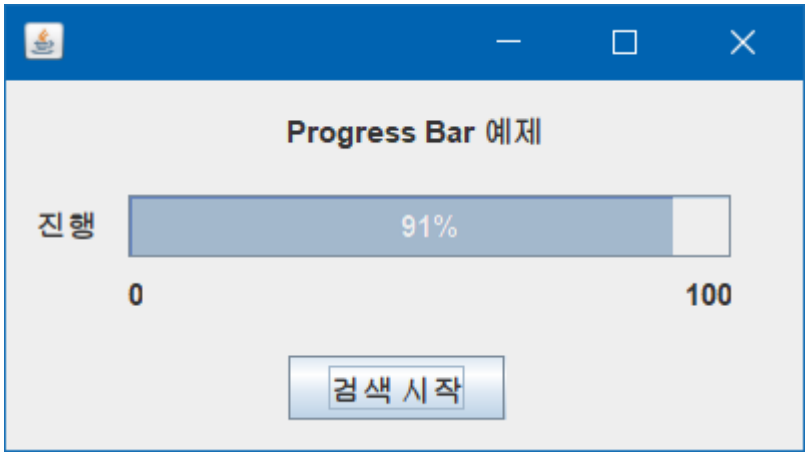
GUI Control III



학습 목표

■ 이 강의를 마치면 학생들은

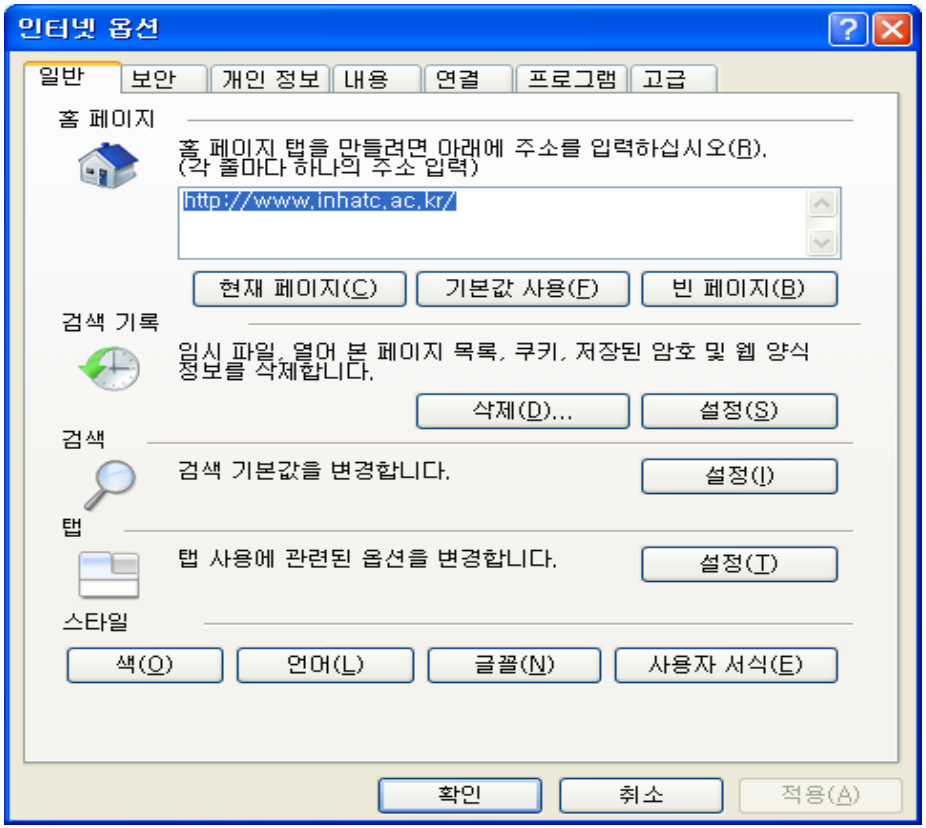
- ❖ TabbedPane Control Function에 대하여 설명할 수 있다.
- ❖ Progress Bar Control Function에 대하여 설명할 수 있다.
- ❖ Slider Control에 대하여 설명할 수 있다.

Tab Control

Tab Control

❖ Function

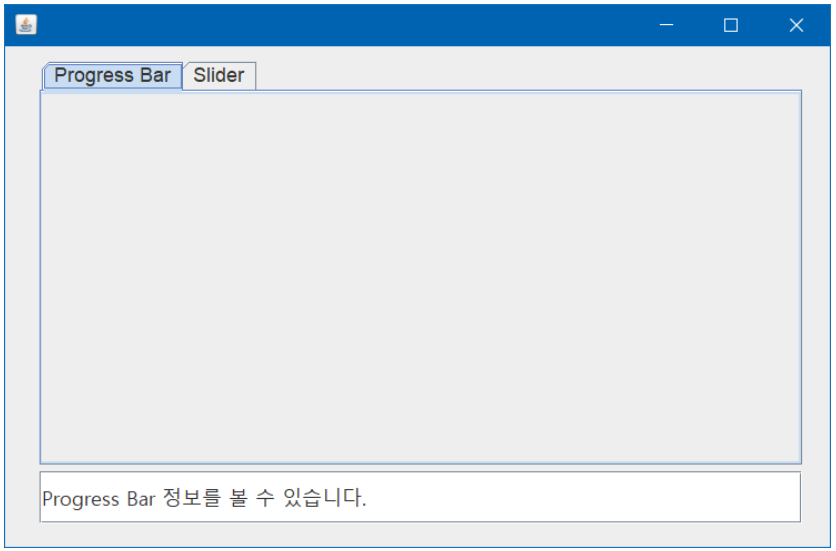
◆ 한 화면에 다양한 Function 표현

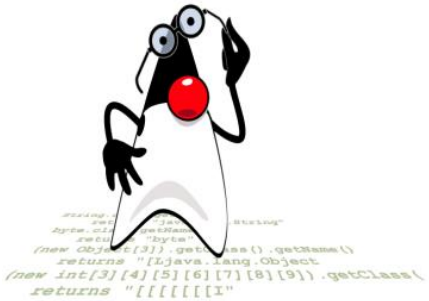


Practice: Tab Control

❖ Project Name: Tab_Source(Practice Time : 20 min)

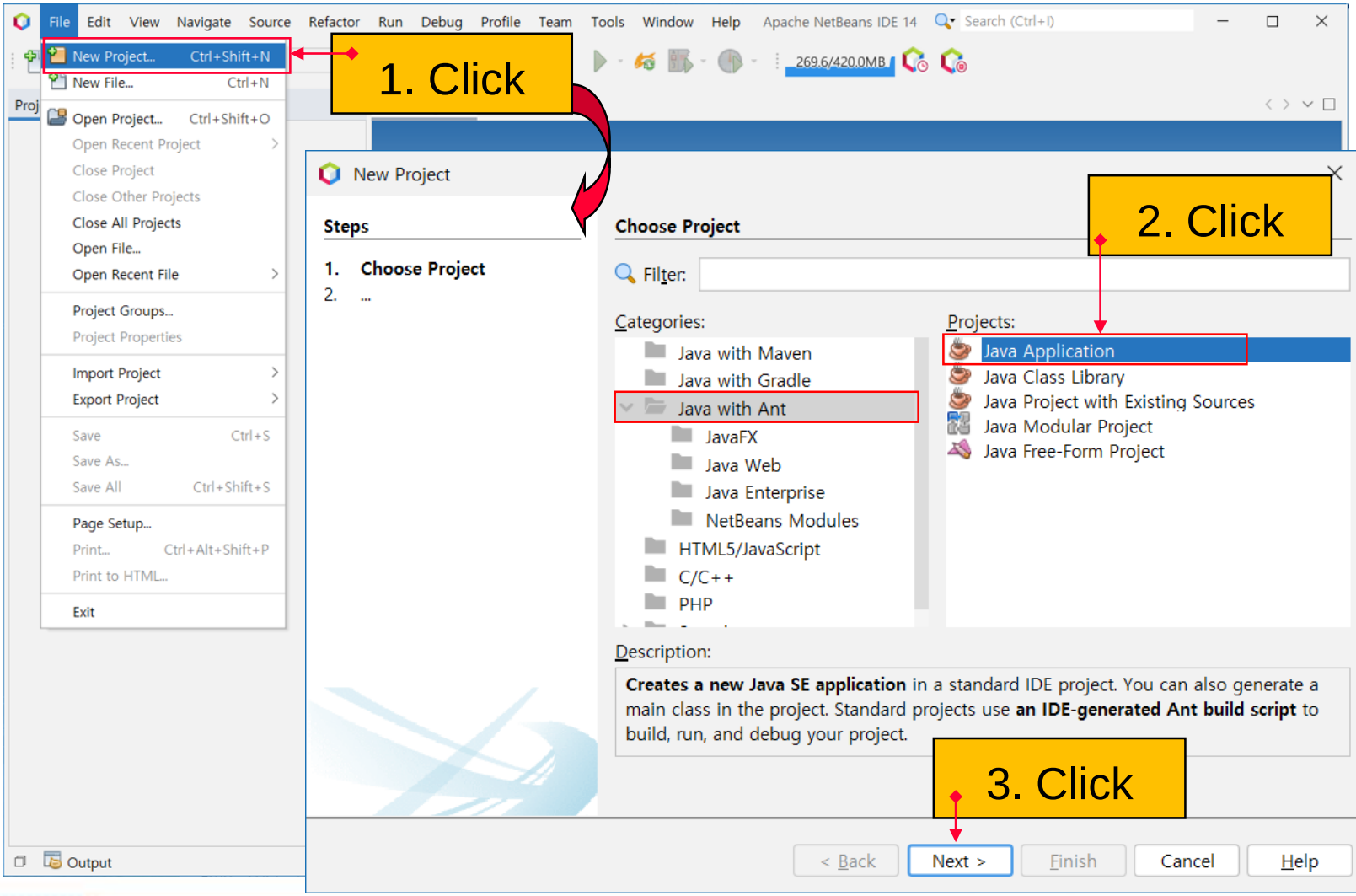
- TabbedPane 버튼 Click
  - Pane의 Title을 상태 정보 출력 TextField에 출력한다.





Practice 1 : Tab Control (1)

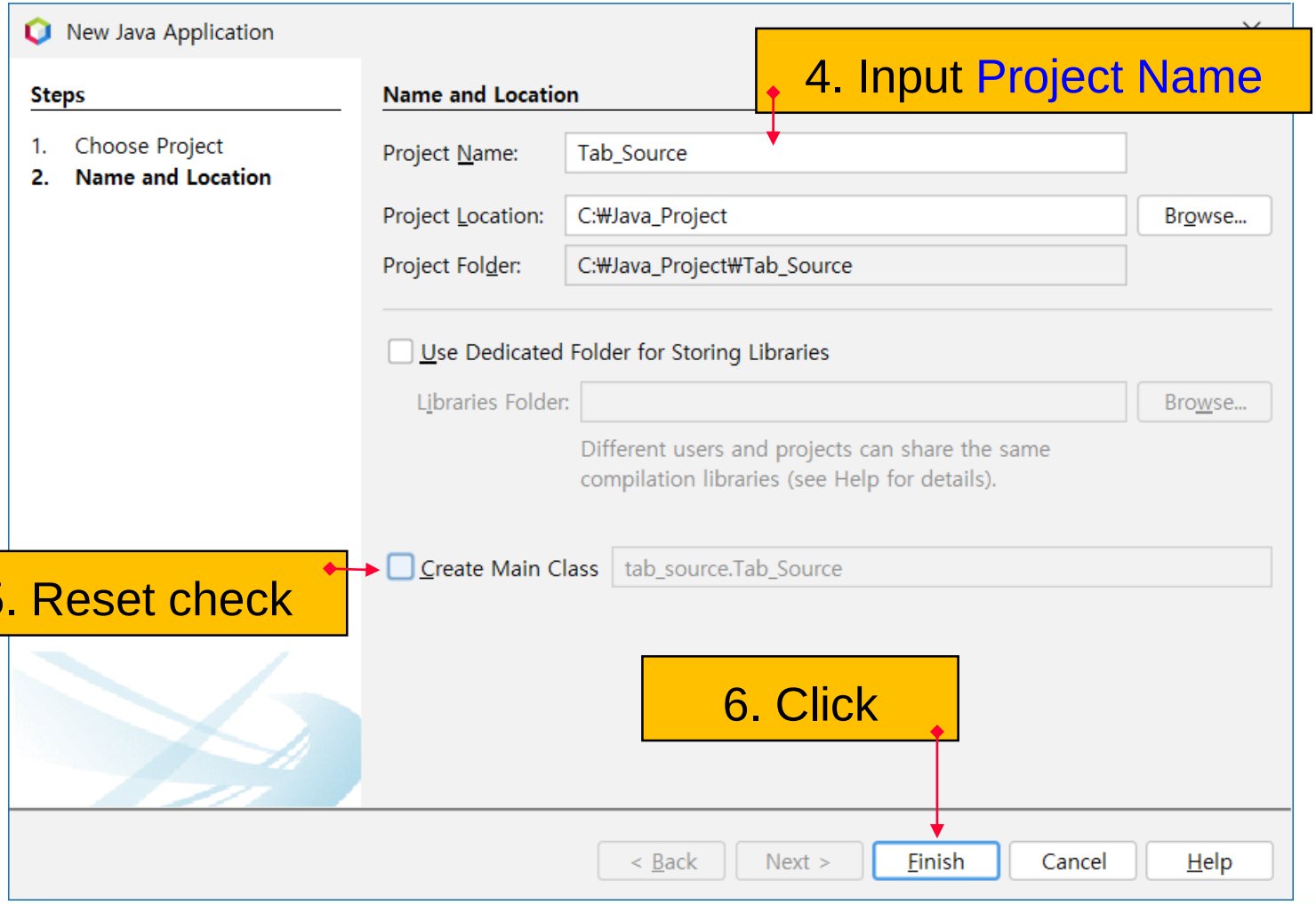
Create Project



Practice 1 : Tab Control (2)

