



GUI 컨트롤 II

CheckBox & RadioButton 예제

결제 방법

☐ 현금 이체

☒ 카드

☐ 포인트

회원 구분

☐ 회원

☒ 비회원

확인(OK)

결제 내역 확인

비회원 자격으로 카드 결제 방법을 선택하였습니다.





학습 목표

■ 이 강의를 마치면 학생들은

- ❖ Checkbox 컨트롤 기능에 대하여 설명할 수 있다.
- ❖ Radio Button 컨트롤 기능에 대하여 설명할 수 있다.
- ❖ Choice와 Combo Box 컨트롤 기능에 대하여 설명할 수 있다.
- ❖ List 컨트롤 기능에 대하여 설명할 수 있다.





Checkbox 컨트롤 (1)

■ Checkbox 컨트롤

❖ 기능

◆ 다중 선택 구현

❖ 생성자

생성자	기능
Checkbox()	Blank checkbox 생성
Checkbox(String text)	Text 문자열의 checkbox 생성
Checkbox(String text, boolean state)	Text 문자열을 가지며, state 값에 따라 선택 여부 결정
Checkbox(String text, boolean state, CheckboxGroup group)	Text 문자열을 가지며, state 값에 따라 선택 여부 결정한다. group으로 명시된 그룹의 checkbox 생성





Checkbox 컨트롤 (2)

❖ Method

Method	기능
setText(String text)	CheckBox의 text 지정
String getText()	CheckBox의 text 반환
setSelected(Boolean b)	CheckBox의 선택 여부 지정
Boolean isSelected()	CheckBox의 선택 여부 반환





Radio Button 컨트롤 (1)

■ Radio Button 컨트롤

❖ 기능

◆ 단일 선택

❖ Method

Method	기능
setText(String text)	Radio Button의 text 지정
String getText()	Radio Button의 text 반환
setSelected(Boolean b)	Radio Button 의 선택 여부 지정
Boolean isSelected()	Radio Button 의 선택 여부 반환



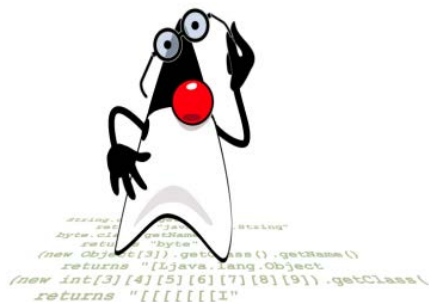


실습: CheckBox & Radio Button 컨트롤

❖ Project Name: CheckBoxRadioButton_Source (실습시간 : 20분)

■ 확인(OK) button Click

- 결제 방법과 회원 구분 에서 선택한 내용을 Text Area에 추가하도록 프로그래밍하시오.



CheckBox & RadioButton 예제

결제 방법

☒ 현금 이체

☐ 카드

☐ 포인트

회원 구분

☒ 회원

☐ 비회원

확인(OK)

결제 내역 확인

회원 자격으로 현금 이체 결제 방법을 선택하였습니다.



실습: CheckBox & Radio Button 컨트롤 (1)

Application 생성

1. Click

2. Click

3. Click

File Edit View Navigate Source Refactor Run

New Project... Ctrl+Shift+N

New File... Ctrl+N

Open Project... Ctrl+Shift+O

Open Recent Project

Close Project

Close Other Projects

Close All Projects

Open File...

Open Recent File

Project Groups...

Project Properties

Import Project

Export Project

Save Ctrl+S

Save As...

Save All Ctrl+Shift+S

Page Setup...

Print... Ctrl+Alt+P

Print to HTML...

Exit

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: CheckBox & Radio Button 컨트롤 (2)

■ Project Name and Location

❖ Project name: CheckBoxRadionButton_Source

4. Project Name 입력

New Java Application

Steps

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

Name and Location

Project Name:

Project Location:

Project Folder:

☐ 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

5. Click

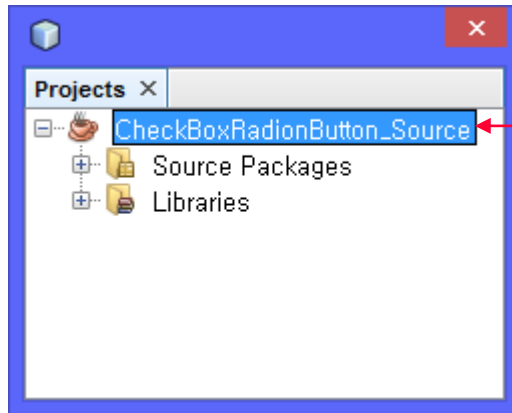
< Back Next > Finish Cancel Help





실습: CheckBox & Radio Button 컨트롤 (3)

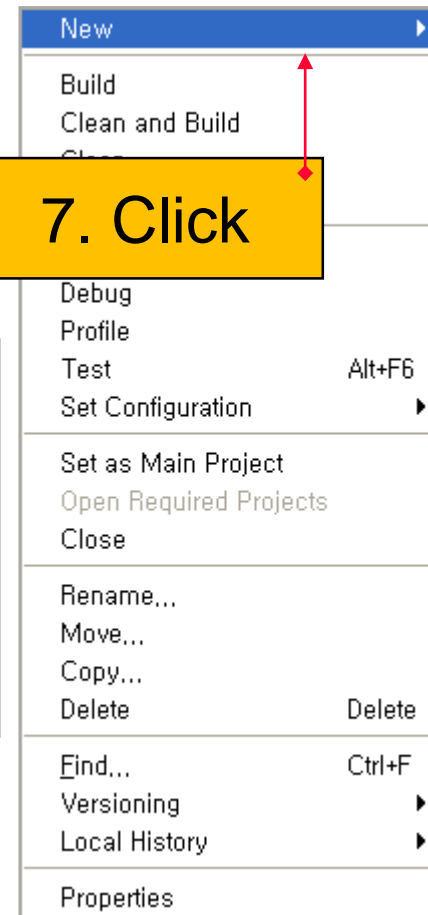
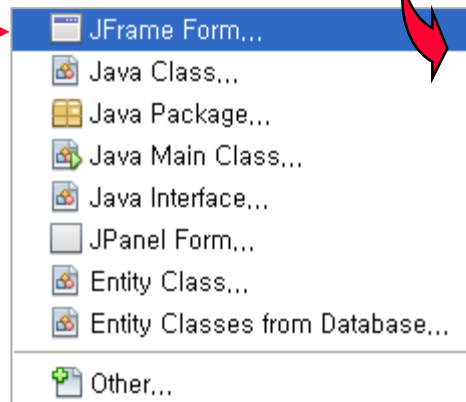
■ JFrame Form 생성



6. 마우스 오른쪽 버튼 Click

7. Click

8. Click





실습: CheckBox & Radio 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: CheckBoxRadionButton_Source

Location: Source Packages

Package:

Created File: C:\Java_Projects\Source\CheckBoxRadionButton_Source\src\MainFrame.java

9. MainFrame 입력

Projects

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

10. Click

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

< Back Next > Finish Cancel Help



실습: CheckBox & Radio Button 컨트롤 (5)

■ 컨트롤 배치 및 속성 지정

The GUI contains the following components:

- jLabel1 (Title)
- jLabel2
- ☐ jCheckBox1
- ☐ jCheckBox2
- ☐ jCheckBox3
- jLabel3
- ☐ jRadioButton1
- ☐ jRadioButton2
- jButton1
- jLabel4
- TextArea (jTextArea1)

컨트롤	속성 지정
jLabel1	• Text: CheckBox & RadioButton 예제
jLabel2	• Text: 결재 방법
jCheckBox1	• Text: 현금 이체 • Selected: 체크
jCheckBox2	• Text: 카드
jCheckBox3	• Text: 포인트
jLabel3	• Text: 회원 구분
jRadiobutton1	• Text: 회원 • Selected: 체크
jRadiobutton2	• Text: 비회원
jLabel4	• Text: 결재 내역 확인
jTextArea1	• Text:
jButton1	• Text: 확인(OK)

컨트롤	속성(Code) 지정
jButton1	• Variable Name: btnOK





실습: CheckBox & Radio Button 컨트롤 (6)

