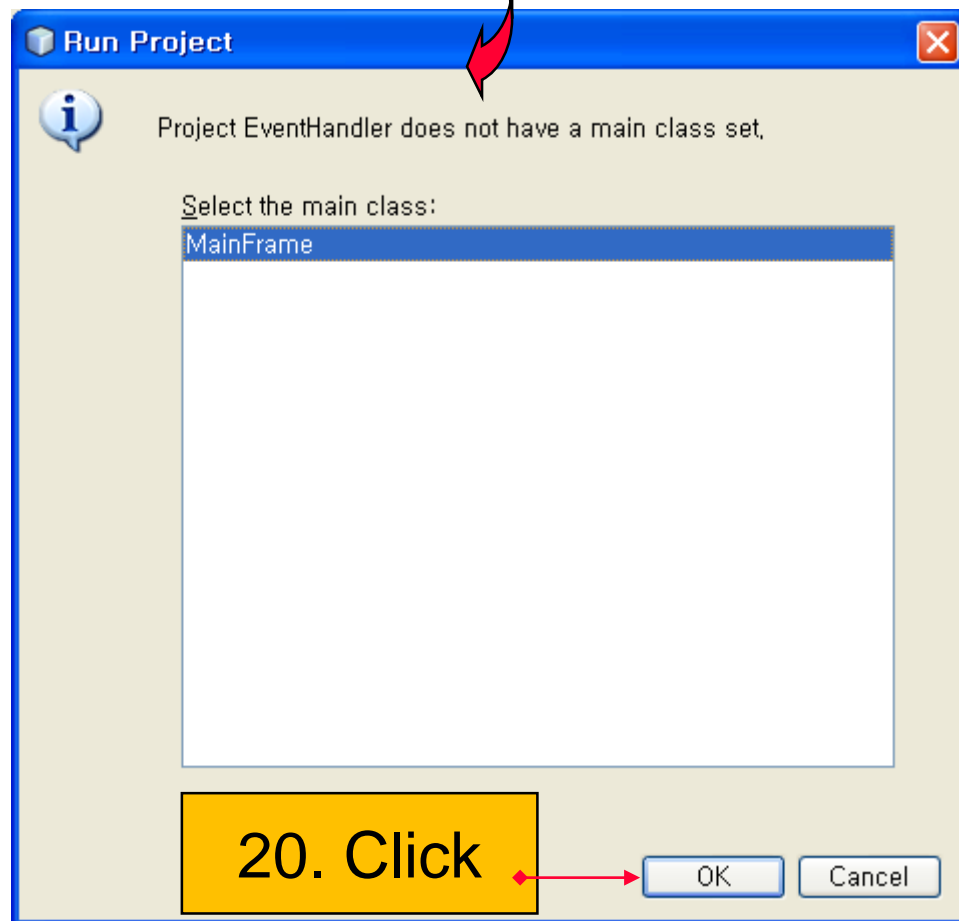


실습: Label & Button 컨트롤 (13)

실행



19. Click



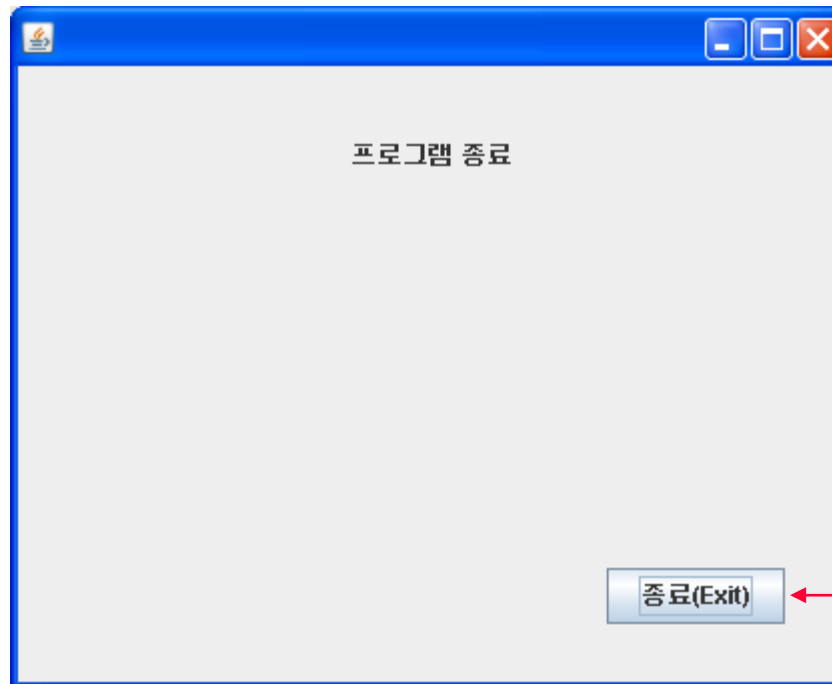


실습: Label & Button 컨트롤 (14)

■ 실행 결과

❖ 종료(Exit) Button Click

◆ 프로그램 종료



21. Click





Text Field 컨트롤

■ TextField 컨트롤

❖ 기능

◆ 단일 문자열 입력

❖ 형식

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

str	Textfield에 표시될 기본 문자열
size	Textfield 크기

❖ Method

Method	기능
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 컨트롤

■ TextArea 컨트롤

❖ 기능

◆ 다중 문자열 입력

❖ 형식

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

str	TextArea에 표시될 기본 문자열
rows	행
cols	열

❖ Method

Method	기능
void append(String str)	기존의 문자열에 str 문자열 추가
int getColumns	열의 수 반환
int getRows	행의 수 반환





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

```

//String s = "byte";
byte b = 1;
String s = "byte";
return (new Object() {
    returns "[Ljava.lang.Object;
(new int[] {4} {5} {6} {7} {8} {9} ).getClass()
returns "[I][I]"

```



실습: Text Field & Text Area 컨트롤 (1)

Application 생성

1. Click

2. Click

3. Click

Steps

1. Choose Project
2. ...

Choose Project

Filter:

Categories:

- Java
- JavaFX
- Maven
- NetBeans Modules
- Samples

Projects:

- Java Application
- Java Class Library
- Java Project with Existing Sources
- Java Free-Form Project

Description:

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.

< Back Next > Finish Cancel Help





실습: Text Field & Text Area 컨트롤 (2)

■ Project Name and Location

❖ Project name: TextFieldTextArea_Source

4. TextFieldTextArea_Source 입력

Steps

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

Name and Location

Project Name: TextFieldTextArea_Source

Project Location: C:\Java_Projects\Source Browse...