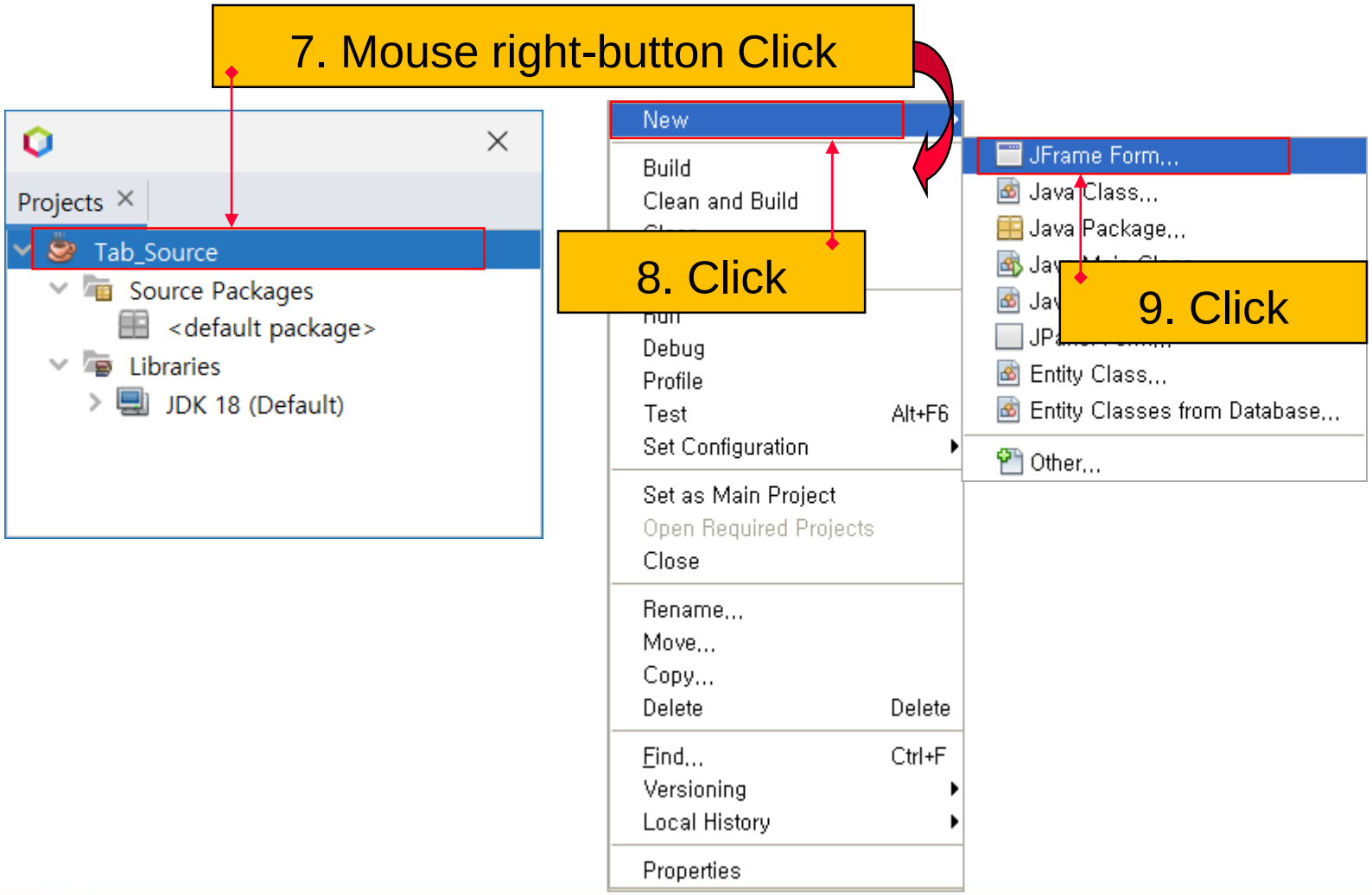
Project Name and Location

- Project name: Tab_Source



Practice 1 : Tab Control (3)

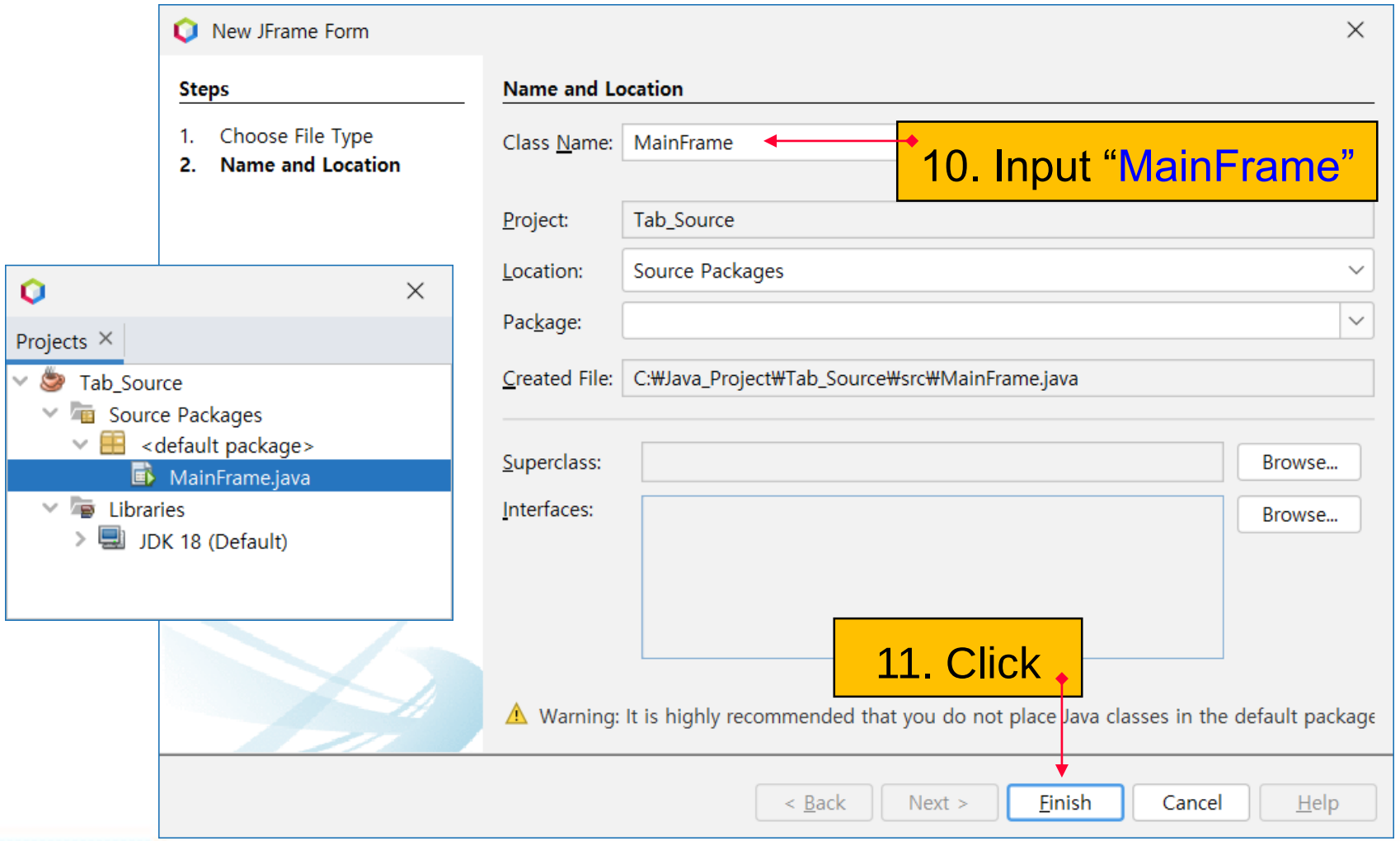
- Create JFrame Form



Practice 1 : Tab Control (4)

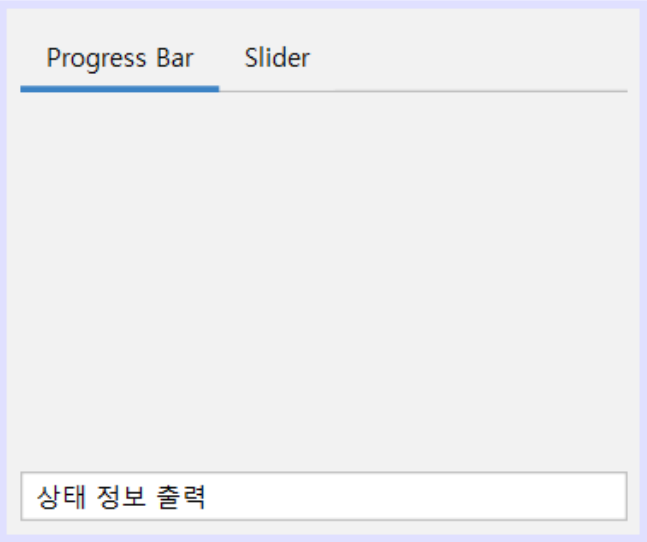
Setting JFrame Form Name

❖ Create MainFrame.java



Practice 1 : Tab Control (5)

Control Layout & Property Setting

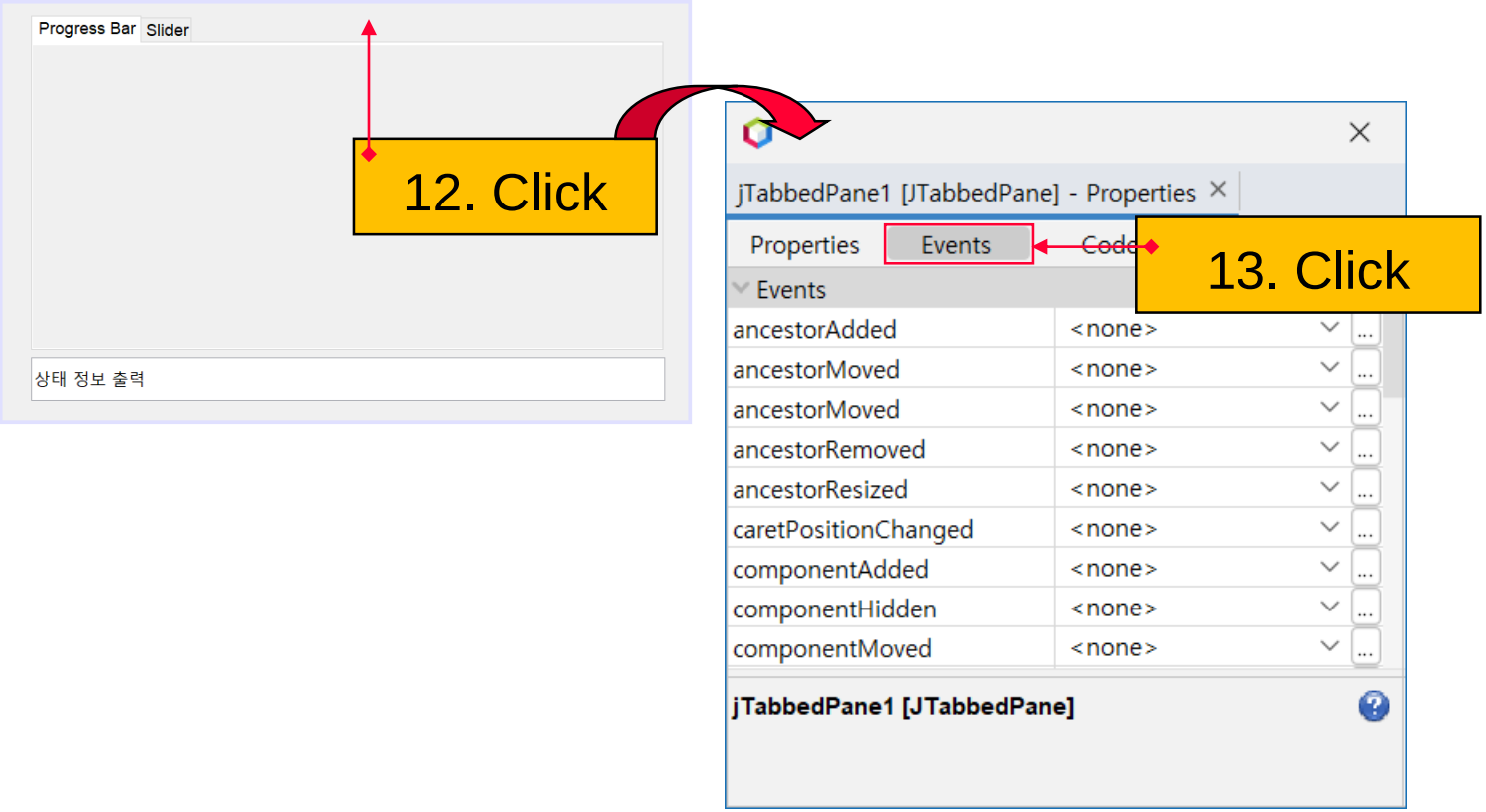


12. UI Design

Control	Properties Setting
jTabbedPane1	
jPanel1	• Tab Title: ProgressBar
jPanel2	• Tab Title: Slider
jTextField1	• Variable Name: txtStateInfo • Text: 상태 정보 출력