■ Button Group 컨트롤 추가

CheckBox & RadioButton 예제

결제 방법

☒ 현금 이체

☐ 카드

☐ 포인트

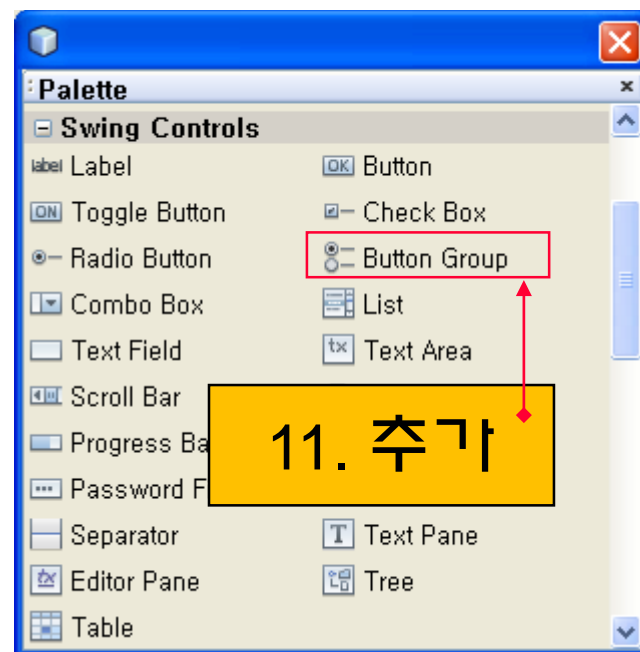
회원 구분

☒ 회원

☐ 비회원

결제 내역 확인

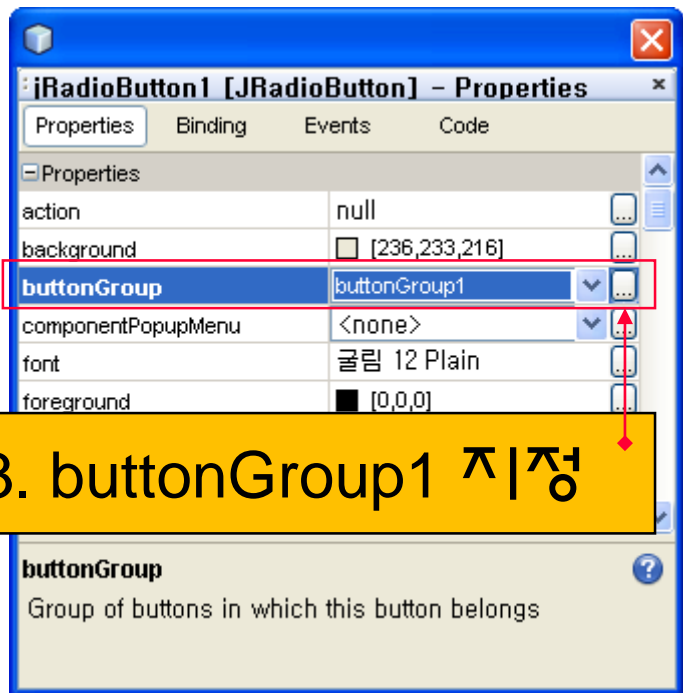
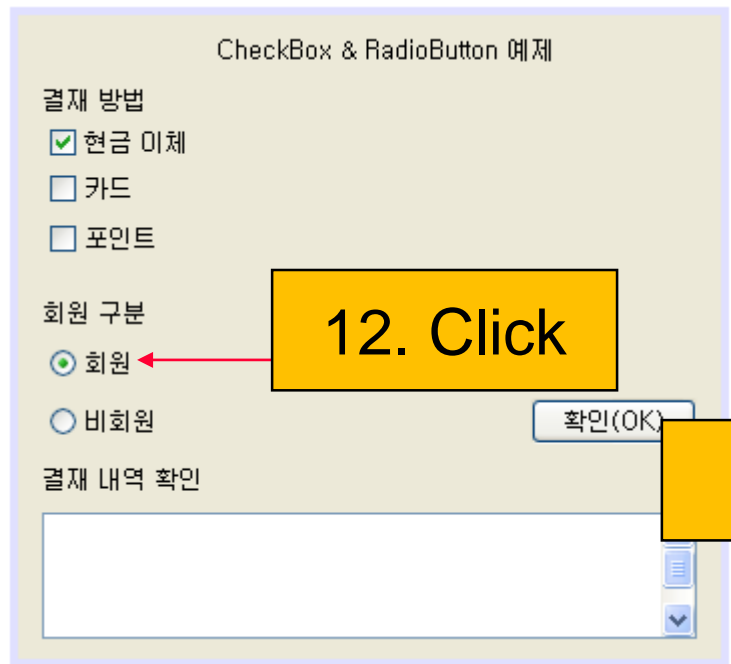
확인(OK)





실습: CheckBox & Radio Button 컨트롤 (7)

❖ Button Group 추가 및 지정



컨트롤	속성 지정
JRadioButton1	• buttonGroup: buttonGroup1
JRadioButton2	• buttonGroup: buttonGroup1





실습: CheckBox & Radio Button 컨트롤 (8)

■ 컨트롤 배치 및 속성 지정 완료

CheckBox & RadioButton 예제

결제 방법

☒ 현금 이체

☐ 카드

☐ 포인트

회원 구분

☒ 회원

☐ 비회원

확인(OK)

결제 내역 확인

↑

↓





실습: CheckBox & Radio Button 컨트롤 (9)

❖ 확인(OK) Button Event Handler

```
MainFrame.java * - Editor
MainFrame.java * x
Source Design
142 private void btnOKActionPerformed(java.awt.event.ActionEvent evt) {
143     String strData = "회원님은 ";           //문자열 변수 strData 선언
144
145     if(jRadioButton1.isSelected())           //RadioButton1 선택 지정
146         strData = "회원 자격으로 ";
147     else
148         strData = "비회원 자격으로 ";
149
150     if(jCheckBox1.isSelected()) strData += jCheckBox1.getText(); //jCheckBox1 선택 지정
151     if(jCheckBox2.isSelected()) strData += jCheckBox2.getText(); //jCheckBox2 선택 지정
152     if(jCheckBox3.isSelected()) strData += jCheckBox3.getText(); //jCheckBox3 선택 지정
153     strData += " 결재 방법을 선택하셨습니다. \n";
154     jTextArea1.setText(strData);             // jTextArea1에 문자열 지정
155     strData = null;
156 }
157
158 /** ... */
161 public static void main(String args[]) {
162     java.awt.EventQueue.invokeLater(new Runnable() {
163         public void run() {
164             new MainFrame().setVisible(true);
165         }
166     });
167 }
170:39 INS
```