Project Folder: Java_Projects\Source\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

5. Click

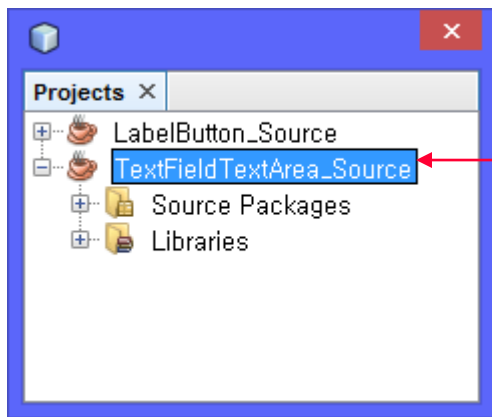
< Back Next > Finish Cancel Help



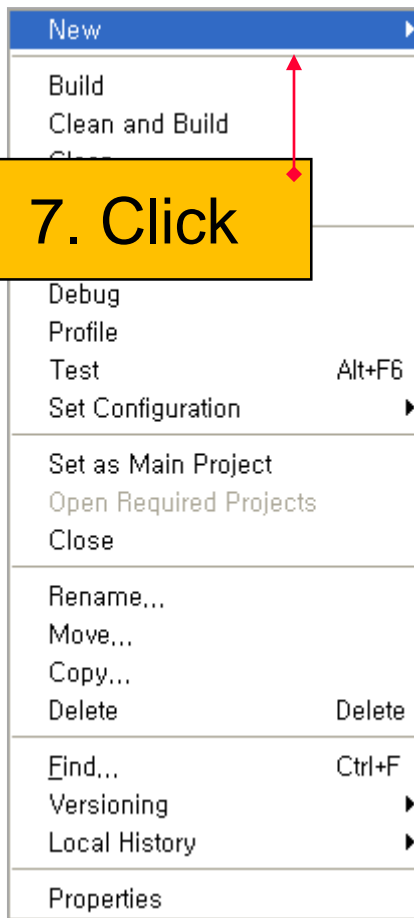


실습: Text Field & Text Area 컨트롤 (3)

JFrame Form 생성

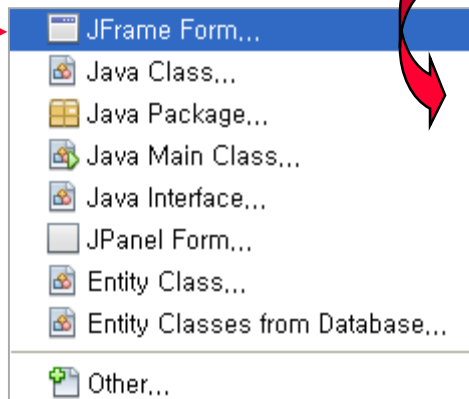


6. 마우스 오른쪽 버튼 Click



7. Click

8. Click





실습: Text Field & Text Area 컨트롤 (4)

■ JFrame Form Name 지정

❖ 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_Projects\Source\TextFieldTextArea_Source\src\MainFrame.java

9. MainFrame 입력

10. Click

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

< Back Next > **Finish** Cancel Help

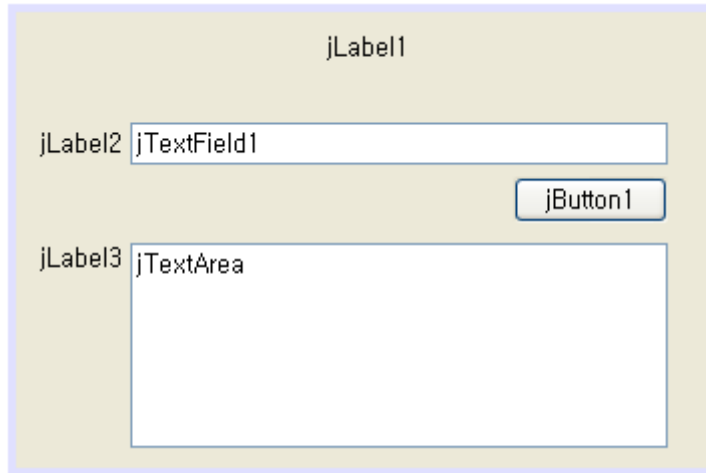
Projects

- LabelButton_Source
- TextFieldTextArea_Source
 - Source Packages
 - <default package>
 - MainFrame.java**
- Libraries

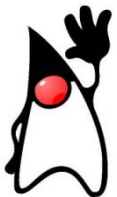


실습: Text Field & Text Area 컨트롤 (5)

■ 컨트롤 Layout 및 속성 지정



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





실습: Text Field & Text Area 컨트롤 (6)

■ 컨트롤 Layout 및 속성 지정 완료

TextField 및 TextArea 예제

한 줄 입력

다중 라인 입력

입력(Input)





실습: Text Field & Text Area 컨트롤 (7)

❖ 입력(Input) Button Event Handler

```
MainFrame.java - Editor
MainFrame.java x
Source Design
20  /** ... */
25  // <editor-fold defaultstate="collapsed" desc="Generated Code">
26  private void initComponents() {...} // </editor-fold>
101
102  private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
103      String strCRLF = "\r\n"; //CR + LF 입력
104      String strData = ""; //TextArea에 입력할 문자열
105      strData = jTextArea1.getText(); //TextField의 문자열을 strData 변수에 지정
106      strData += jTextField1.getText() + strCRLF; //TextField 문자열을 TextArea 문자열에 추가
107      jTextArea1.setText(strData);
108  }
109
110  /**
111   * @param args the command line arguments
112   */
113  public static void main(String args[]) {
114      java.awt.EventQueue.invokeLater(new Runnable() {
115          public void run() {
116              new MainFrame().setVisible(true);
117          }
118      });
119  }
128:48 INS
```