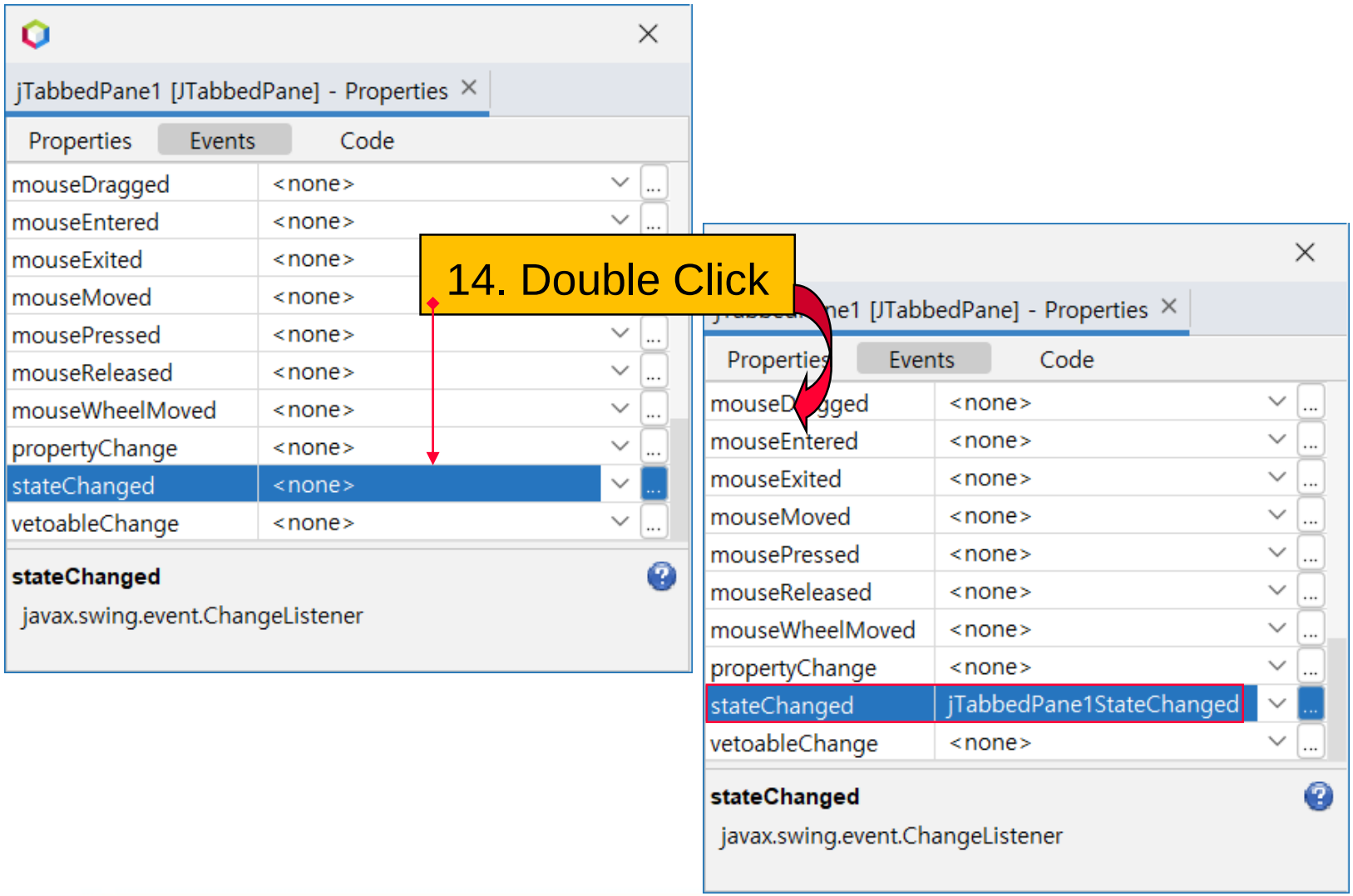
Practice 1 : Tab Control (6)

- jTabbedPane1 Control Event 지정



Practice 1 : Tab Control (7)

- jTabbedPane1 Control StateChanged() Event 지정



Practice 1 : Tab Control (8)

❖ jTabbedPane1StateChanged() Event Handler

MainFrame.java - Editor

MainFrame.java

```
18      }
72
        private void jTabbedPane1StateChanged(javax.swing.event.ChangeEvent evt) {
            int iTabindex = 0;                  // jTabbedPane1 index 저장
            String strTabTitle = null;          // jTabbedPane1 Title 저장
76
77          iTabindex = jTabbedPane1.getSelectedIndex();            // jTabbedPane1 index 반환
78          strTabTitle = jTabbedPane1.getTitleAt(iTabindex);       // jTabbedPane1 Title 반환
79          txtStateInfo.setText( strTabTitle + " 정보를 볼 수 있습니다.");
80      }
81
82      /**
83       * @param args the command line arguments
84       */
85      public static void main(String args[]) {
            java.awt.EventQueue.invokeLater(new Runnable() {
                public void run() {
88                  new MainFrame().setVisible(true);
89              }
90          });
91      }
```

15. Coding

16:25 INS

Practice: Tab Control (9)

- jTabbedPane Control Declaration Code

MainFrame.java - Editor

MainFrame.java

Source | Design | History

```
82      /**
83       * @param args the command line arguments
84       */
85      public static void main(String args[]) {
            java.awt.EventQueue.invokeLater(new Runnable() {
                public void run() {
                    new MainFrame().setVisible(true);
                }
            });
        }

        // Variables declaration - do not modify
        private javax.swing.JTabbedPane jTabbedPane1;
        private javax.swing.JTabbedPane jTabbedPane2;
        private javax.swing.JTabbedPane jTabbedPane3;
        private javax.swing.JTextField txtStateInfo;
        // End of variables declaration

    }
```

102:1 INS

Practice 1 : Tab Control (10)

❖ jTabbedPane Control Properties Setting Code

MainFrame.java - Editor

MainFrame.java

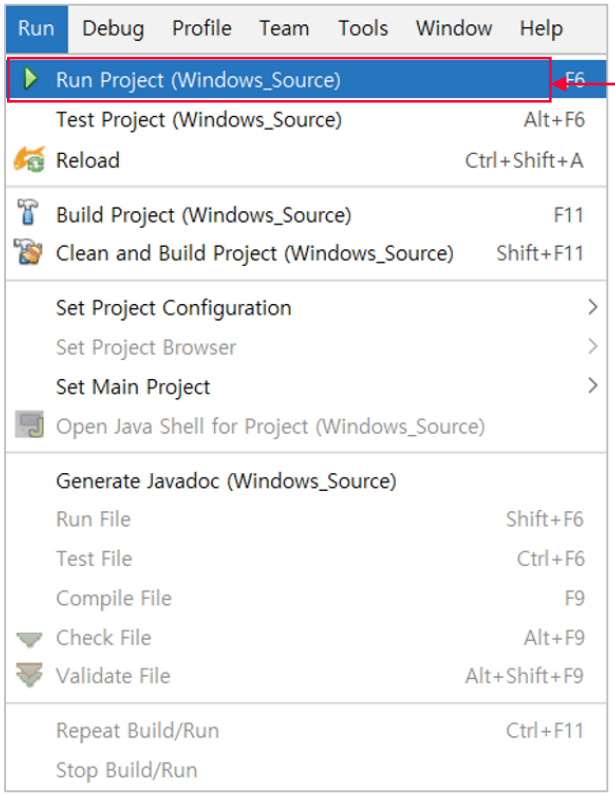
Source | Design | History

```
25  // <editor-fold defaultstate="collapsed" desc="Generated Code">
26  private void initComponents() {
27
28      jTabbedPane1 = new javax.swing.JTabbedPane();
29      jTabbedPane2 = new javax.swing.JTabbedPane();
30      jTabbedPane3 = new javax.swing.JTabbedPane();
31      txtStateInfo = new javax.swing.JTextField();
32
33      setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
34
35      jTabbedPane1.addChangeListener(new javax.swing.event.ChangeListener() {
36          public void stateChanged(javax.swing.event.ChangeEvent evt) {
37              jTabbedPane1StateChanged(evt);
38          }
39      });
40      jTabbedPane1.addTab("Progress Bar", jTabbedPane2);
41      jTabbedPane2.getAccessibleContext().setAccessibleParent(this);
42
43      jTabbedPane1.addTab("Slider", jTabbedPane3);
44
45      txtStateInfo.setText("상태 정보 출력");
```

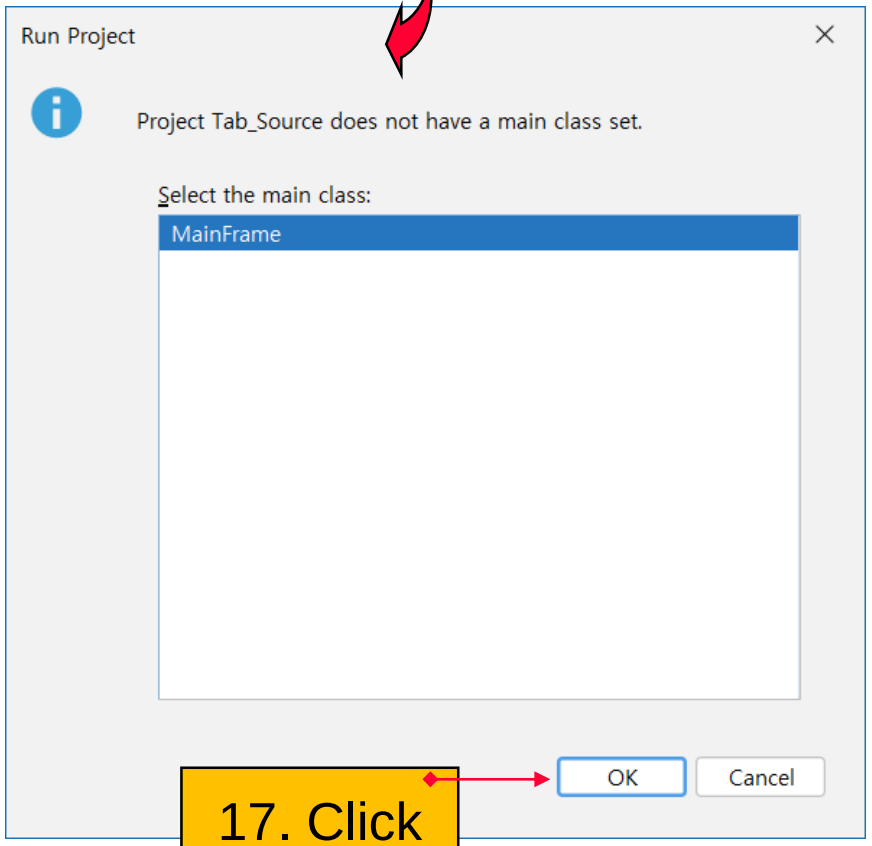
101:1 | INS | A breakpoint cannot be set at this location.