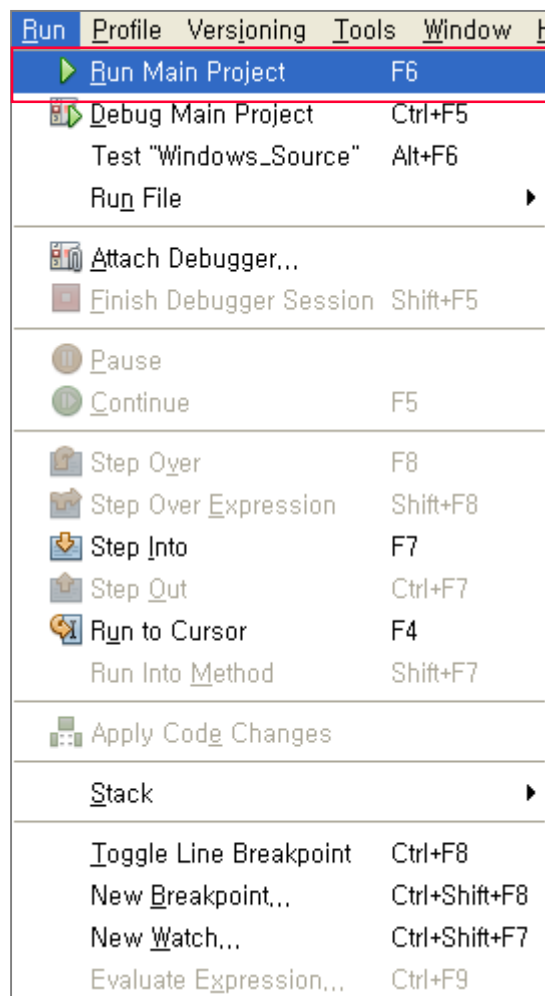
14. Coding



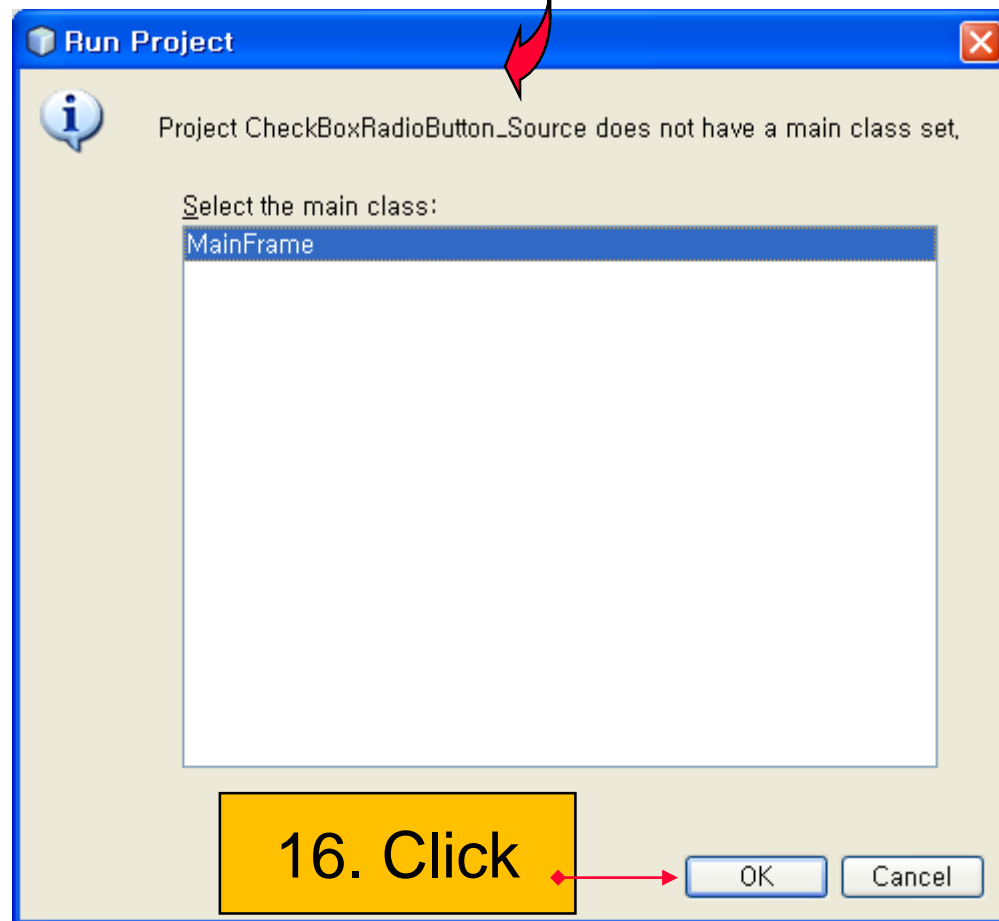


실습: CheckBox & Radio Button 컨트롤 (10)

실행



15. Click



16. Click





실습: CheckBox & Radio Button 컨트롤 (11)

■ 실행 결과

- ❖ 결제 방법 선택
- ❖ 회원 구분 선택
- ❖ 확인(OK) Button Click

CheckBox & RadioButton 예제

결제 방법

☒ 현금 이체

☐ 카드

☐ 포인트

회원 구분

☒ 회원

☐ 비회원

확인(OK)

결제 내역 확인





실습: CheckBox & Radio Button 컨트롤 (12)

■ 실행 결과

CheckBox & RadioButton 예제

결제 방법

☒ 현금 이체

☐ 카드

☐ 포인트

회원 구분

☒ 회원

☐ 비회원

확인(OK)

결제 내역 확인

회원 자격으로 현금 이체 결제 방법을 선택하였습니다.

CheckBox & RadioButton 예제

결제 방법

☐ 현금 이체

☒ 카드

☐ 포인트

회원 구분

☐ 회원

☒ 비회원

확인(OK)

결제 내역 확인

비회원 자격으로 카드 결제 방법을 선택하였습니다.





Homework # 1

❖ 회원 가입 모듈 구현 (Project 명: Member_ID)

■ [회원 가입] 버튼 Click

- 비밀번호가 일치하면 아이디, 비밀번호, 이름, 회원 구분, 관심 분야를 **member.txt** 파일에 저장한다.
- 회원의 관심분야는 아래와 같이 **bit연산**을 이용하여 계산한 값을 **member.txt** 파일에 저장한다.

수영	게임	골프	스키	독서	테니스
32	16	8	4	2	1

회원 가입

아이디: kdhong 중복 확인

비밀번호: ****

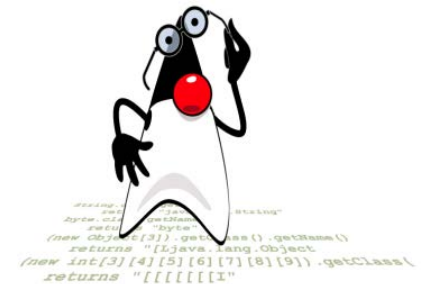
비밀번호 확인: ****

성명: 홍길동

회원 구분: ☒ 정회원 ☐ 준회원 ☐ 학생회원

관심 분야: ☒ 수영 ☐ 게임 ☐ 골프
☐ 스키 ☐ 독서 ☒ 테니스

회원 가입



member.txt - 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

kdhong 1234 홍길동 정회원 33





Combo Box 컨트롤

■ Combo Box 컨트롤

❖ 기능

◆ Drop-down 리스트 구현

◆ addItem() 메서드를 사용하여 drop-down 항목 추가

❖ Method

Method	기능
int getItemCount()	Item 수 반환
String getItemAt(int index)	Index에 해당하는 Item 반환
void addItem(Object obj)	Obj를 Combo Box Item으로 추가
Object getSelcetItem()	선택된 Item 반환
int getSelcetedIndex()	선택된 Item의 Index를 반환

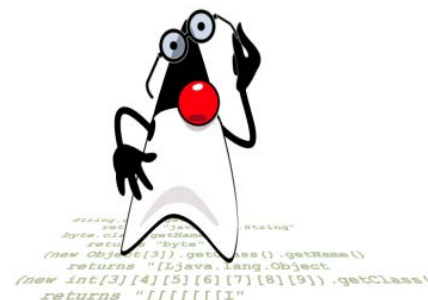
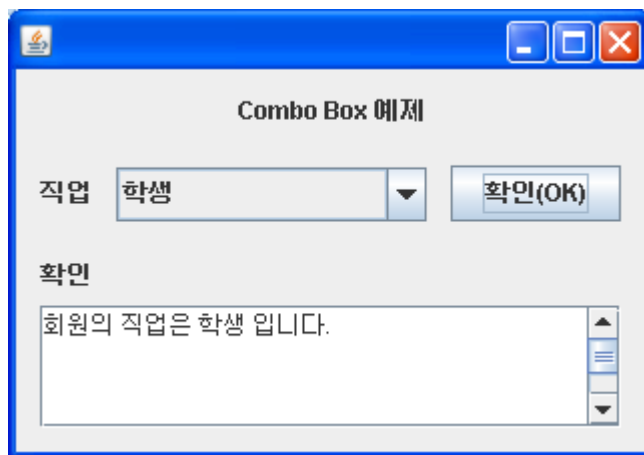




실습: Combo Box 컨트롤

❖ Project Name : ComboBox_Source (실습시간 : 20분)

- Combo Box에서 직업 선택
- 확인(OK) 버튼 Click
 - Combo Box에서 선택한 직업을 Text Area에 표현한다.





실습: Combo Box 컨트롤 (1)

Application 생성

1. Click

2. Click

3. Click

File Edit View Navigate Source Refactor Run

New Project... Ctrl+Shift+N

New File... Ctrl+N

Open Project... Ctrl+Shift+O

Open Recent Project

Close Project

Close Other Projects

Close All Projects

Open File...

Open Recent File

Project Groups...

Project Properties

Import Project

Export Project

Save Ctrl+S

Save As...

Save All Ctrl+Shift+S

Page Setup...

Print... Ctrl+Alt+P

Print to HTML...