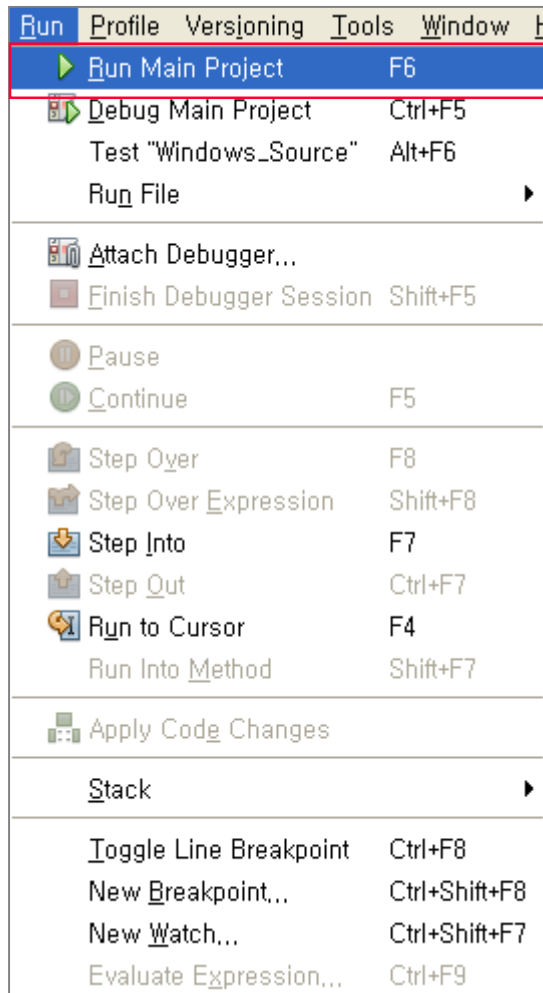
11. Coding



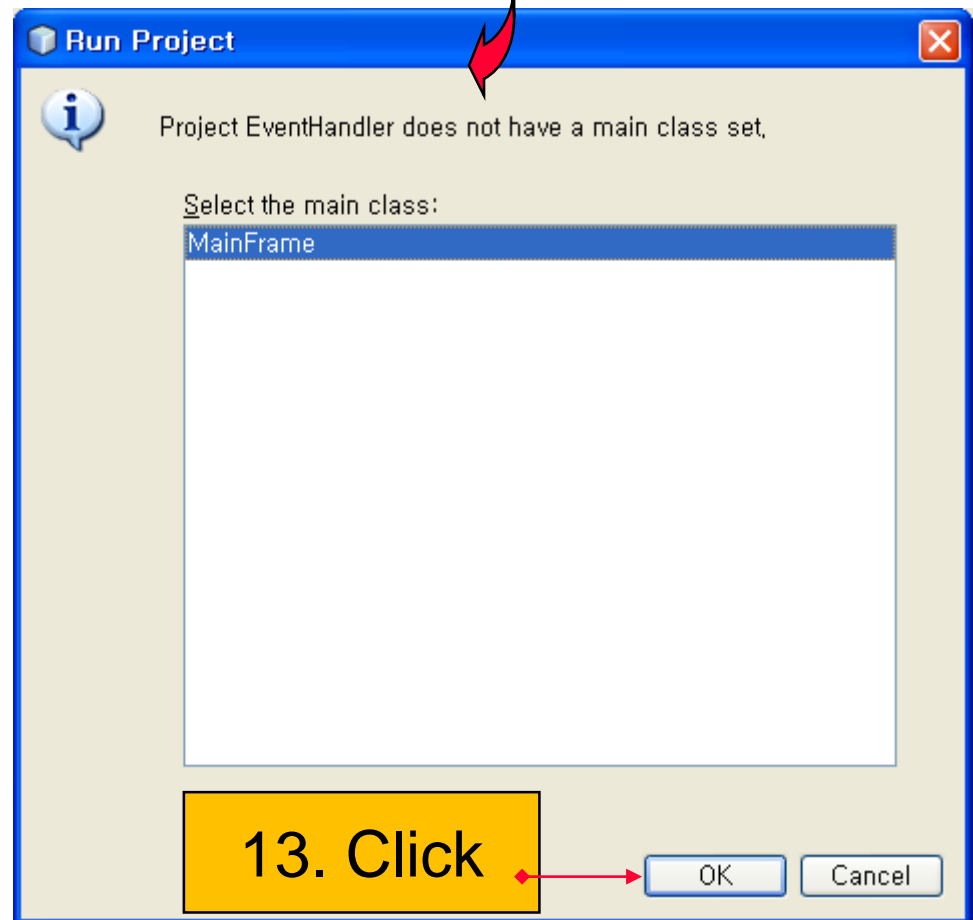


실습: Text Field & Text Area 컨트롤 (8)

실행



12. Click





실습: Text Field & Text Area 컨트롤 (9)

■ 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)





실습: Text Field & Text Area 컨트롤 (10)

TextField 및 TextArea 예제

한 줄 입력 재미 있다.

다중 라인 입력 자바

입력(Input)

14. 입력

15. Click

TextField 및 TextArea 예제

한 줄 입력 재미 있다.

다중 라인 입력 자바
재미 있다.

입력(Input)





실습2: Text Field & Text Area 컨트롤

❖ Project Name: TextAreaAppend_Source (실습시간 : 20분)

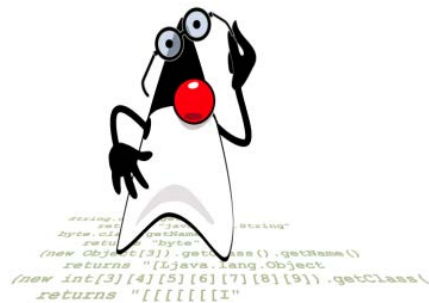
- 다중 라인 입력 TextArea의 문자열에 한 줄 입력 TextField의 문자열을 추가 한다.
 - append() method를 이용하여 동일한 기능을 하도록 구현하시오.

TextField 및 TextArea 예제

한 줄 입력 재미 있다.

다중 라인 입력 자바

입력(Input)





실습2: Solution

❖ 입력(Input) Button Event Handler

```
MainFrame.java * - Editor
MainFrame.java *
Source Design
13 public class MainFrame extends javax.swing.JFrame {
14
15     /** Creates new form MainFrame */
16     public MainFrame() {
17         initComponents();
18     }
19
20     /** ... */
25 Generated Code
101
102 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
103     String strData = null; //TextArea에 입력할 문자열
104     strData = jTextField1.getText() + "\n"; //TextField의 문자열을 strData 변수에 지정
105     jTextArea1.append(strData); //TextField 문자열을 TextArea 문자열에 추가
106 }
107
112:57 INS
```

16. Coding

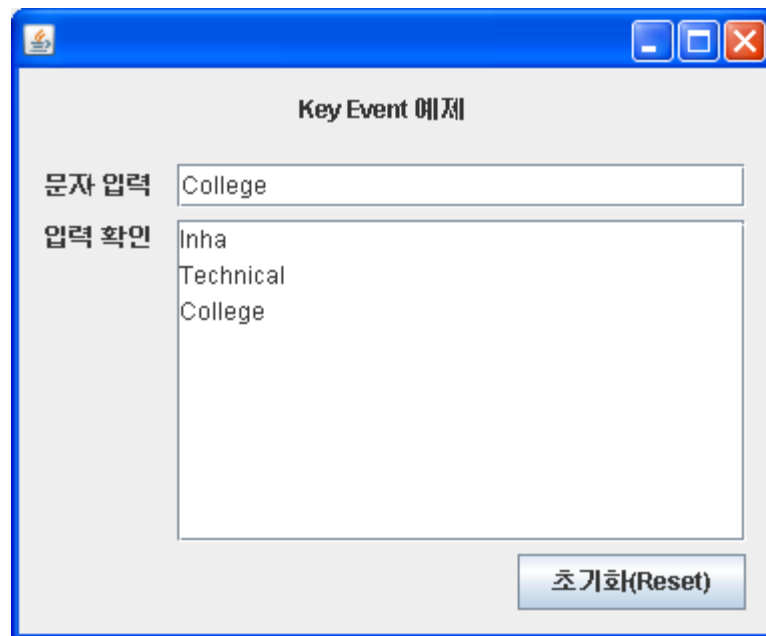
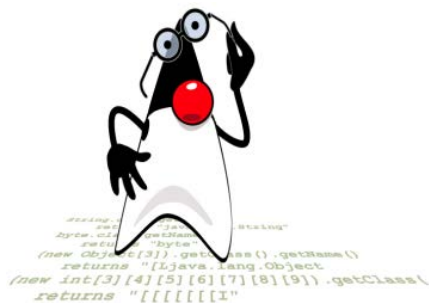