Practice 1 : Tab Control (11)

Run



16. Click

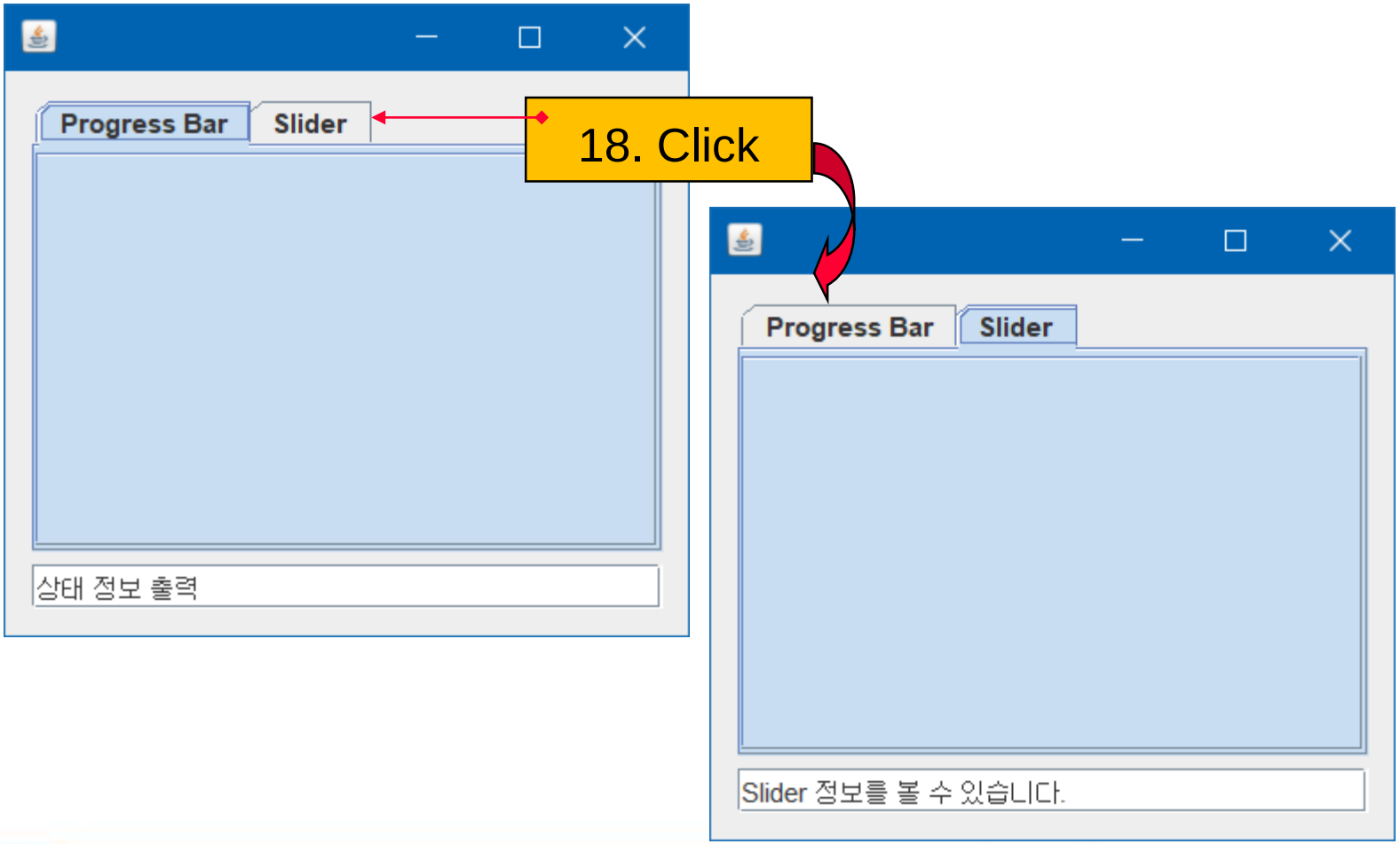


17. Click

Practice 1 : Tab Control (12)

Run

❖ Tab Click

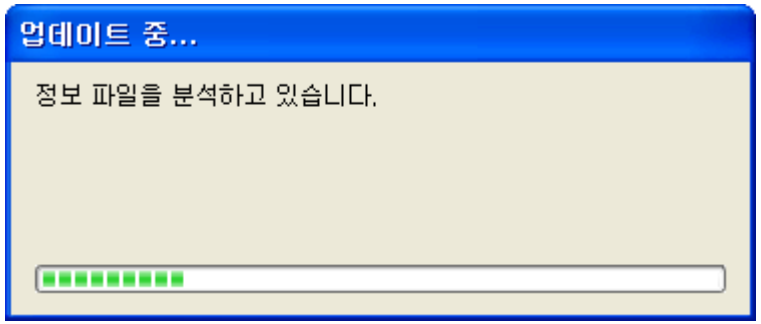


Progress Bar Control

ProgressBar Control

❖ Function

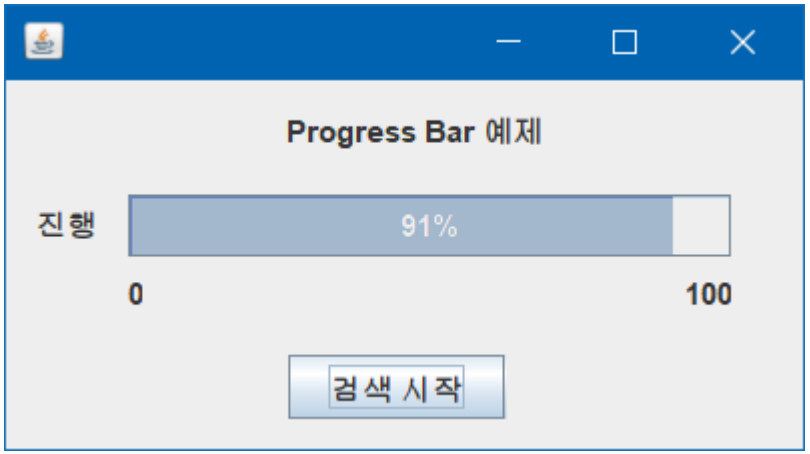
◆ 작업 진행 상태 표현

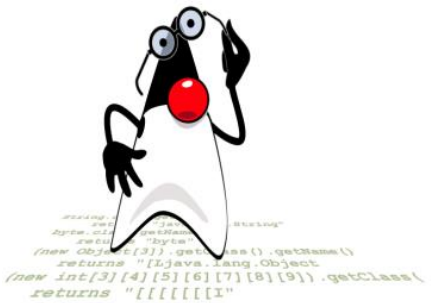


Practice 2 : Progress Bar Control

❖ Project Name: ProgressBar_Source(Time : 20 min)

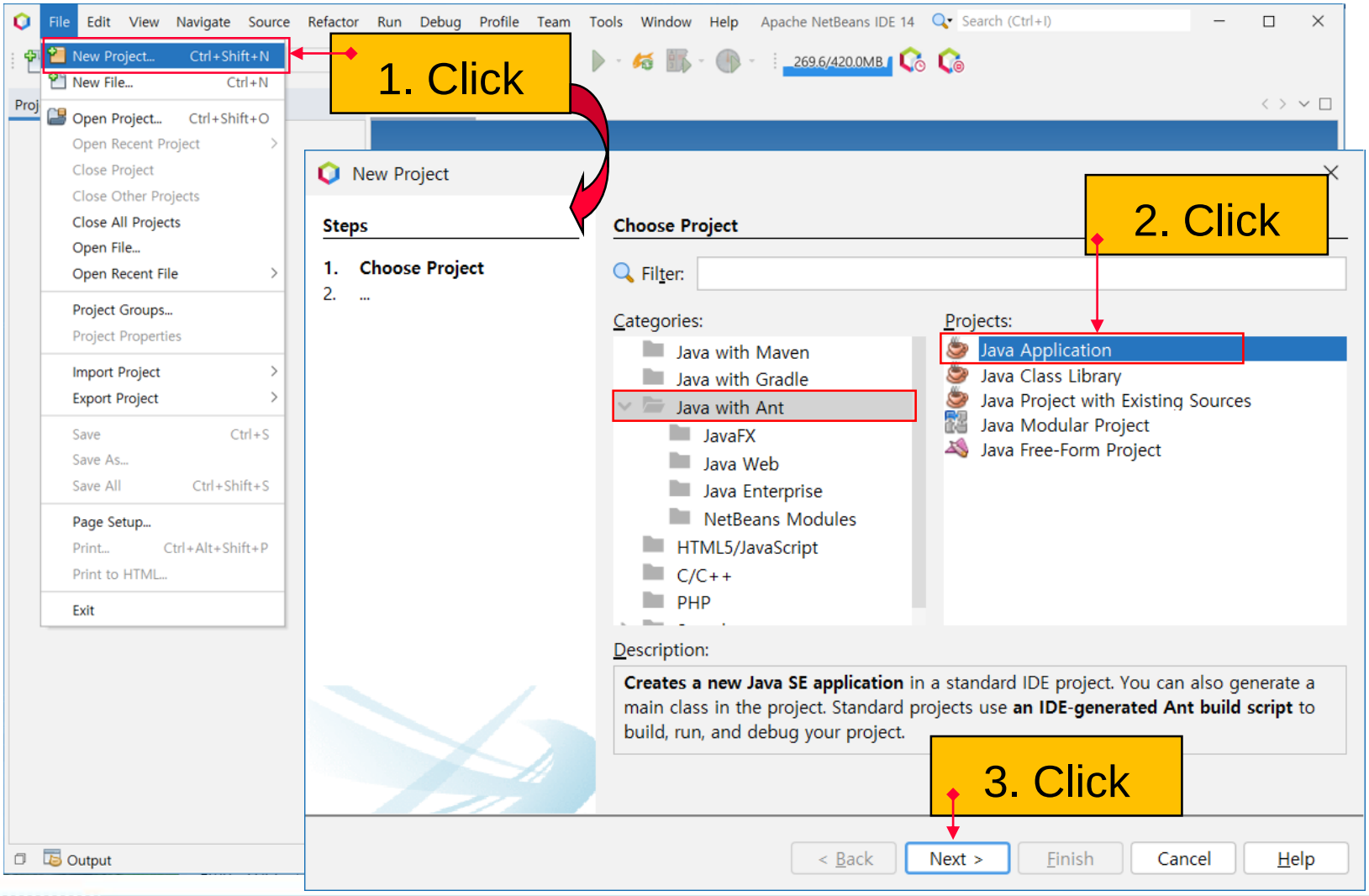
- [검색 시작] Button Click
  - 진행 상태를 Progress Bar에 나타낸다.





Practice 2 : Progress Bar Control (1)

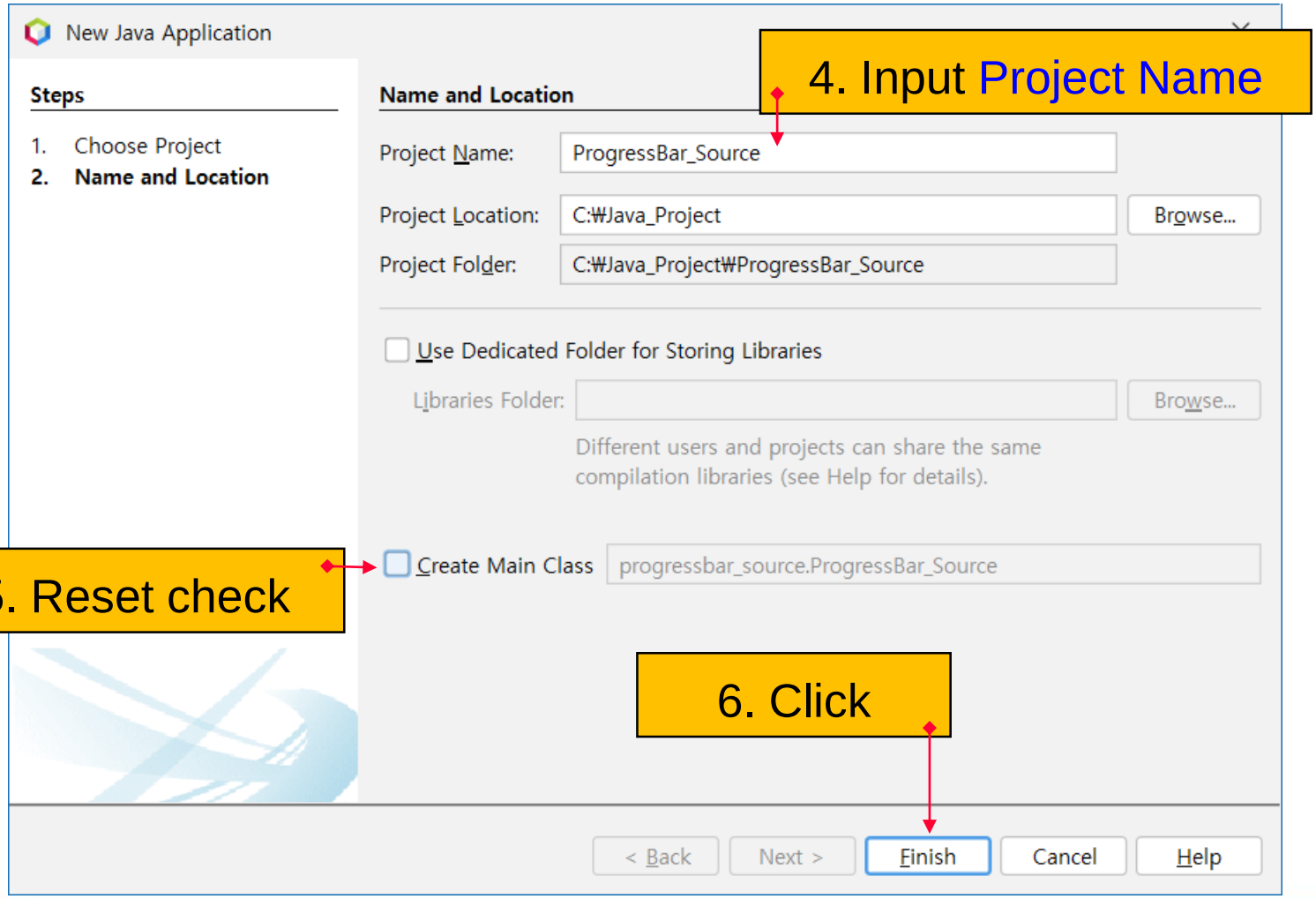
Create Project



Practice 2 : Progress Bar Control (2)