Exit

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: Combo Box 컨트롤 (2)

■ Project Name and Location

❖ Project name: ComboBox_Source

New Java Application

Steps

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

Name and Location

Project Name: ComboBox_Source

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

Project Folder: C:\Java_Projects\Source\ComboBox_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 combobox_source, ComboBox_Source

5. Click

< Back Next > **Finish** Cancel Help

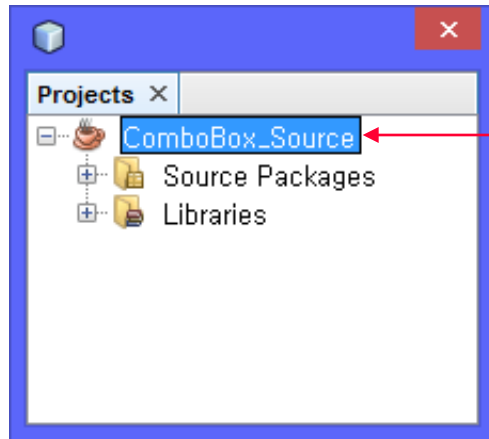
4. Project Name 입력



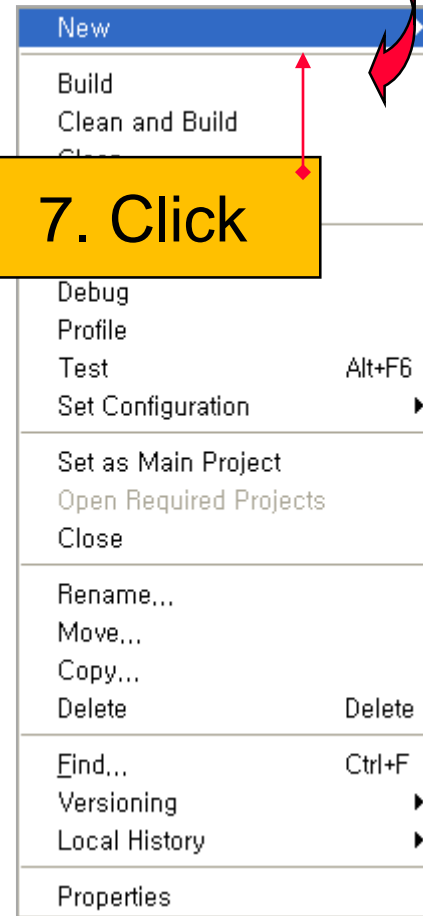


실습: Combo Box 컨트롤 (3)

JFrame Form 생성

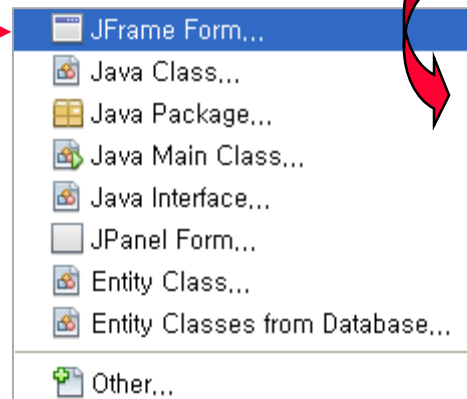


6. 마우스 오른쪽 버튼 Click



7. Click

8. Click





실습: Combo Box 컨트롤 (4)

■ JFrame Form Name 지정

❖ MainFrame.java 생성

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

Class Name: MainFrame

Project: ComboBox_Source

Location: Source Packages

Package:

Created File: C:\Java_Projects\Source\ComboBox_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: Combo Box 컨트롤 (5)

■ 컨트롤 배치 및 속성 지정

Combo Box 예제

직업

확인

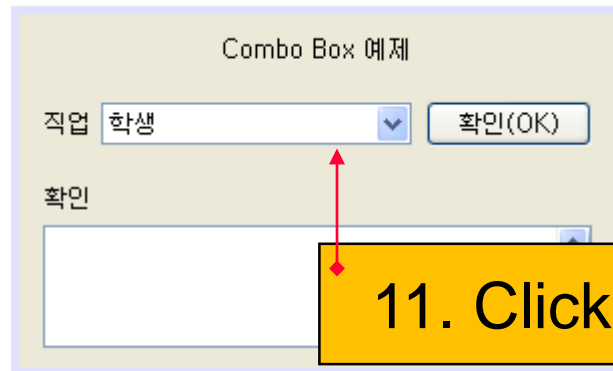
컨트롤	속성 지정
jLabel1	• Text: Combo Box 예제
jLabel2	• Text: 직업
jLabel3	• Text: 확인
jCombo Box	
jButton1	• Variable Name: btnOK • Text: 확인(OK)
jText Area	Text:



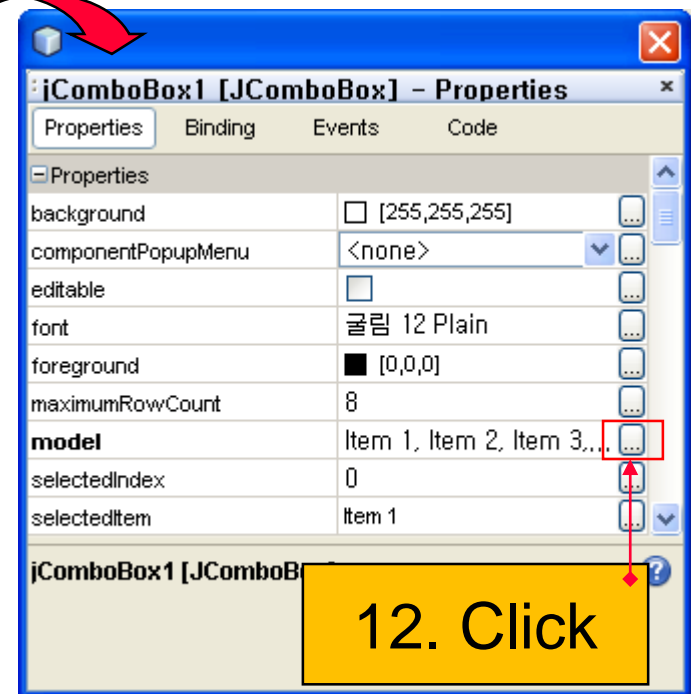


실습: Combo Box 컨트롤 (6)

❖ Combo Box 컨트롤 Item 추가



11. Click

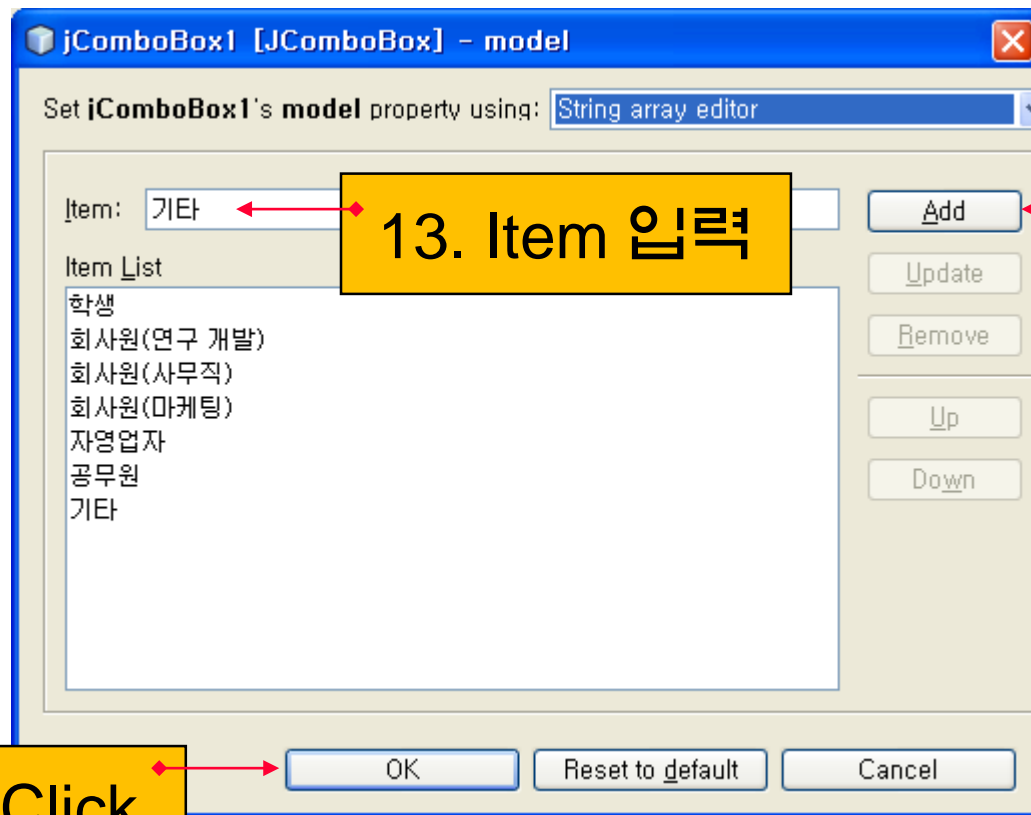


12. Click





실습: Combo Box 컨트롤 (7)