실습 1 : Key Event Handler

❖ Project Name: KeyEvnet_Source(실습시간 : 20분)

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





실습 1 : Key Event Handler (1)

Application 생성

The screenshot illustrates the process of creating a new application in the NetBeans IDE. It is divided into three numbered steps:

- 1. Click**: A red arrow points to the 'New Project...' option in the 'File' menu, which is highlighted with a red box.
- 2. Click**: A red arrow points to the 'Java Application' option in the 'Projects' list of the 'New Project' dialog, which is highlighted with a red box.
- 3. Click**: A red arrow points to the 'Next >' button at the bottom of the 'New Project' dialog, which is highlighted with a red box.

The 'New Project' dialog shows the following details:

- Steps**:
 1. Choose Project
 2. ...
- Choose Project**:
 - Filter: []
 - Categories: Java, JavaFX, Maven, NetBeans Modules, Samples
 - Projects: Java Application (selected), Java Class Library, Java Project with Existing Sources, Java Free-Form Project
- Description**:

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.
- Buttons**: < Back, Next > (highlighted), Finish, Cancel, Help





실습 1 : 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_Projects\Source Browse...

Project Folder: C:\Java_Projects\Source\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

4. Project Name 입력

5. Click

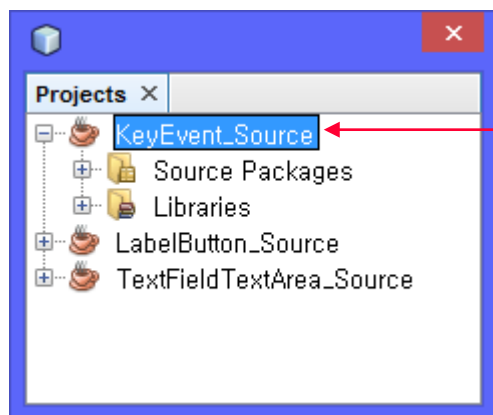
< Back Next > Finish Cancel Help



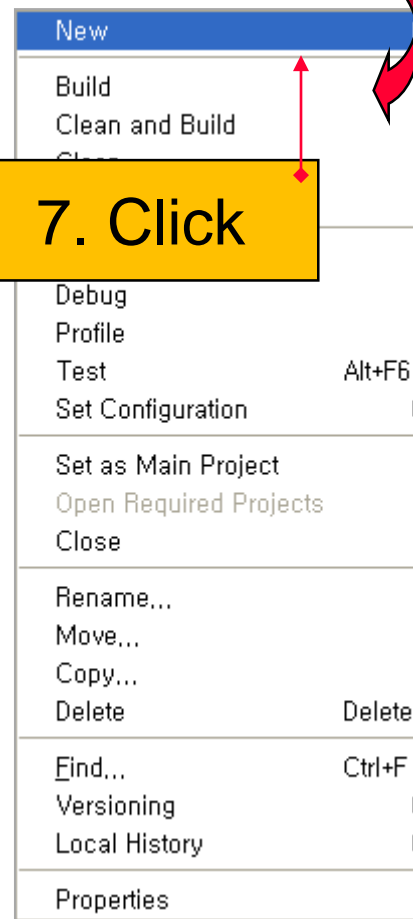


실습 1 : Key Event Handler (3)

JFrame Form 생성

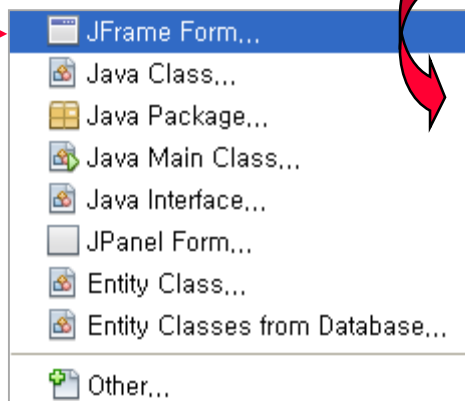


6. 마우스 오른쪽 버튼 Click



7. Click

8. Click





실습 1 : Key Event Handler (4)

JFrame Form Name 지정

❖ MainFrame.java 생성

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_Projects\Source\KeyEvent_Source\src\MainFrame.java

9. MainFrame 입력

10. Click

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

< Back Next > **Finish** Cancel Help





실습 1 : Key Event Handler (5)

■ 컨트롤 Layout 및 속성 지정

Key Event 예제

문자 입력

입력 확인

컨트롤	속성지정
jLabel1	• Text: Key Event 예제
jLabel2	• Text: 문자 입력
jLabel3	• Text: 입력 확인
jTextField1	• Text:
jTextArea1	• Text:
jButton1	• Text: 초기화(Reset)