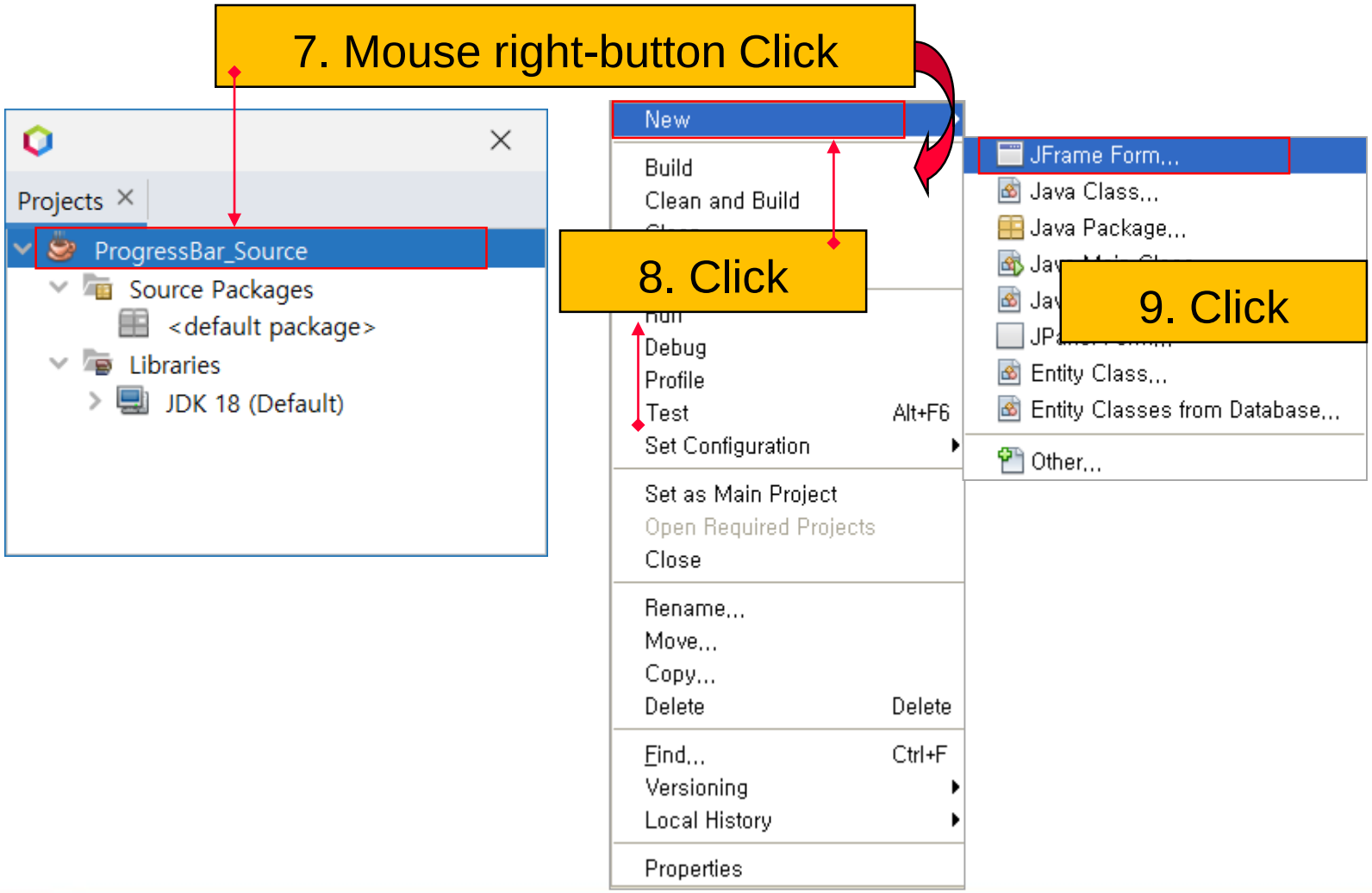
Project Name and Location

- Project name: ProgressBar_Source



Practice 2 : Progress Bar Control (3)

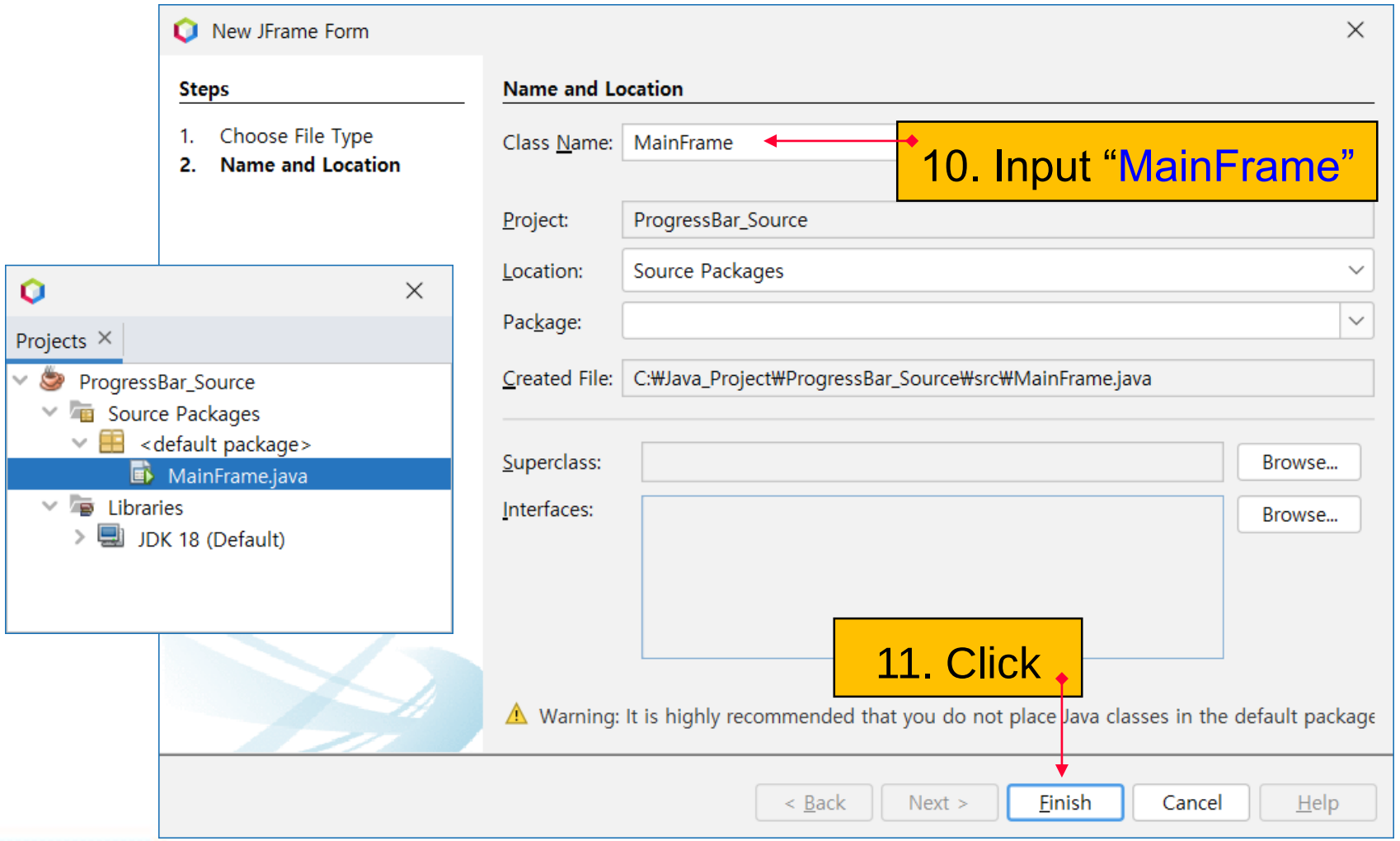
- Create JFrame Form



Practice 2 : Progress Bar Control (4)

Setting JFrame Form Name

❖ Create MainFrame.java



Practice 2 : Progress Bar Control (5)

Control Layout & Property Setting

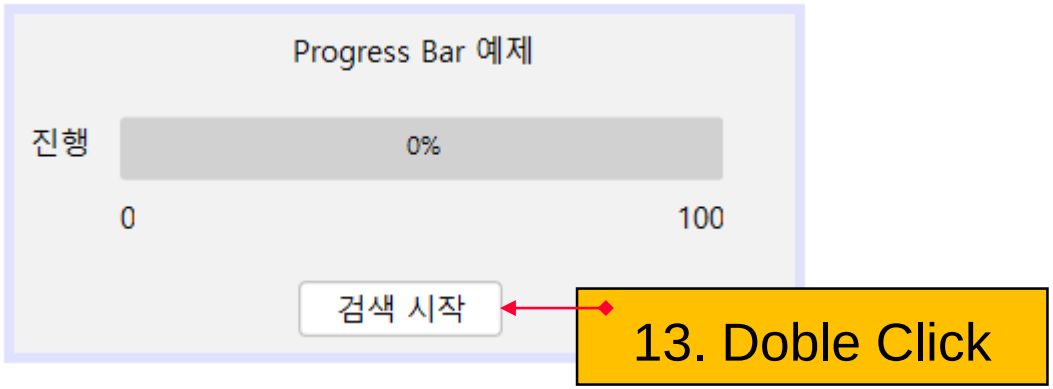
12. UI Design

Control	Properties Setting
jProgressBar	•maximun: 100 •minimum: 0 •borderPainted: ☑ •string: 0% •stringPainted: ☑

Control	Properties Setting
jLabel1	• Variable Name: lblTitle • Text: Progress Bar 예제
jLabel2	• Variable Name: lblProgress • Text: 진행
jLabel3	• Variable Name: lblStartVal • Text: 0
jLabel4	• Variable Name: lblEndVal • Text: 100
jButton1	• Variable Name: btnSearch • Text: 검색 시작

Practice 2 : Progress Bar Control (6)

- btnSearchActionPerformed() Event Handler Setting



Practice 2 : Progress Bar Control (7)

- btnSearchActionPerformed() Event Handler

MainFrame.java - Editor

MainFrame.java

Source | Design | History

```
1
2   import javax.swing.JProgressBar;
3
4   public class MainFrame extends javax.swing.JFrame {
5
6       /** Creates new form MainFrame */
7       public MainFrame() {
8           initComponents();
9       }
10
11      /** This method is called from within the constructor to ...5 lines */
16      Generated Code
92
        private void btnSearchActionPerformed(java.awt.event.ActionEvent evt) {
94          Thread objThread = new PBarThread(jProgressBar1);    //Thread Object
95          objThread.start();
96      }
```

134:5 INS

14. Coding

Practice 2 : Progress Bar Control (8)

❖ Progress BarThread

15. Coding

MainFrame.java - Editor

MainFrame.java

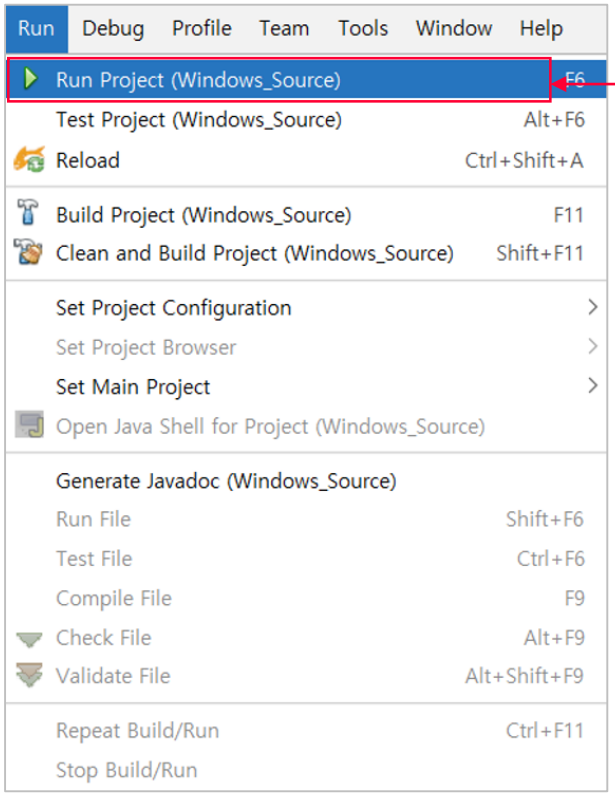
Source | Design | History

```
98      //Inner Class PBarThread
99      public class PBarThread extends Thread {
100        final static int DELAY = 500;                    //Thread delay 0.5 second
101        JProgressBar objProgressBar;
102
103        //Constructor
104        public PBarThread(JProgressBar objPB) {
105          objProgressBar = objPB;
106          objProgressBar.setStringPainted(true);         //Show ProgressBar Value
107        }
108
           public void run() {
110            int minValue = objProgressBar.getMinimum();  //ProgressBar.minimum value
111            int maxValue = objProgressBar.getMaximum();  //ProgressBar.maximum value
112
113            for (int idx = minValue; idx < maxValue; idx++) {
114              try {
115                int iValue = objProgressBar.getValue();
116                objProgressBar.setValue(iValue + 1);     //Setting jProgressBar Value
                   Thread.sleep(DELAY);                    //Thread Sleep
118              } catch (InterruptedException ignoredException) {
119              }
120            } //end of for loop
121        } //end of run() method
122    } //end of PBarThread Class
```