13. Item 입력

14. Click

15. Click





실습: Combo Box 컨트롤 (8)

❖ 입력(Input) Button Event Handler

MainFrame.java * - Editor

MainFrame.java * x

Source Design

```
19
20  /** ... */
25  Generated Code
101
102  private void btnOKActionPerformed(java.awt.event.ActionEvent evt) {
103      String strData = null;
104      strData = "회원의 직업은 " + jComboBox1.getSelectedItem().toString() + " 입니다. \n";
105      jTextArea1.setText(strData);
106  }
107
108  /**
109   * @param args the command line arguments
110   */
111  public static void main(String args[]) {
112      java.awt.EventQueue.invokeLater(new Runnable() {
113          public void run() {
114              new MainFrame().setVisible(true);
115          }
116      });
117  }
```

123:40 INS

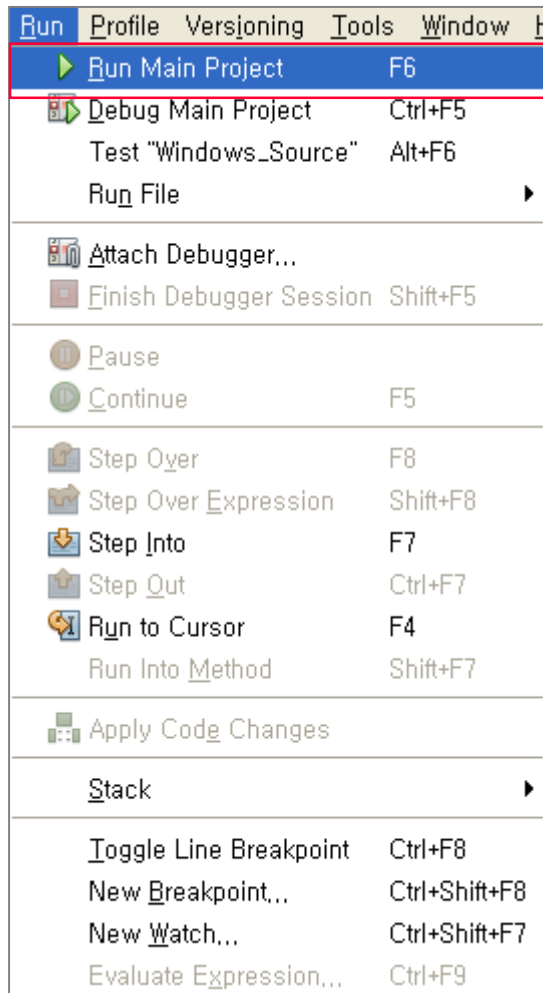
16. Coding



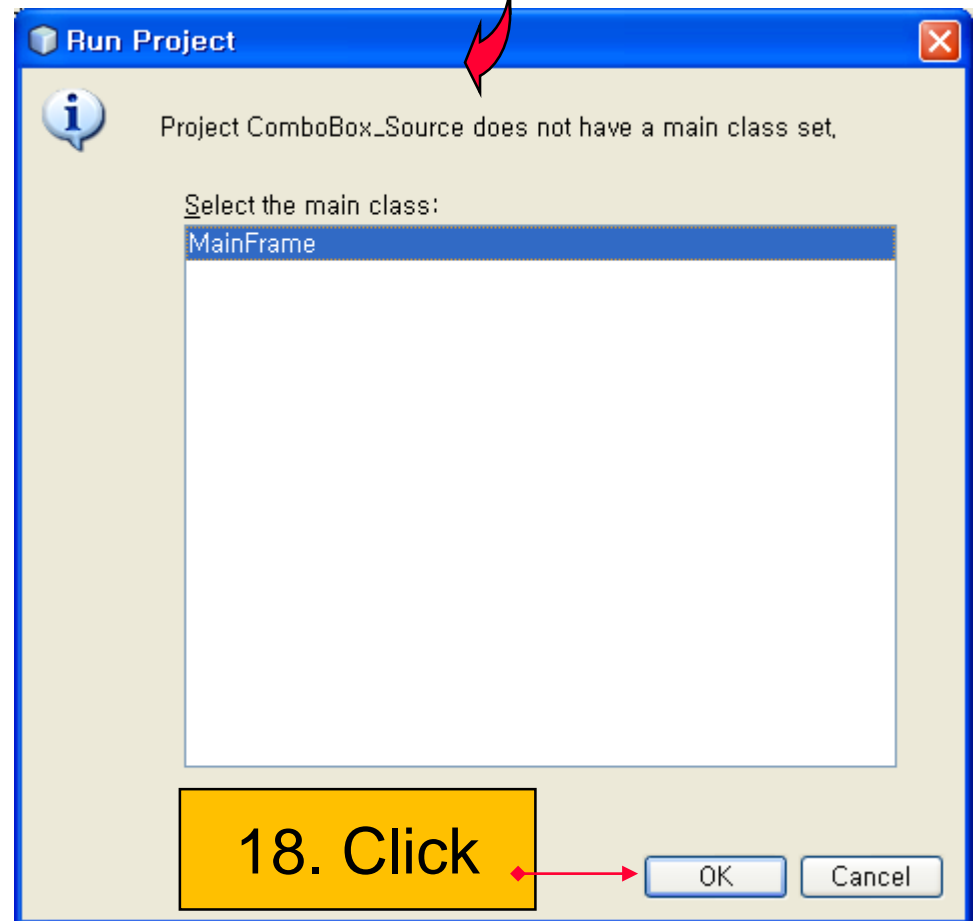


실습: Combo Box 컨트롤 (9)

실행



17. Click



18. Click



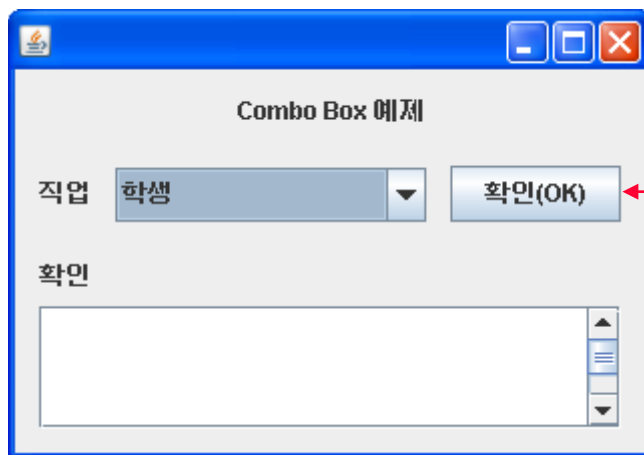


실습: Combo Box 컨트롤 (10)

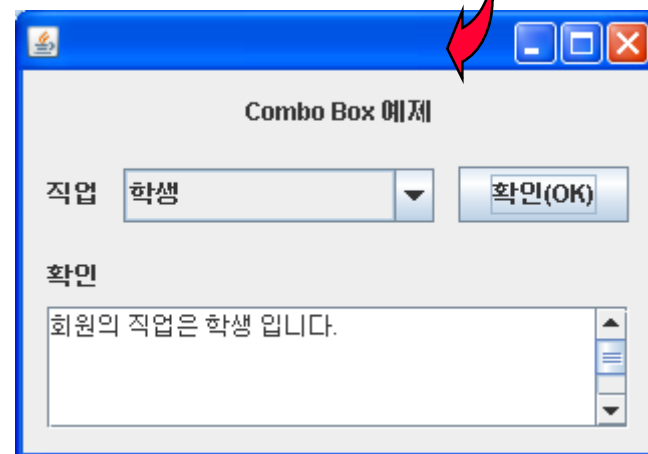
■ 실행 결과

❖ 확인(OK) Button Click

◆ Combo Box에서 선택한 직업을 Text Area에 표현한다.