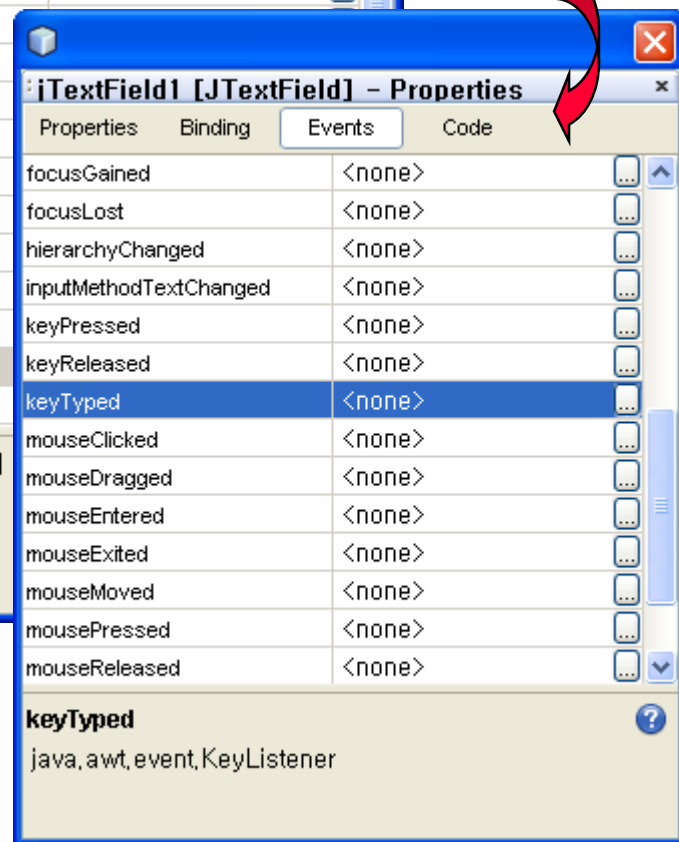
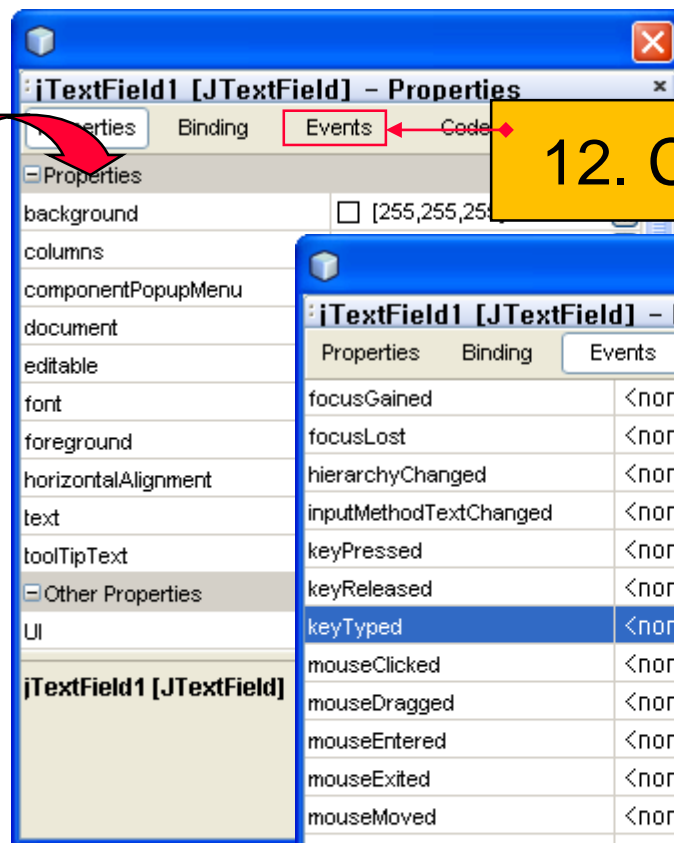
컨트롤	Event
jTextField1	• ActionPerformed(Event evt) • KeyTyped(Event evt)
jButton1	• ActionPerformed(Event evt)





실습 1 : Key Event Handler (6)

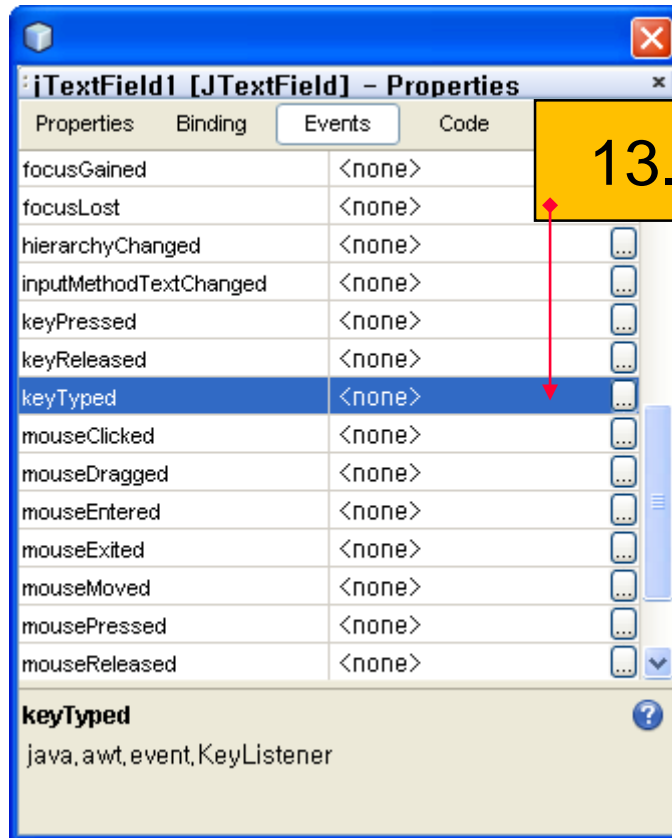
■ JTextField Event 지정



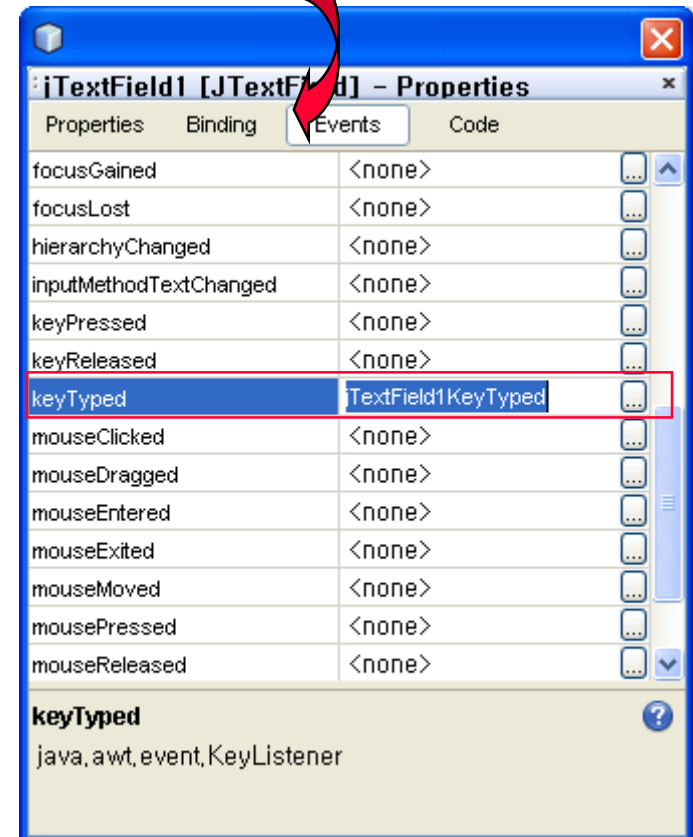


실습 1 : Key Event Handler (7)

■ jTextField1 KeyTyped Event 지정



13. Double Click





실습 1 : Key Event Handler (8)

❖ KeyTyped Event Handler

MainFrame.java * - Editor

MainFrame.java * x

Source Design

```
1  /** ... */
6
7  import java.awt.event.KeyEvent;           //VK_BACK_SPACE, VK_ENTER Key 값 사용
8  /** ... */
12 public class MainFrame extends javax.swing.JFrame {
13     /** ... */
14     public MainFrame() {
15         initComponents();
16     }
17
18     /** ... */
23     Generated Code
109
110    private void jTextField1KeyTyped(java.awt.event.KeyEvent evt) {
111        if ( evt.getKeyChar() != KeyEvent.VK_BACK_SPACE){
112            jTextArea1.setText(jTextArea1.getText() + evt.getKeyChar());           //입력 문자 추가
113        } else {
114            if ( evt.getKeyChar() == KeyEvent.VK_ENTER)
115                jTextArea1.setText(jTextArea1.getText() + evt.getKeyChar() + "\n"); //new line 추가
116        }
117    }
```