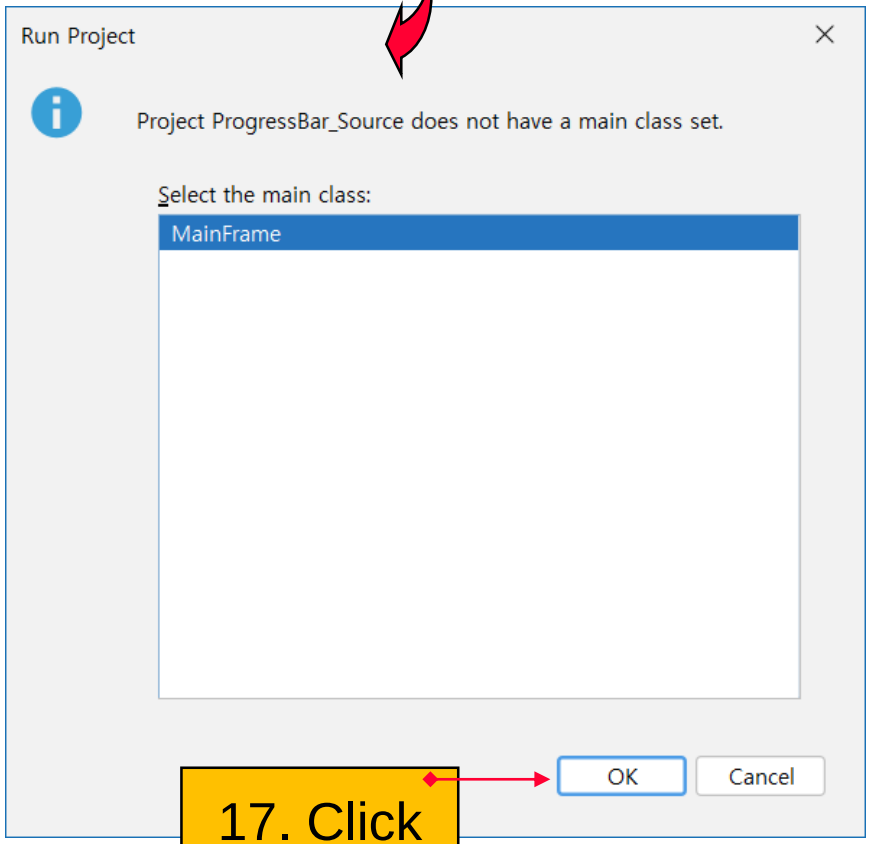
1:1 INS

Practice 2 : Progress Bar Control (9)

Run



16. Click

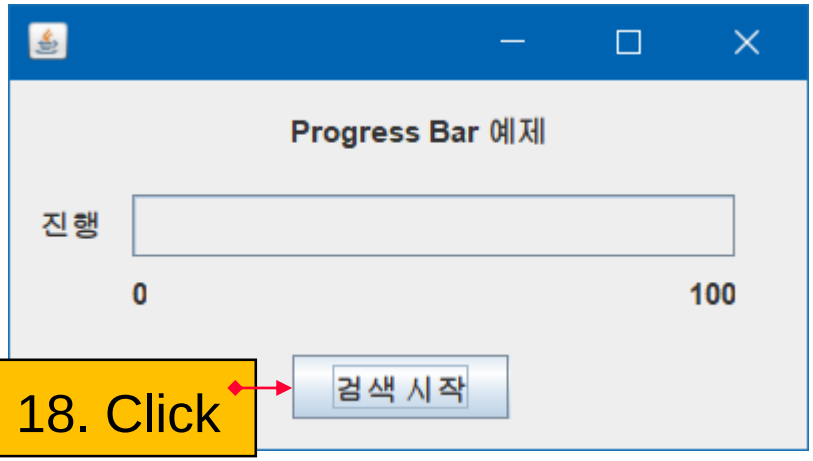


17. Click

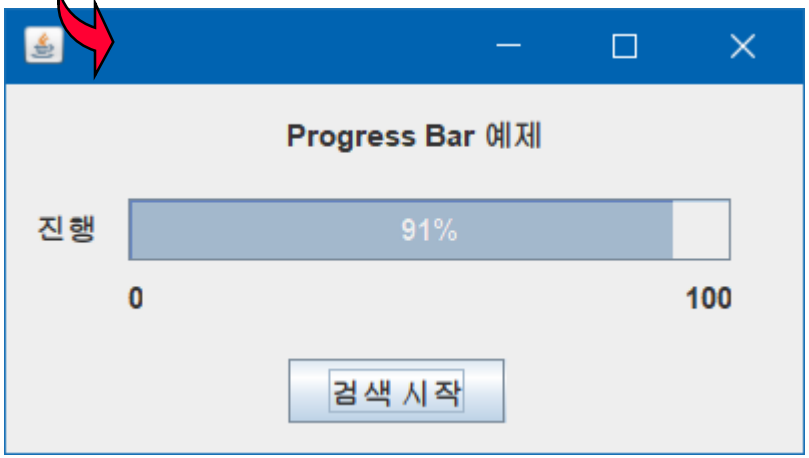
Practice 2 : Progress Bar Control (10)

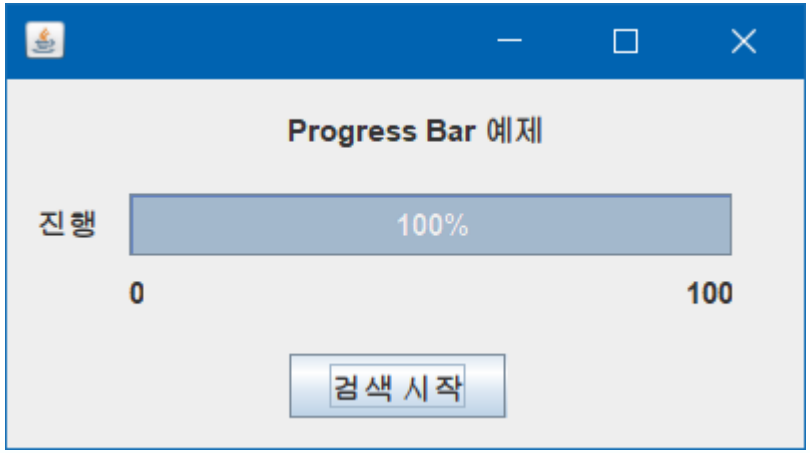
Run

❖ [검색 시작] button Click



18. Click





Slider Control

Slider Control

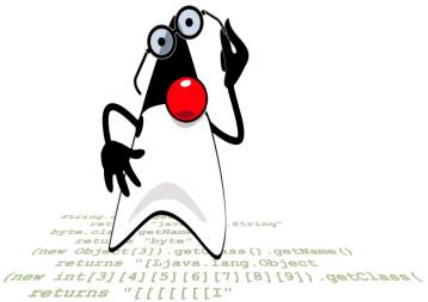
❖ Function

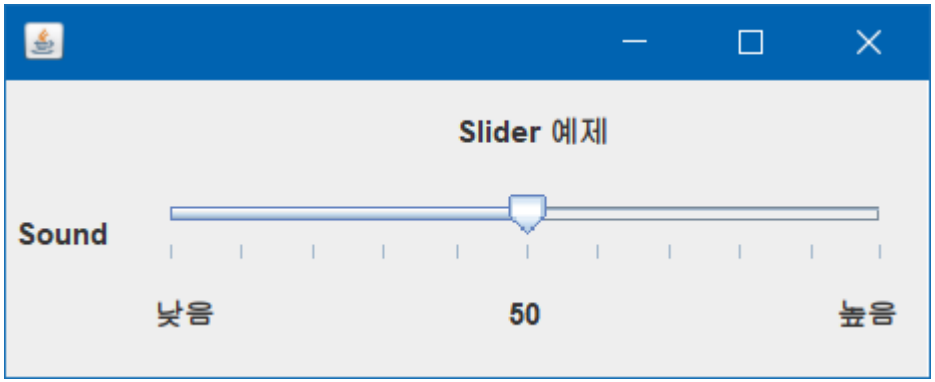
◆ 값 조정 및 진행 상태 표현

Practice 3 : Slider Control

❖ Project Name: Slider_Source(Time: 20 min)

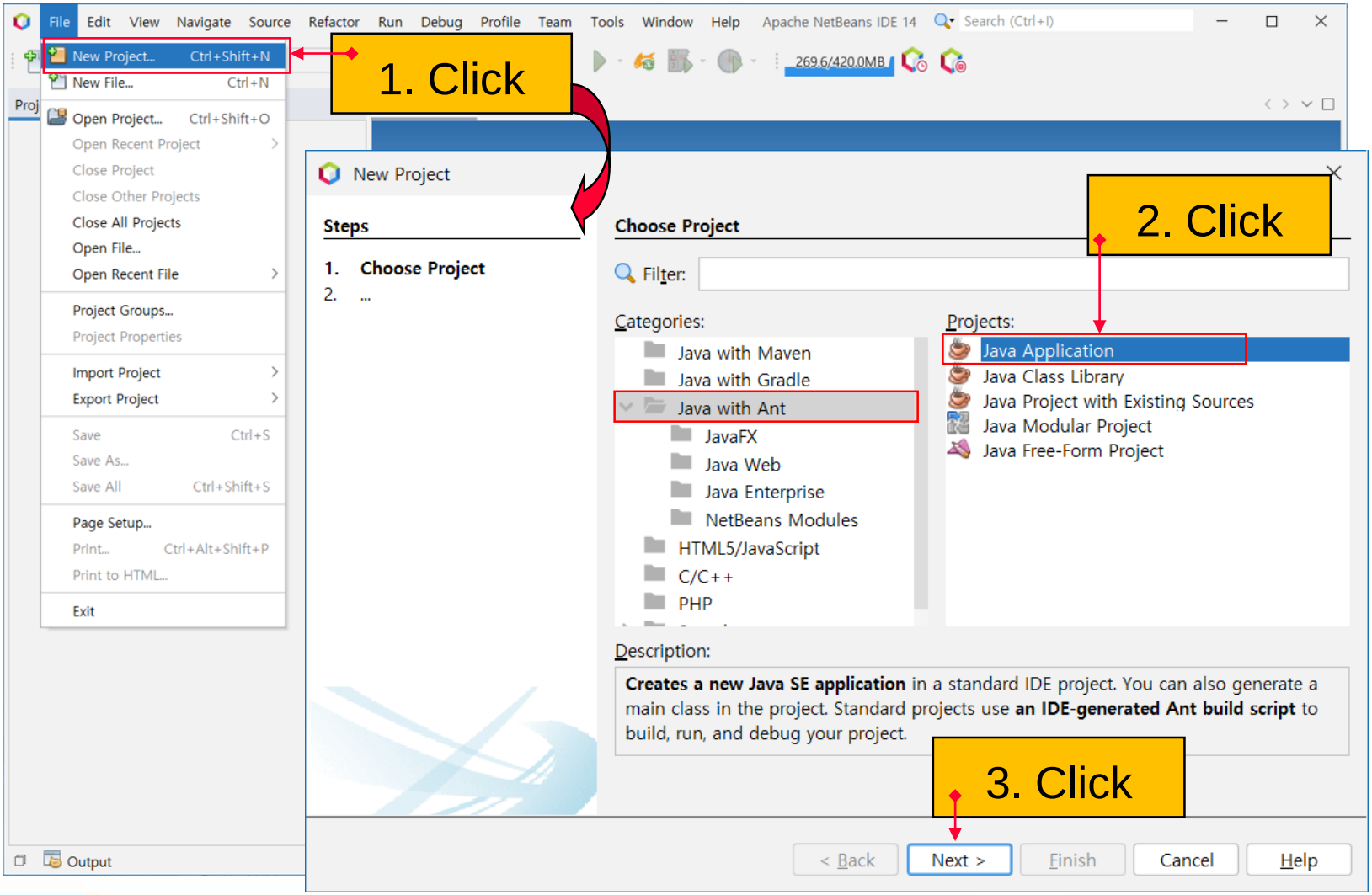
- Slider 조정
  - Slider를 조정한 위치의 값을 Label Control에 표현하시오.