19. Click



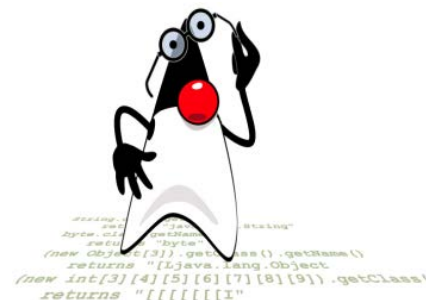
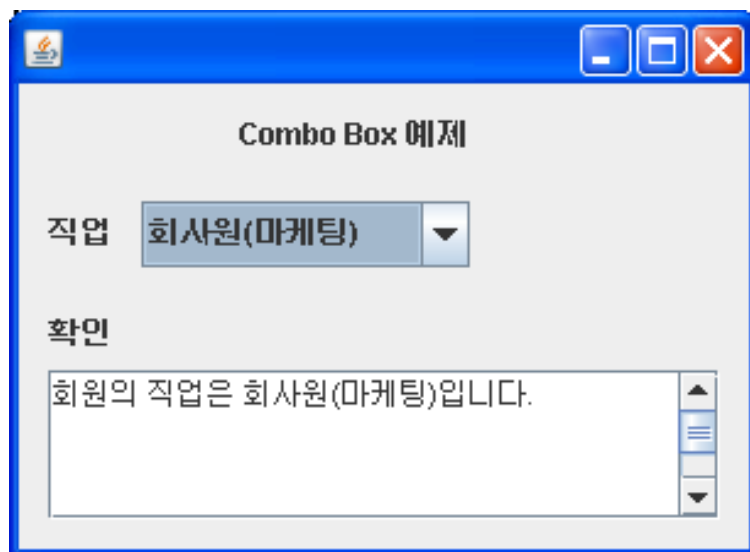


실습 2: Combo Box 컨트롤

❖ Project Name : ComboBox2_Source (실습시간 : 20분)

- Combo Box에서 해당 직업을 선택하면 Text Area에 표현한다.
- Event Handler

Event Handler	기능
ItemStateChanged(ItemEvent evt)	• Item 선택시 발생하는 event 처리





Choice 컨트롤

■ Choice 컨트롤

❖ 기능

- ◆ Drop-down 리스트를 구현할 때 사용
- ◆ addItem() 메서드를 사용하여 drop-down 항목 추가

❖ Method

Method	기능
int countItems()	Choice 컨트롤의 Item 수 반환
String getItem(int index)	Index에 해당하는 Item 반환
void addItem(String item)	Item을 choice Item으로 추가
String getSelcetItem()	선택된 Item 반환
int getSelcetedIndex()	선택된 항목의 Index 반환
void selcet(int index)	Index에 해당하는 Item 선택
void selcet(String str)	Str에 해당하는 Item 선택





List 컨트롤 (1)

List 컨트롤

❖ 기능

◆ 다중 Item이 리스트에 나타나도록 구현

❖ 생성자

생성자	설명
<code>JList();</code>	전달 인자가 없는 생성자
<code>JList(Object[] listData);</code>	배열 형태의 데이터를 갖는 list 생성
<code>JList(Vector<?> listData);</code>	벡터를 데이터로 갖는 list 생성
<code>JList(ListModel dataModel);</code>	데이터 모델을 갖는 list 생성





List 컨트롤 (2)

❖ Method

Method	기능
getSize()	List의 Item 수 반환
void setListData(Object[] listData)	Object 클래스로 List에 추가
void setListData(Vector<?> listData)	Vector 클래스로 List에 추가
int getSelectedIndex()	선택된 Item의 index 반환
int[] getSelectedIndices()	다중 선택 모드일때, 선택된 Item의 index 를 배열로 반환
int setSelectedIndex()	선택된 Item의 index 지정
int[] setSelectedIndices()	다중 선택 모드일때, 선택된 Item의 index 배열 지정

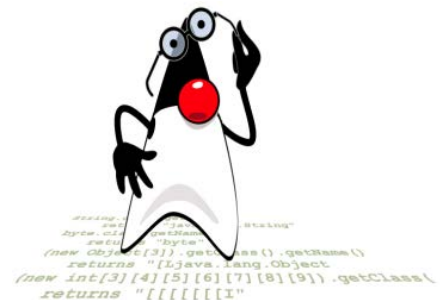
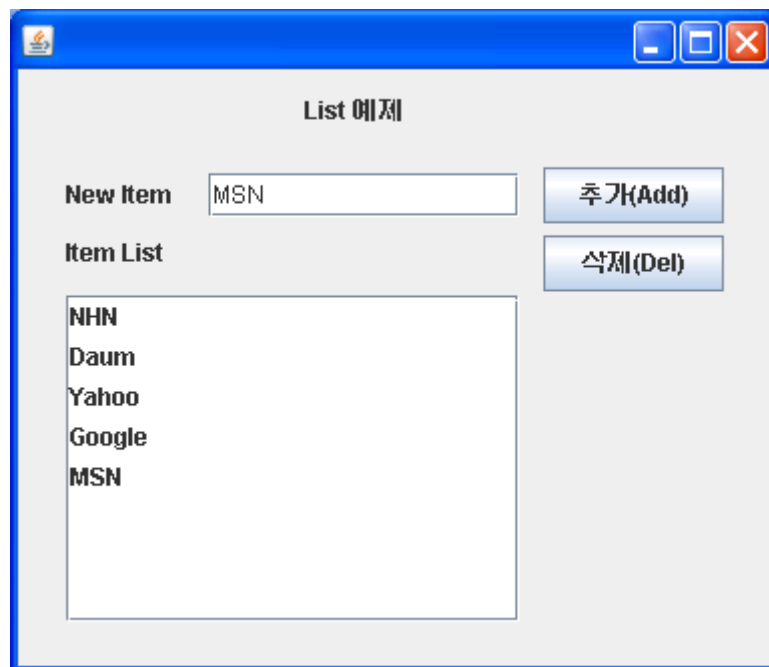




실습: List 컨트롤

❖ Project_Name: List_Source (실습시간 : 20분)

- 추가(Add) Button
 - Text Field에 입력한 New Item 을 Item List에 추가한다.
- 삭제(Del) Button
 - Item List에서 선택한 Item을 삭제한다.





실습: List 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: List 컨트롤 (2)

■ Project Name and Location

❖ Project name: List_Source

4. Project Name 입력

5. Click

Steps

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

Name and Location

Project Name: List_Source

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

Project Folder: C:\Java_Projects\Source\List_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 list_source, List_Source

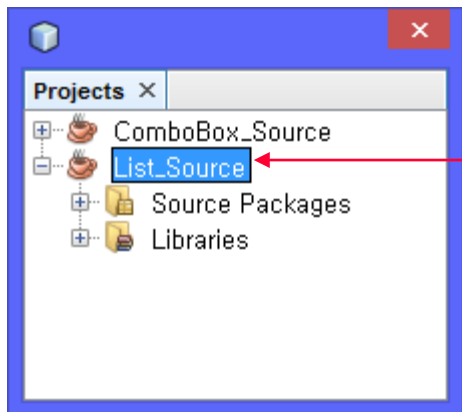
< Back Next > **Finish** Cancel Help



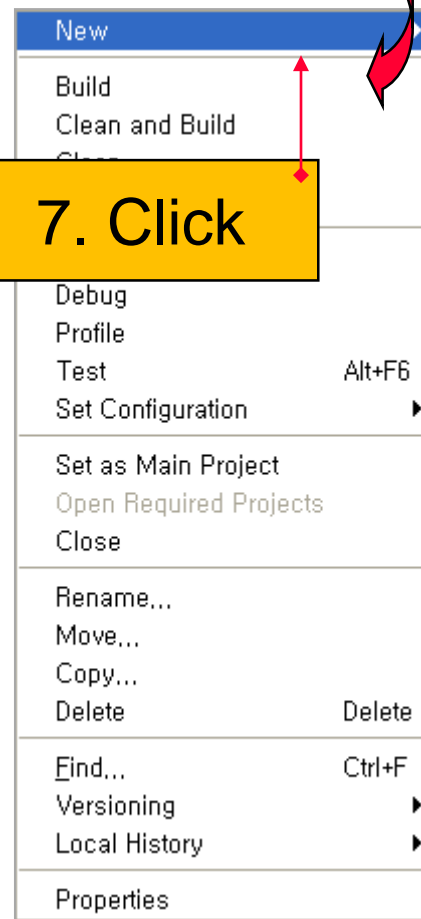


실습: List 컨트롤 (3)