138:5 INS

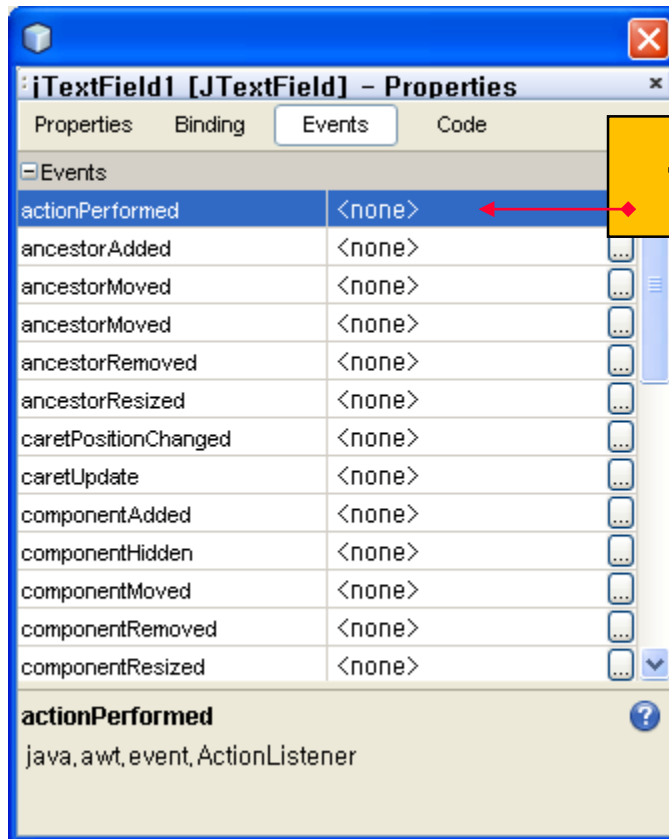
14. Coding



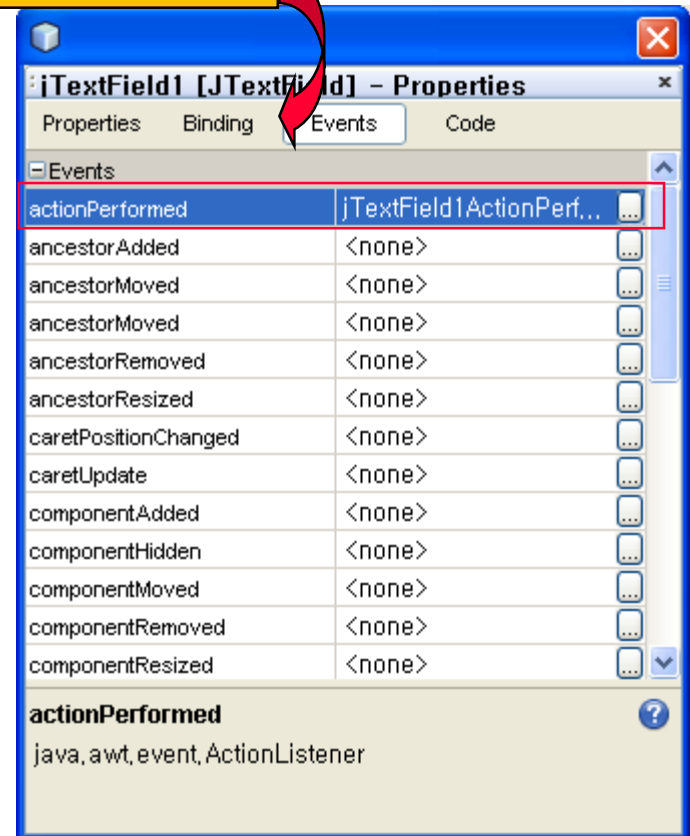


실습 1 : Key Event Handler (9)

■ jTextField1 ActionPerformed Event 지정



15. Double Click





실습 1 : Key Event Handler (10)

❖ jTextField1 ActionPerformed Event Handler

```
MainFrame.java * - Editor
MainFrame.java * x
Source Design [Icons]
110 private void jTextField1KeyTyped(java.awt.event.KeyEvent evt) {...}
118
119 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
120     jTextField1.setText(null); // jTextField1 문자열 초기화
121     jTextArea1.setText(null); // jTextArea1 문자열 초기화
122 }
123
124 private void jTextField1ActionPerformed(java.awt.event.ActionEvent evt) {
125     jTextField1.setText(null); // jTextField1 문자열 초기화
126 }
127
128 /** ... */
131 public static void main(String args[]) {
132     java.awt.EventQueue.invokeLater(new Runnable() {
133         public void run() {
134             new MainFrame().setVisible(true);
135         }
136     });
137 }
143:40 INS
```

16. Coding





실습 1 : Key Event Handler (11)