Practice 3 : Slider Control (1)

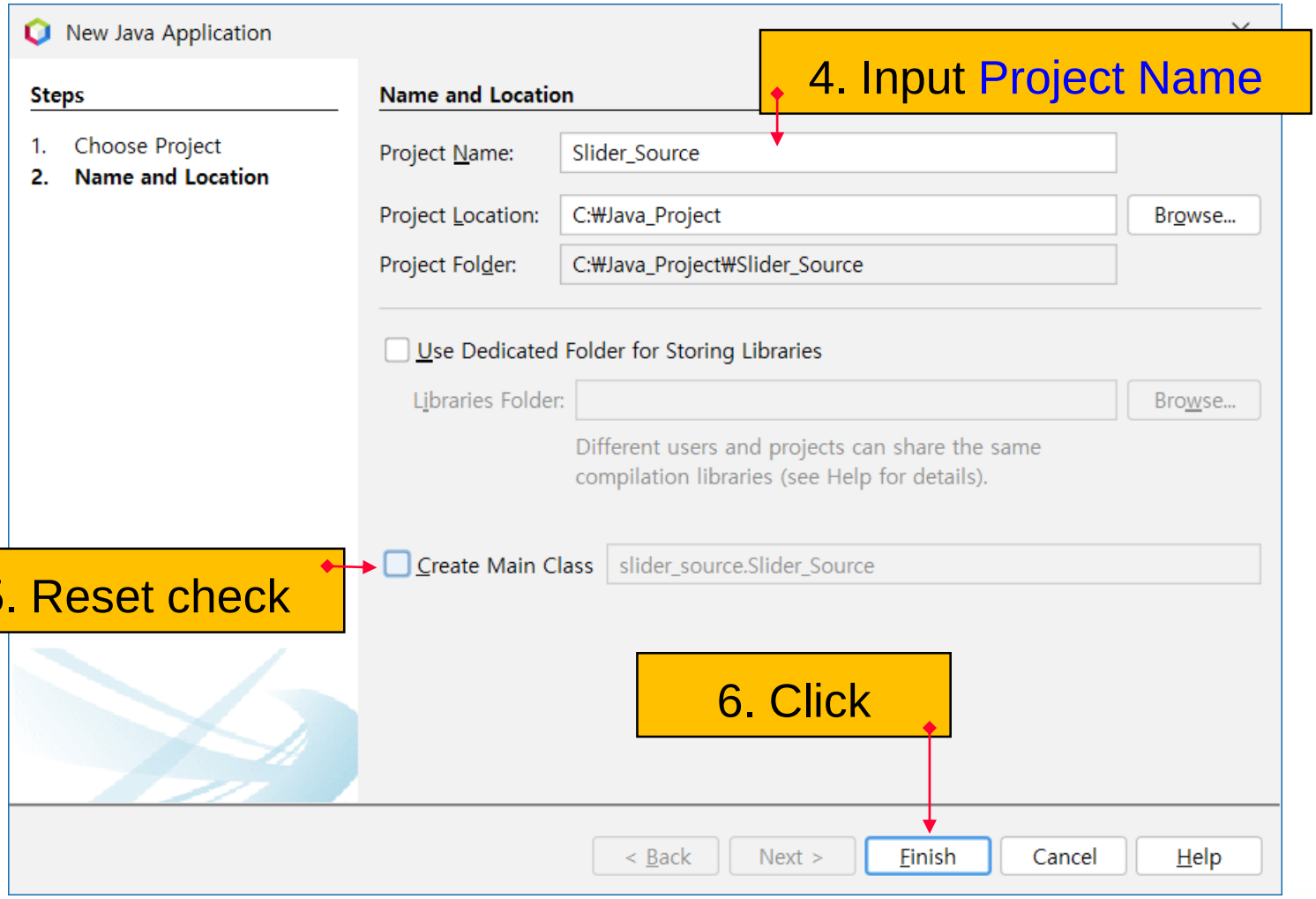
Create Project



Practice 3 : Slider Control (2)

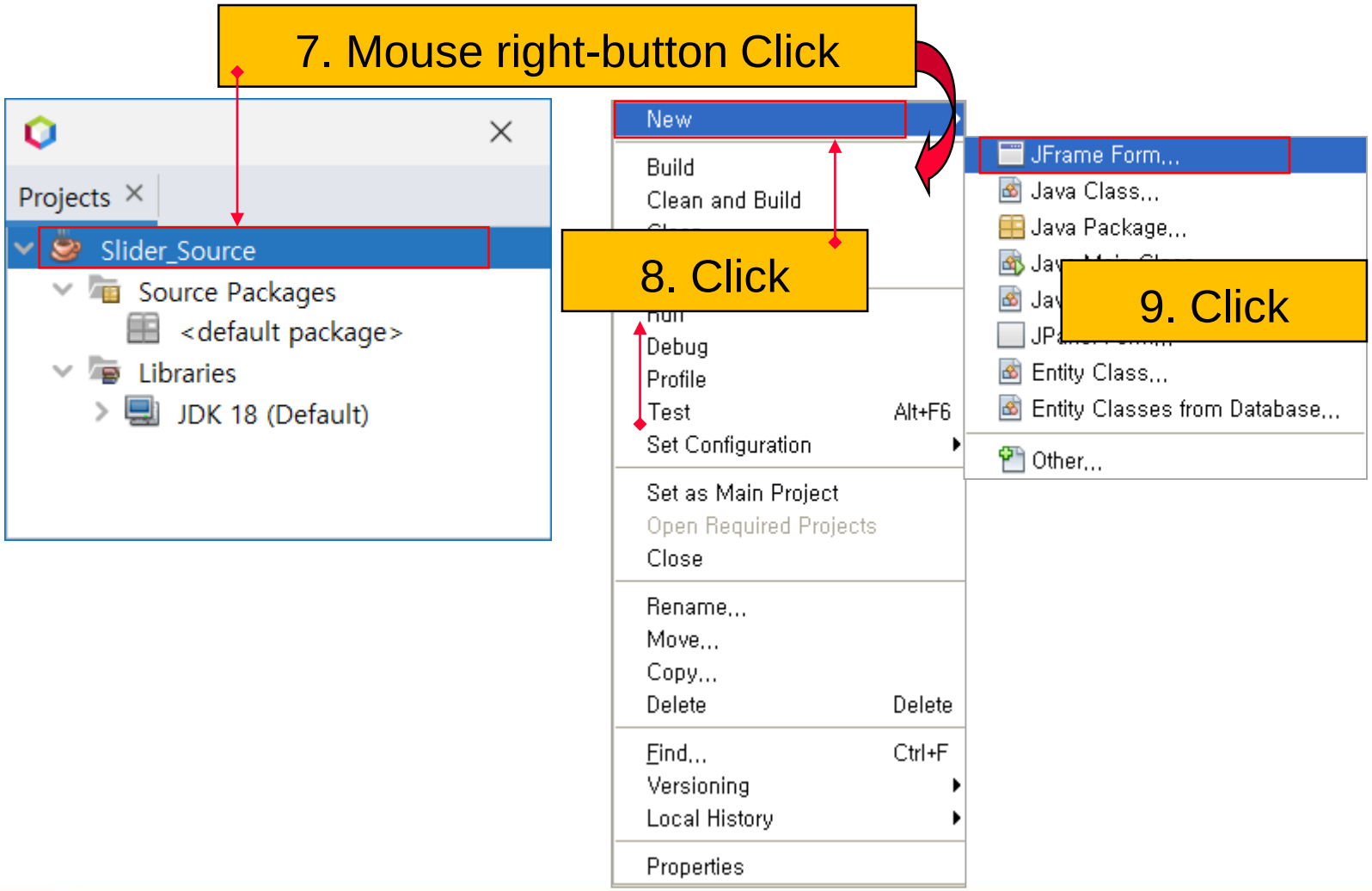
Project Name and Location

- Project name: Slider_Source



Practice 3 : Slider Control (3)

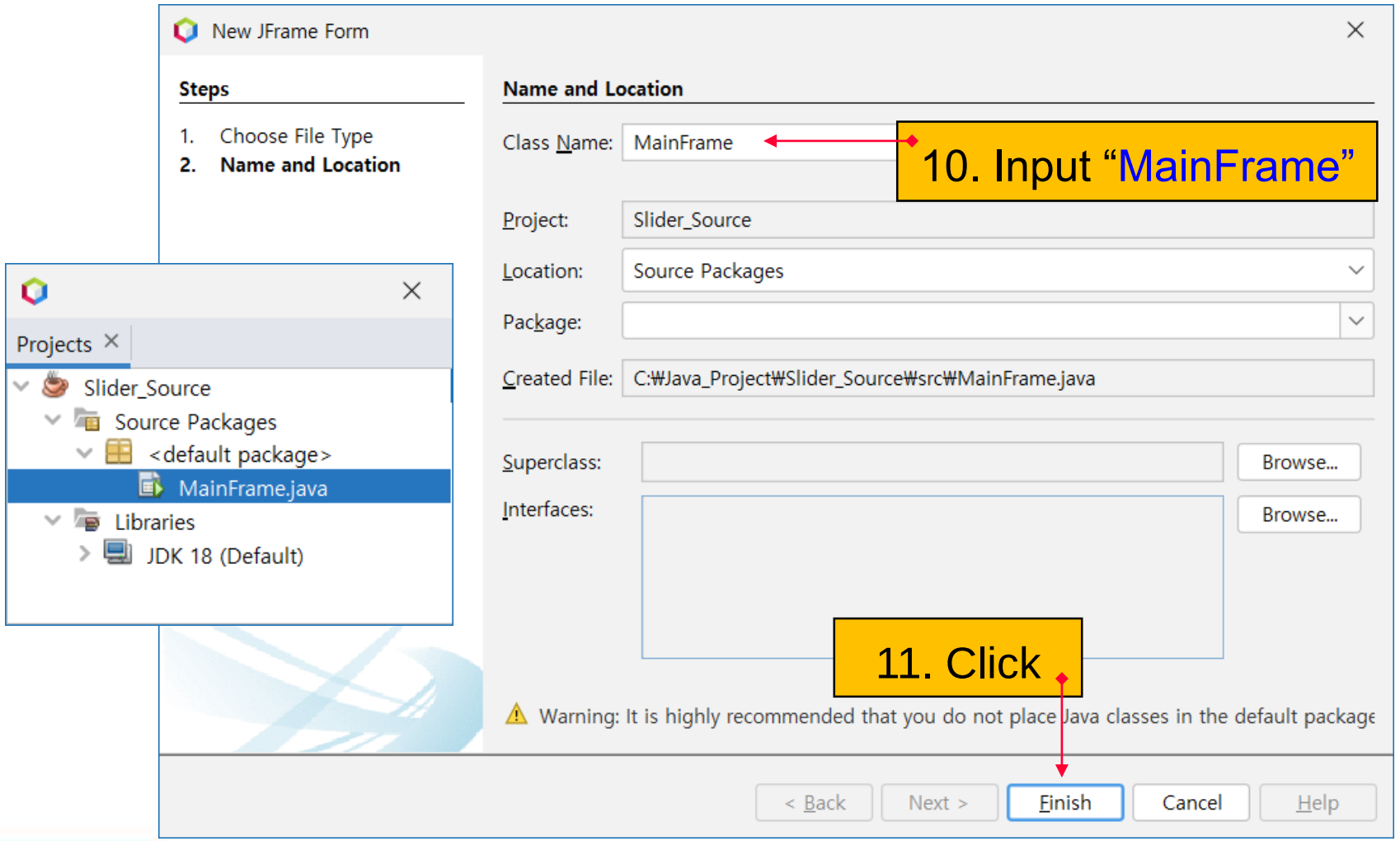
- Create JFrame Form



Practice 3 : Slider Control (4)

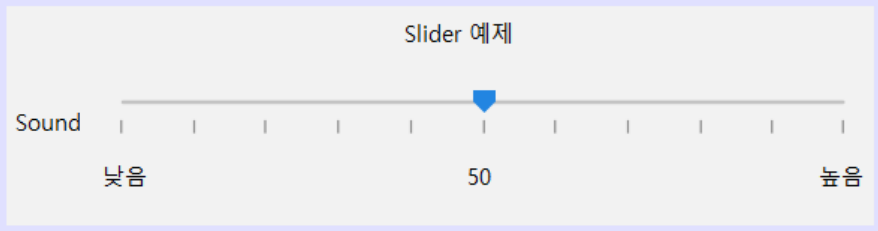
Setting JFrame Form Name

❖ Create MainFrame.java



Practice 3 : Slider Control (5)