JFrame Form 생성

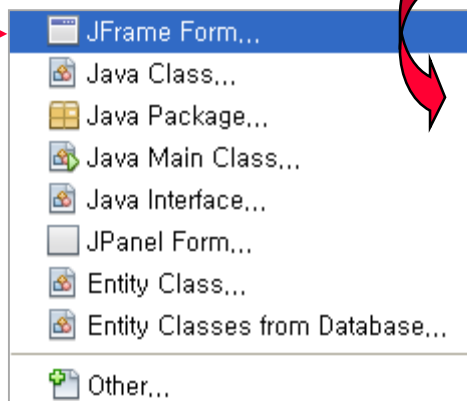


6. 마우스 오른쪽 버튼 Click



7. Click

8. Click





실습: List 컨트롤 (4)

■ JFrame Form Name 지정

❖ MainFrame.java 생성

Steps

1. Choose File Type
2. **Name and Location**

Name and Location

Class Name: MainFrame

Project: List_Source

Location: Source Packages

Package:

Created File: C:\Java_Projects\Source\List_Source\src\MainFrame.java

9. MainFrame 입력

10. Click

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

< Back Next > Finish Cancel Help



실습: List 컨트롤 (5)

■ 컨트롤 배치 및 속성 지정

List 예제

New Item

Item List

NHN
Daum
Yahoo
Google

추가(Add)

삭제(Del)

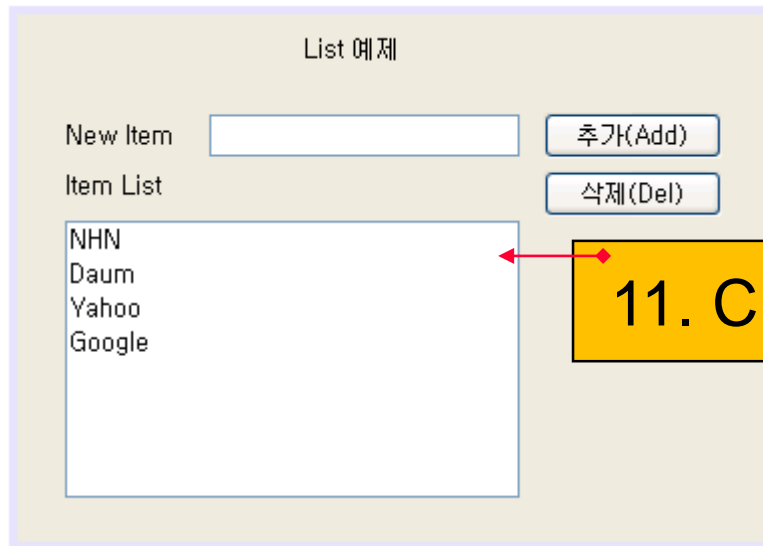
컨트롤	속성 지정
jLabel1	<ul style="list-style-type: none">• Variable Name: lblTitle• Text: List 예제
jText Field1	<ul style="list-style-type: none">• Variable Name: txtInput• Text:
jLabel2	<ul style="list-style-type: none">• Variable Name: lblItem• Text: New Item
jLabel3	<ul style="list-style-type: none">• Variable Name: lblItemlist• Text: Item List
List	<ul style="list-style-type: none">• Variable Name: lstCompany• model: 4개 Item 추가
jButton1	<ul style="list-style-type: none">• Variable Name: btnAdd• Text: 추가(Add)
jButton2	<ul style="list-style-type: none">• Variable Name: btnDel• Text: 삭제(Del)



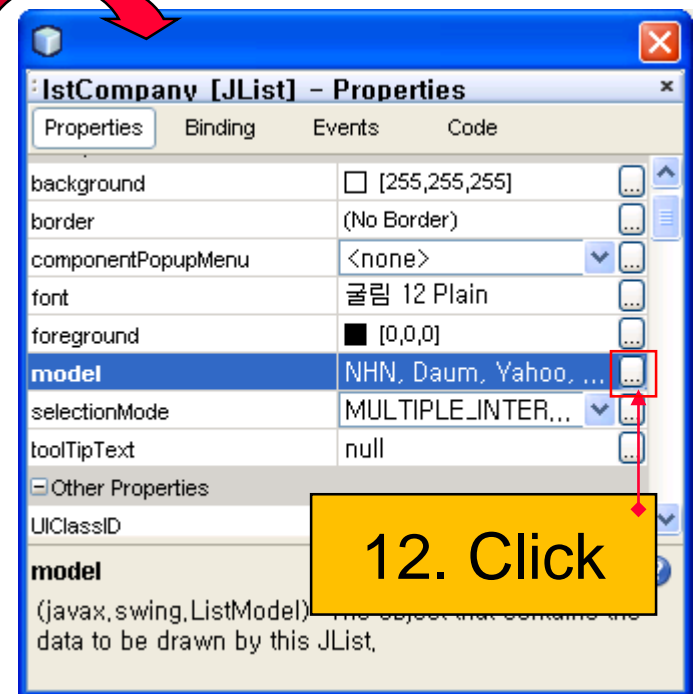


실습: List 컨트롤 (6)

❖ List 컨트롤 model 추가



11. Click

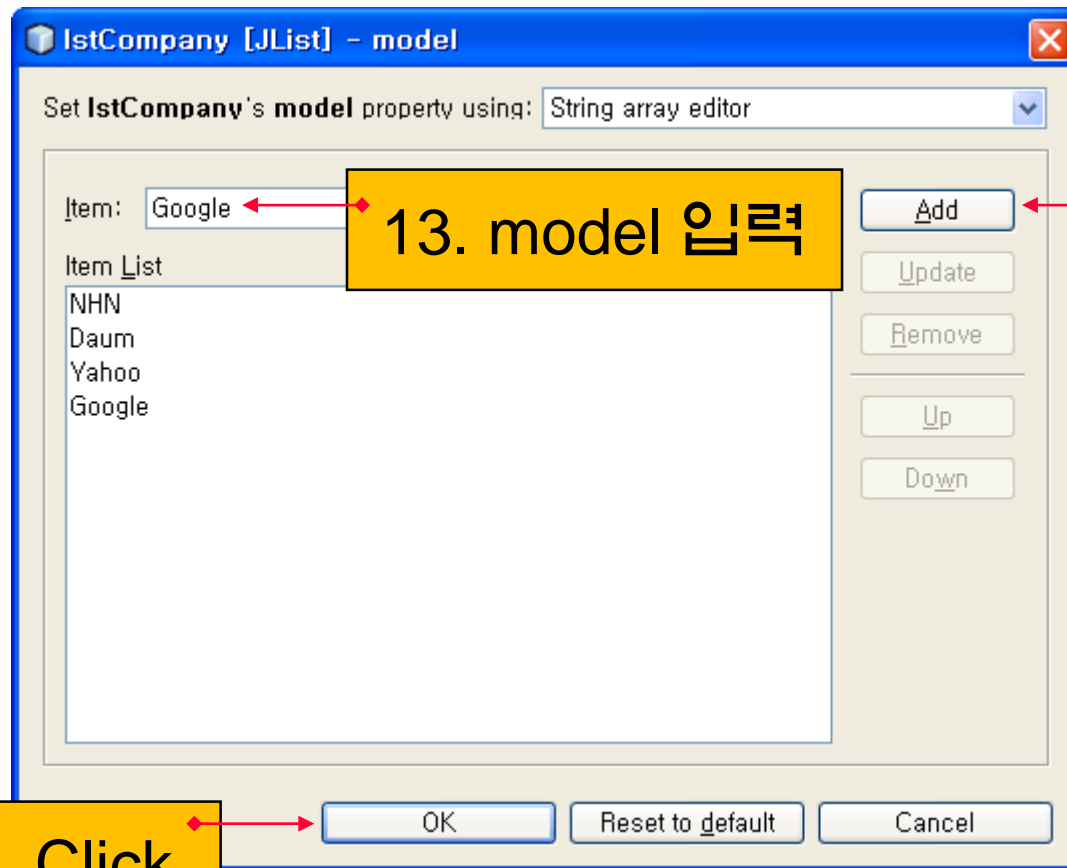


12. Click





실습: List 컨트롤 (7)