■ jButton1 ActionPerformed Event 지정

Key Event 예제

문자 입력

입력 확인

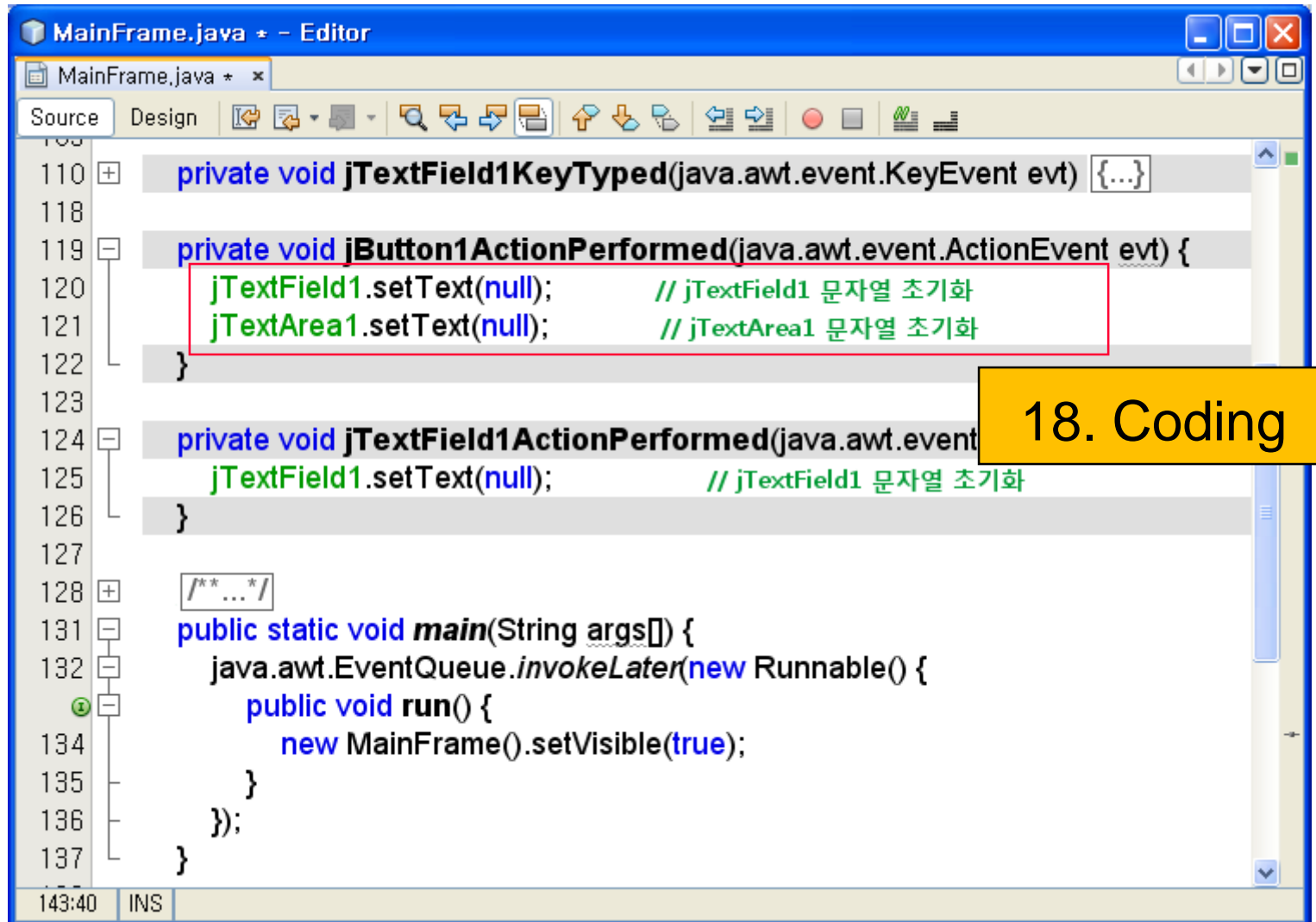
17. Double Click





실습 1 : Key Event Handler (12)

❖ jButton1 ActionPerformed Event Handler



```
MainFrame.java - Editor
MainFrame.java *
Source Design
110 private void jTextField1KeyTyped(java.awt.event.KeyEvent evt) {...}
118
119 private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
120     jTextField1.setText(null); // jTextField1 문자열 초기화
121     jTextArea1.setText(null); // jTextArea1 문자열 초기화
122 }
123
124 private void jTextField1ActionPerformed(java.awt.event
125     jTextField1.setText(null); // jTextField1 문자열 초기화
126 }
127
128 /** ... */
131 public static void main(String args[]) {
132     java.awt.EventQueue.invokeLater(new Runnable() {
133         public void run() {
134             new MainFrame().setVisible(true);
135         }
136     });
137 }
143:40 INS
```

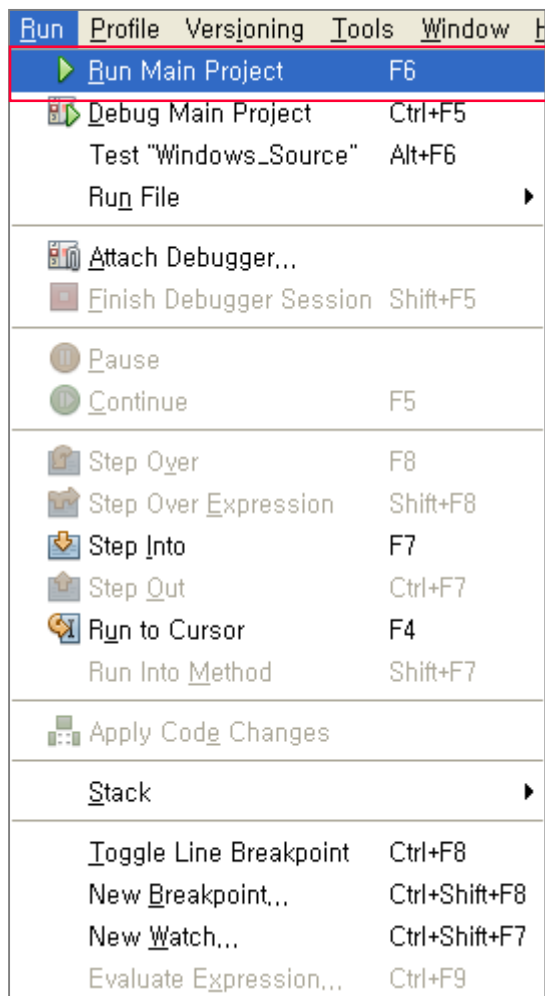
18. Coding



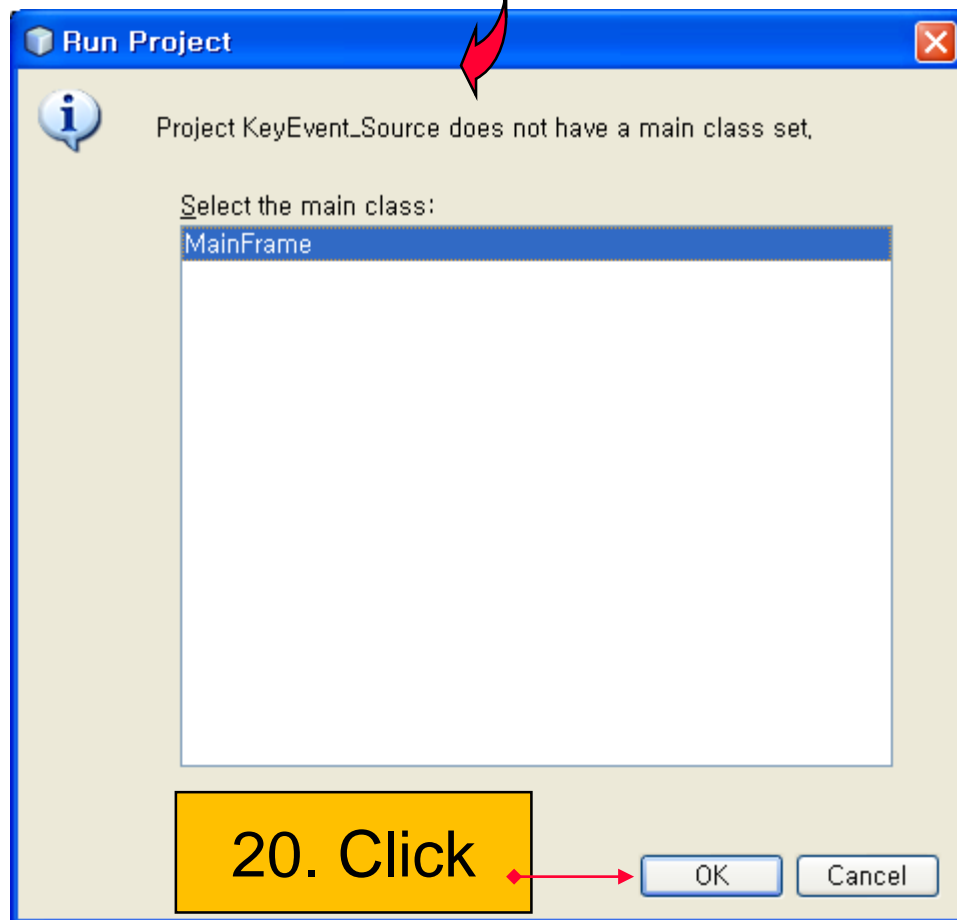


실습 1 : Key Event Handler (13)