Control Layout & Property Setting



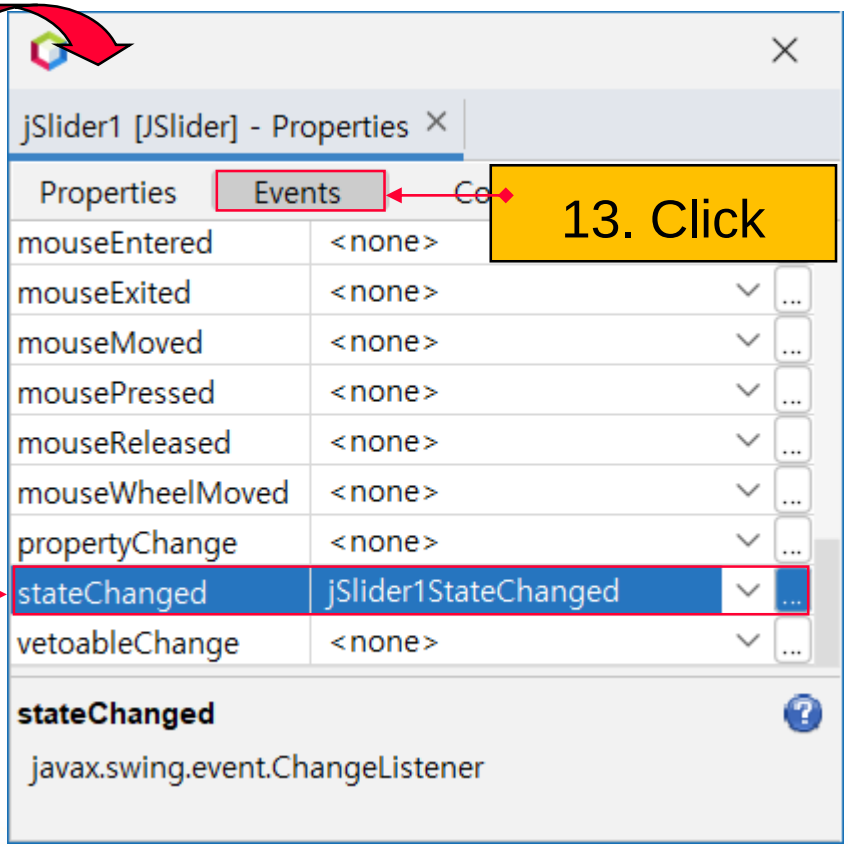
Control	Properties Setting
jLabel1	• Variable Name : lblTitle • Text : Slider 예제
jLabel2	• Variable Name : lblSound • Text : Sound
jLabel3	• Variable Name : lblLow • Text : 낮음
jLabel4	• Variable Name : lblHigh • Text : 높음
jLabel5	• Variable Name : lblValue • Text : 50

Control	Properties Setting
jSlider1	• majorTickSpacing : 10 • paintTicks : ☑ • paintTrack : ☑

Practice 3 : Slider Control (6)

Setting StateChanged() Event Handler

12. Click



13. Click

14. Setting

Practice 3 : Slider Control (7)

❖ jSlider1StateChanged() Event Handler

MainFrame.java - Editor

MainFrame.java

Source | Design | History

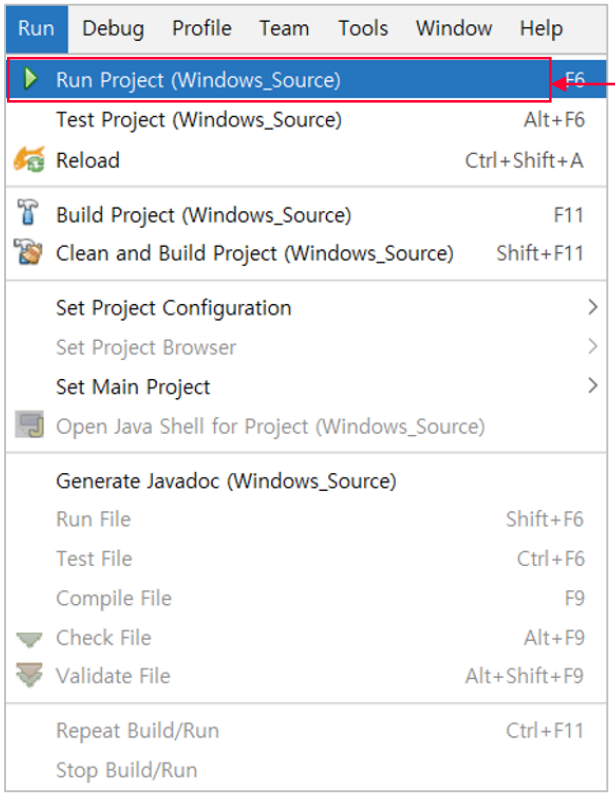
```
90
    private void jSlider1StateChanged(javax.swing.event.ChangeEvent evt) {
        String strData = null;
        int iValue = 0;

94
95      iValue = jSlider1.getValue();           //Return jSlider1 Value
96      strData = Integer.toString( iValue);    //Integer -> String
97      lblValue.setText(strData);
98  }
99
100 /**
101  * @param args the command line arguments
102  */
103 public static void main(String args[]) {
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
106             new MainFrame().setVisible(true);
107         }
108     });
109 }
```

15. Coding

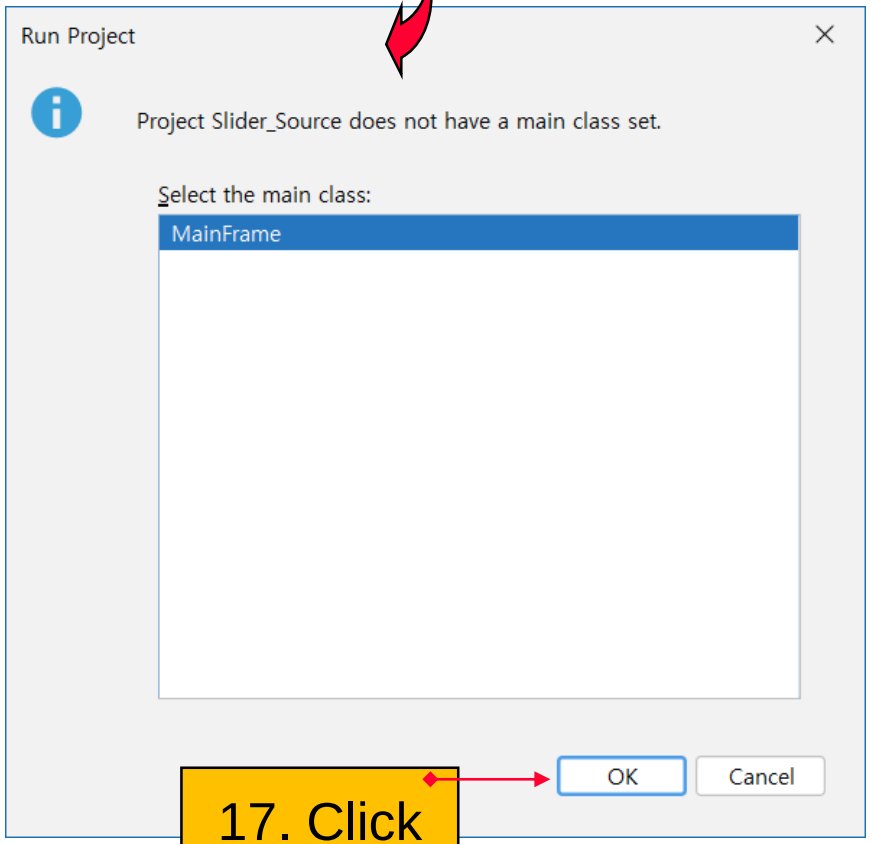
1:1 INS

Practice 3 : Slider Control (8)

Run



16. Click

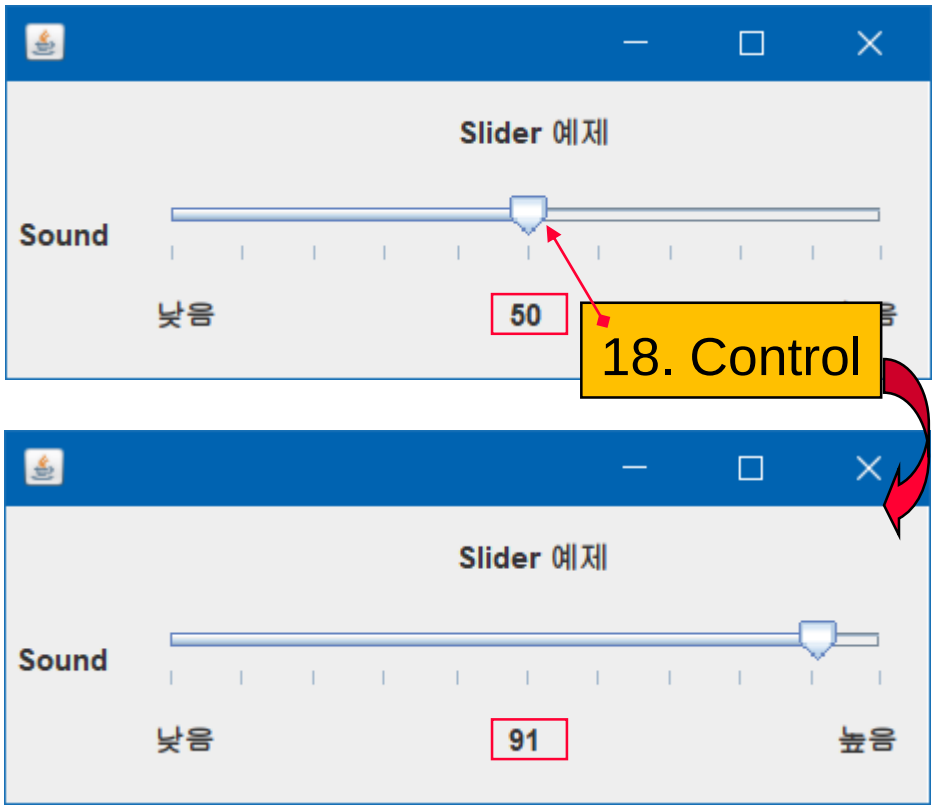


17. Click

Practice 3 : Slider Control (9)

Run

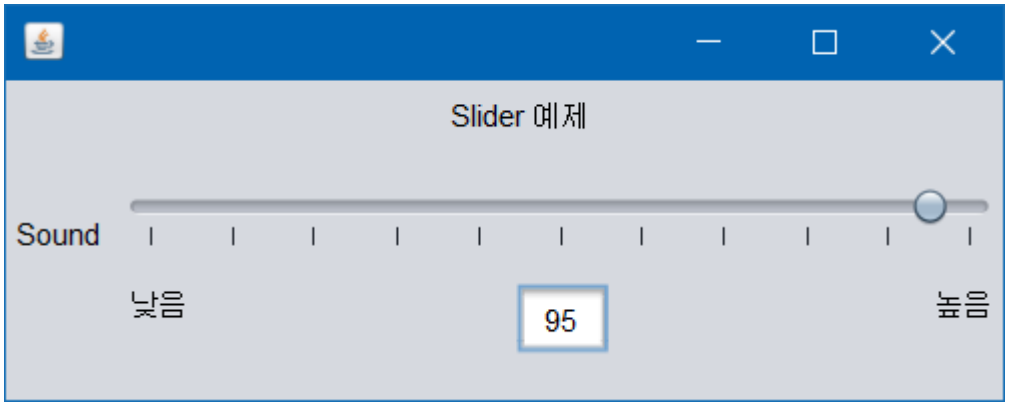
- Slider 조정

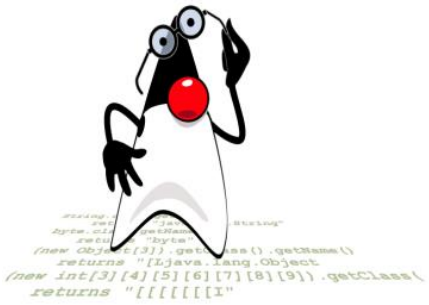


Practice 4 : Slider Control

❖ Project Name: Slider2_Source(Time: 20 min)

- Input Slider Value
  - Slider를 조정한 위치의 값을 txtField Control에 출력하도록 구현하시오.
  - Slider의 입력값에 따라 Slider를 조정하도록 구현하시오.





Practice 4 : Slider Control (1)

- jSlider1StateChanged() Event Handler

SliderFrame.java - Editor

SliderFrame.java

Source | Design | History

```
1
2   import java.awt.event.KeyEvent;
3
4   public class SliderFrame extends javax.swing.JFrame {
5
6       /** Creates new form SliderFrame ...3 lines */
9       public SliderFrame() {...3 lines }
12
13      /** This method is called from within the constructor to initialize the form ...4 lines */
17      @SuppressWarnings("unchecked")
18      Generated Code
99
        private void jSlider1StateChanged(javax.swing.event.ChangeEvent evt) {
            String strData = null;
            int iValue = 0;
103
104         iValue = jSlider1.getValue();           //Return jSlider1 Value
105         strData = Integer.toString( iValue);    //Integer -> String
106             ❶
107     }
```

111:4 INS

Practice 4 : Slider Control (2)

❖ KeyTyped() Event Handler

15. Coding

SliderFrame.java - Editor

SliderFrame.java

Source | Design | History

```
1
2   import java.awt.event.KeyEvent;
3
4   public class SliderFrame extends javax.swing.JFrame {
5
6       /** Creates new form SliderFrame ...3 lines */
9       public SliderFrame() {...3 lines }
12
13      /** This method is called from within the constructor to initialize the form ...4 lines */
17      @SuppressWarnings("unchecked")
18      Generated Code
99
        private void jSlider1StateChanged(javax.swing.event.ChangeEvent evt) {...8 lines }
108
109     private void txtFD_ValueKeyTyped(java.awt.event.KeyEvent evt) {
            String strData = null;
            int iValue = 0;
112
113         if(                    ❷                    ){
114             strData = txtFD_Value.getText();
115                    ❸                          //String -> Integer
116                                               //Setting jSlider1 Value
117         }
118     }
```

145:25 INS

학습 요약

- TabbedPane Control
- Progress Bar Control
- Slider Control