13. model 입력

14. Click

15. Click





실습: List 컨트롤 (8)

❖ 추가(Add) Button Event Handler

MainFrame.java - Editor

MainFrame.java x

Source Design

```
20
21 /** TH
22 * init
23 * WA
24 * alw
25 */
26 // <e
27 priva

117
118 private void btnAddActionPerformed(java.awt.event.ActionEvent evt) {
119     Vector myVC = new Vector();           //New Item 저장을 위한 Vector 클래스
120     int iListSize = 0;                     //List Size 저장
121
122     iListSize = lstCompany.getModel().getSize();           //List Size 반환
123     for(int idx = 0; idx < iListSize; idx++)               //List의 Item -> Vector 전송
124         myVC.addElement( lstCompany.getModel().getElementAt(idx));
125     myVC.addElement(txtInput.getText());                   //txtInput 문자열 -> Vector 추가
126     lstCompany.setListData(myVC);                         //List의 Data 지정
127 }
```

148:57 INS

16. Coding



실습: List 컨트롤 (9)

❖ 삭제(Del) Button Event Handler

The screenshot displays a Java IDE with a source code editor. The code is for a class named `IstCompany` and a method `btnDelActionPerformed`. The code is organized into two highlighted sections:

```
lstCompany.setModel(new javax.swing.AbstractListModel() {  
    String[] strings = { "NHN", "Daum", "Yahoo", "Google" };  
    public int getSize() { return strings.length; }  
    public Object getElementAt(int i) { return strings[i]; }  
});  
jScrollPane1.setViewportViewView(lstCompany);  
  
private void btnDelActionPerformed(java.awt.event.ActionEvent evt) {  
    Vector myVC = new Vector();           //New Item 저장을 위한 Vector 클래스  
    int iList_index = 0;                  //List Item index 저장  
    int iListSize = 0;                    //List Size 저장  
  
    iListSize = lstCompany.getModel().getSize();           //List Size 반환  
    for(int idx = 0; idx < iListSize; idx++)                //List의 Item -> Vector 추가  
        myVC.addElement( lstCompany.getModel().getElementAt(idx));  
  
    iList_index = lstCompany.getSelectedIndex();           //List에서 선택한 Item의 index 값  
    myVC.removeElementAt(iList_index);                     //Vector에서 해당 index 값 삭제  
    lstCompany.setListData(myVC);                         //List의 Data 지정  
}
```

The IDE interface includes a project explorer on the left showing 'MainFrame.java', a source/design view toggle, a line number column, and a status bar at the bottom indicating '148:57' and 'INS'.

17. Coding





실습: List 컨트롤 (10)

■ 실행 결과

❖ 추가(Add) Button

◆ Text Field에 입력한 New Item 을 Item List에 추가한다.

18. Item 입력

19. Click



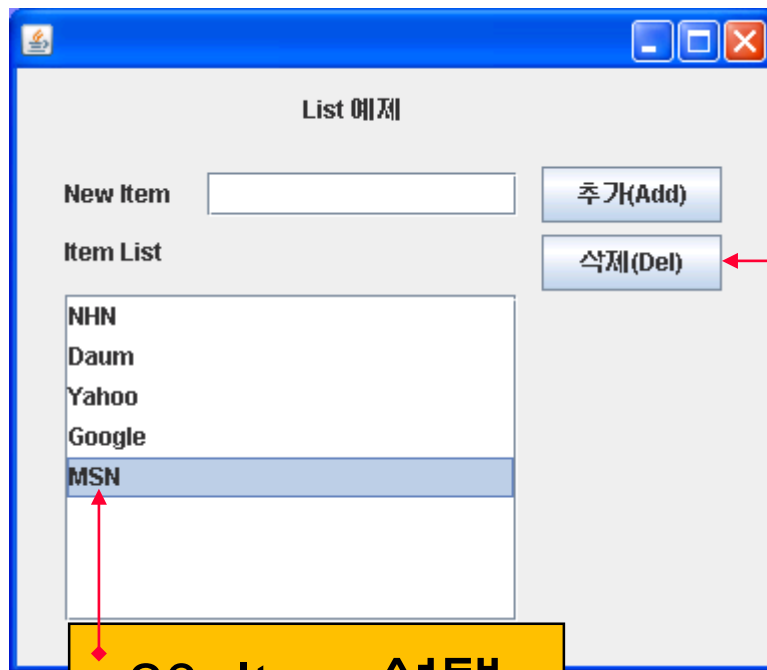


실습: List 컨트롤 (11)

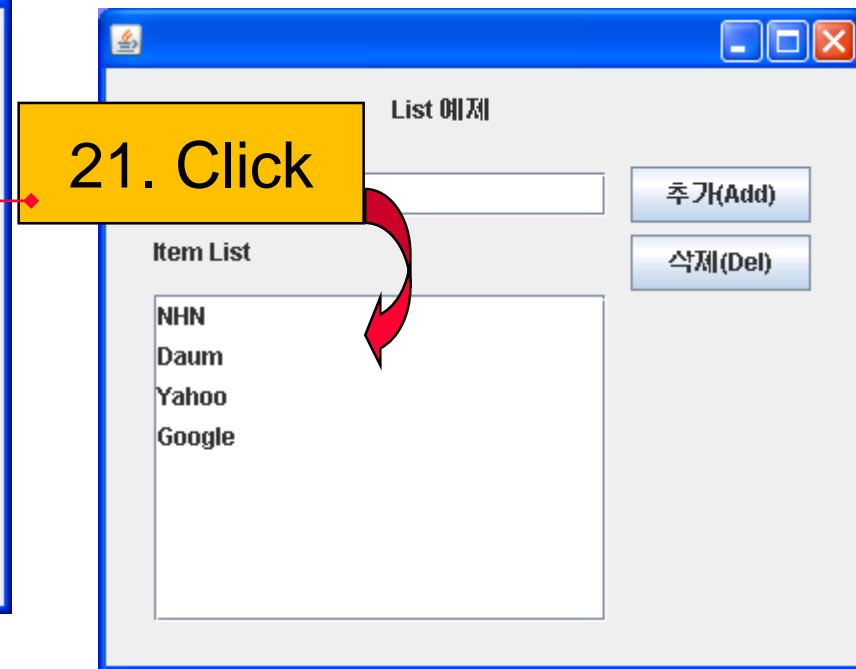
■ 실행 결과

❖ 삭제(Add) Button

◆ Item List에서 선택한 Item을 삭제한다.



20. Item 선택



21. Click





Homework # 2

❖ 회원 가입 모듈 구현 (Project 명: Member_ID)

■ [회원 가입] 버튼 Click

- 비밀번호가 일치하면 아이디, 비밀번호, 이름, 회원 구분, 관심 분야, 직업, 결제 방법을 **member.txt** 파일에 저장한다.

회원 가입

아이디: kdgong 중복 확인

비밀번호: ****

비밀번호 확인: ****

성명: 홍길동

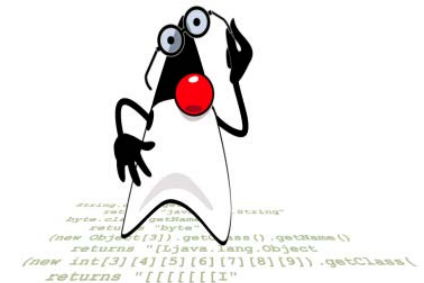
회원 구분: ☒ 정회원 ☐ 준회원 ☐ 학생회원

관심 분야: ☒ 수영 ☐ 게임 ☐ 골프
☐ 스키 ☐ 독서 ☒ 테니스

직업: 학생

결제 방법: 현금
카드
포인트

회원 가입



member.txt - 메모장

파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

kdhong 1234 홍길동 정회원 33 학생 현금





학습 요약

- CheckBox 컨트롤
- RadioButton 컨트롤
- Choice와 Combo Box 컨트롤
- List 컨트롤