실행



19. Click



20. Click



실습 1 : Key Event Handler (14)

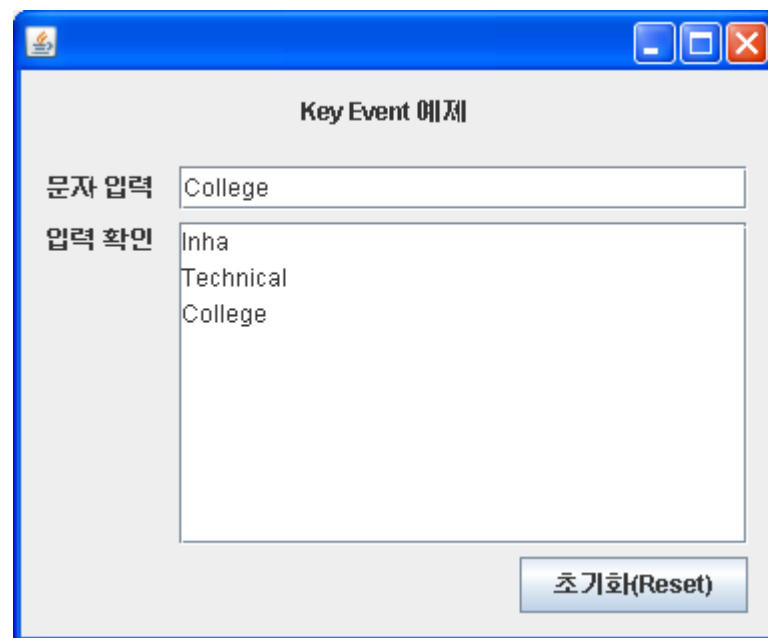
■ 실행 결과

❖ 문자 입력

- ◆ Inha
- ◆ Technical
- ◆ College

❖ 초기화(Reset) Button Click

- ◆ TextField 초기화
- ◆ Text Are 초기화

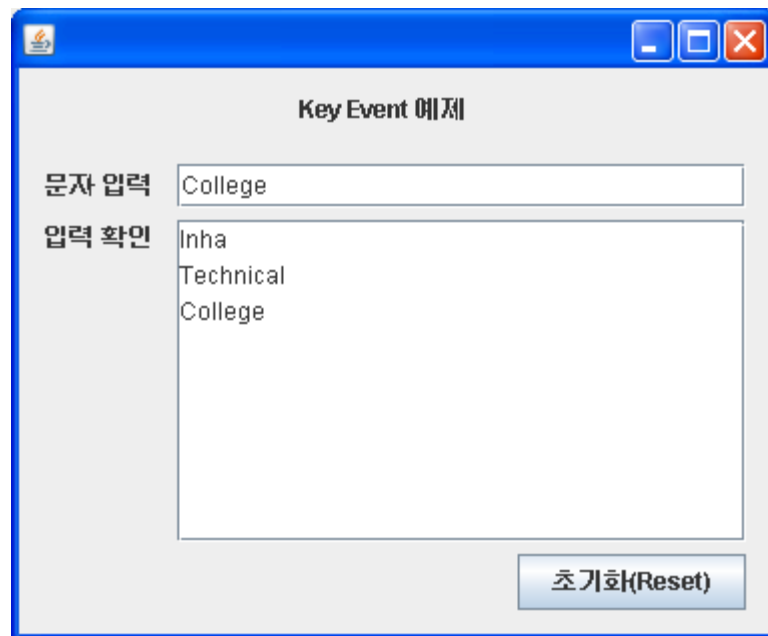
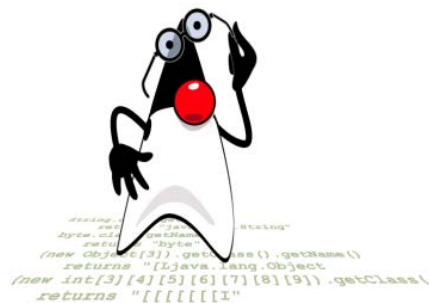




실습 2 : Key Event Handler

❖ Project Name: KeyEvnet2_Source(실습시간 : 20분)

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





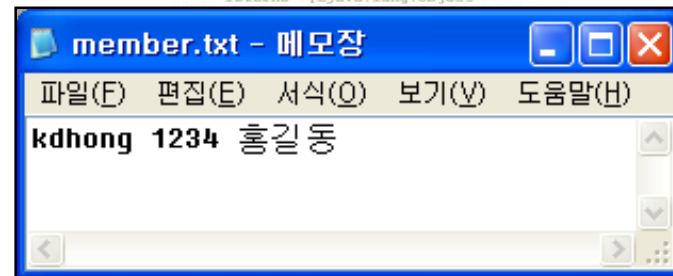
Homework # 1

❖ 회원 가입 모듈 구현

- Project 명: Member_ID
- [중복 확인] 버튼 Click
 - **member.txt** 파일에서 회원정보를 **read**하여 중복되는 “아이디”가 존재하는지를 검색하여 결과를 출력한다.
- [회원 가입] 버튼 Click
 - 비밀번호가 일치하면 아이디, 비밀번호, 이름을 **member.txt** 파일에 저장한다.

회원 가입

아이디	<input type="text" value="kdhong"/>	중복 확인
비밀번호	<input type="password" value="****"/>	
비밀번호 확인	<input type="password" value="****"/>	회원 가입
성명	<input type="text" value="홍길동"/>	



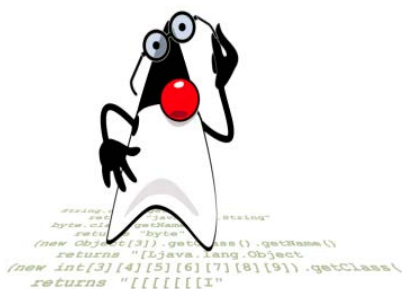


Homework # 2

❖ 계산기 구현

■ Project 명: Calculator_Source

- 일반용
- 공학용





학습 요약

- Label 컨트롤
- Button 컨트롤
- Text Field 컨트롤
- Text Area 컨트롤

