객체지향프로그래밍응용 과제09 컴퓨터소프트웨어공학과 2-YA 20202296 전채린

[문제01]

사용자로부터 입력받은 이메일과 비밀번호를 콘솔에 출력하시오. (로그인버튼 클릭)

[소스코드]

// 01. 다음과 같은 프로그램을 작성하시오.

// 사용자로부터 입력받은 이메일과 비밀번호를 콘솔에 출력하시오. (로그인버튼 클릭)

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class Practice01 extends JFrame {

public Practice01() {

setTitle("LOGIN FORM");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

Container c = getContentPane();

c.setLayout(new GridLayout(3, 2));

c.add(new JLabel("이메일"));

JTextField tfEmail = new JTextField();

c.add(tfEmail);

c.add(new JLabel("비밀번호"));

JTextField tfPwd = new JTextField();

c.add(tfPwd);

JButton btnLogin = new JButton("로그인");

JButton btnCancel = new JButton("취소");

c.add(btnLogin);

c.add(btnCancel);

btnLogin.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

String inEmail = tfEmail.getText();

String inPwd = tfPwd.getText();

System.out.println("이메일: " + inEmail + ", 비밀번호: " + inPwd);

}

});

btnCancel.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

tfEmail.setText("");

tfPwd.setText("");

}

});

setSize(300, 150);

setVisible(true);

}

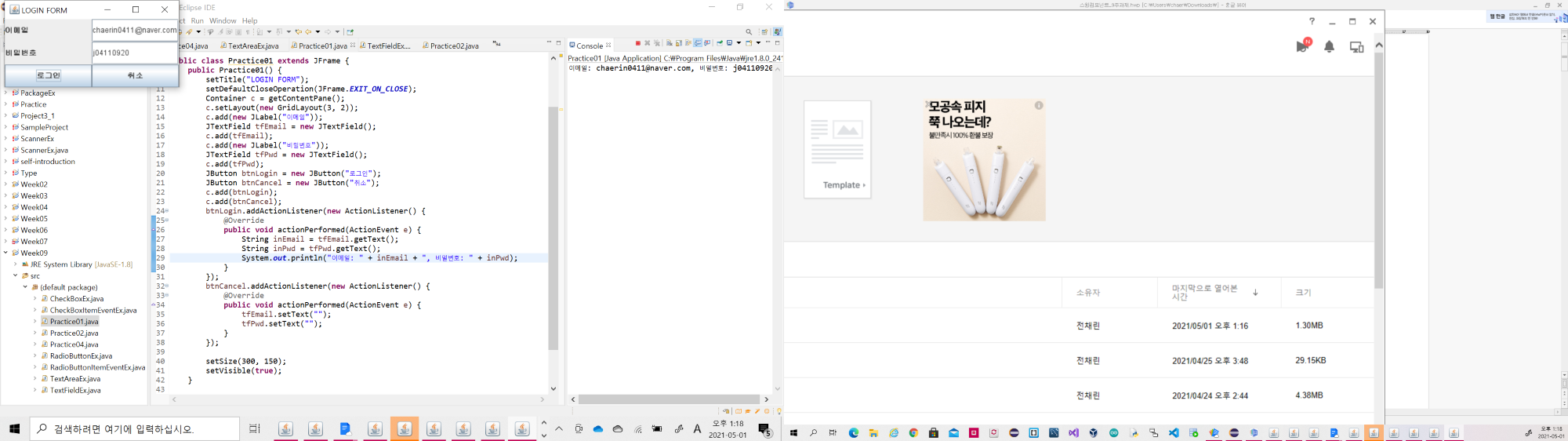
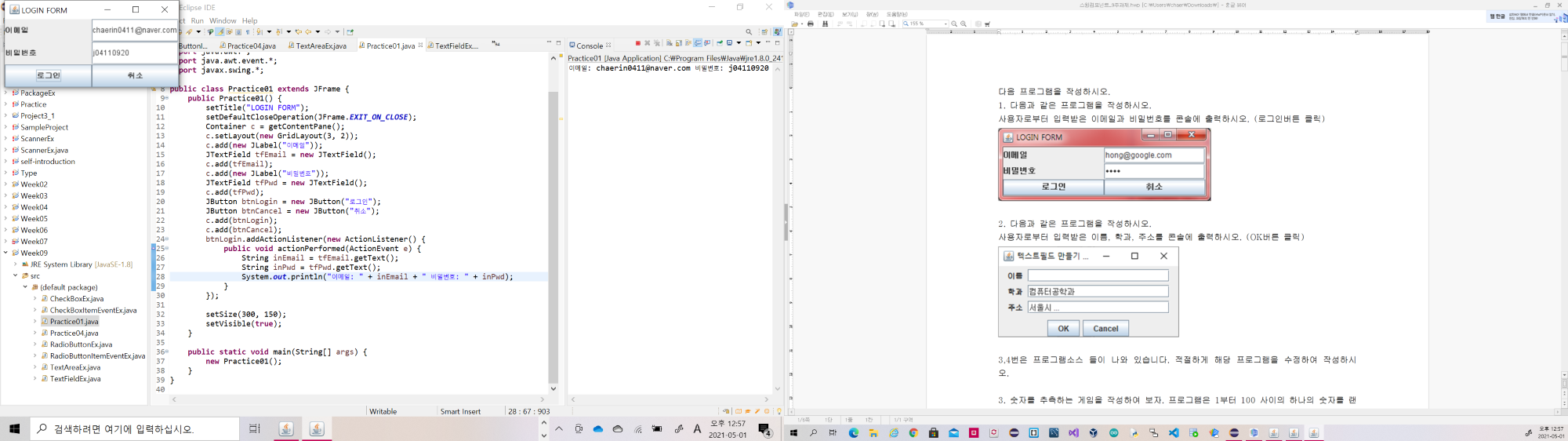
public static void main(String[] args) {

new Practice01();

}

}

[실행결과]



[문제02]

사용자로부터 입력받은 이름, 학과, 주소를 콘솔에 출력하시오. (OK버튼 클릭)

[소스코드]

// 02. 다음과 같은 프로그램을 작성하시오.

// 사용자로부터 입력받은 이름, 학과, 주소를 콘솔에 출력하시오. (OK버튼 클릭)

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class Practice02 extends JFrame {

public Practice02() {

setTitle("텍스트필드 만들기 예제");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

Container c = getContentPane();

c.setLayout(new FlowLayout());

c.add(new JLabel("이름 "));

JTextField tfName = new JTextField(20); // 창의 열 개수 20

c.add(tfName);

c.add(new JLabel("학과 "));

JTextField tfDept = new JTextField("컴퓨터소프트웨어공학과", 20); // 창의 열 개수 20

c.add(tfDept);

c.add(new JLabel("주소 "));

JTextField tfAddr = new JTextField("서울시 ...", 20); // 창의 열 개수 20

c.add(tfAddr);

JPanel pnBtn = new JPanel();

JButton btnOk = new JButton("OK");

JButton btnCancel = new JButton("Cancel");

btnOk.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

String inName = tfName.getText();

String inDept = tfDept.getText();

String inAddr = tfAddr.getText();

System.out.println("이름: " + inName + ", 학과: " + inDept + ", 주소: " + inAddr);

}

});

btnCancel.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

tfName.setText(""); tfDept.setText(""); tfAddr.setText("");

}

});

pnBtn.add(btnOk);

pnBtn.add(btnCancel);

c.add(pnBtn);

setSize(300, 160);

setVisible(true);

}

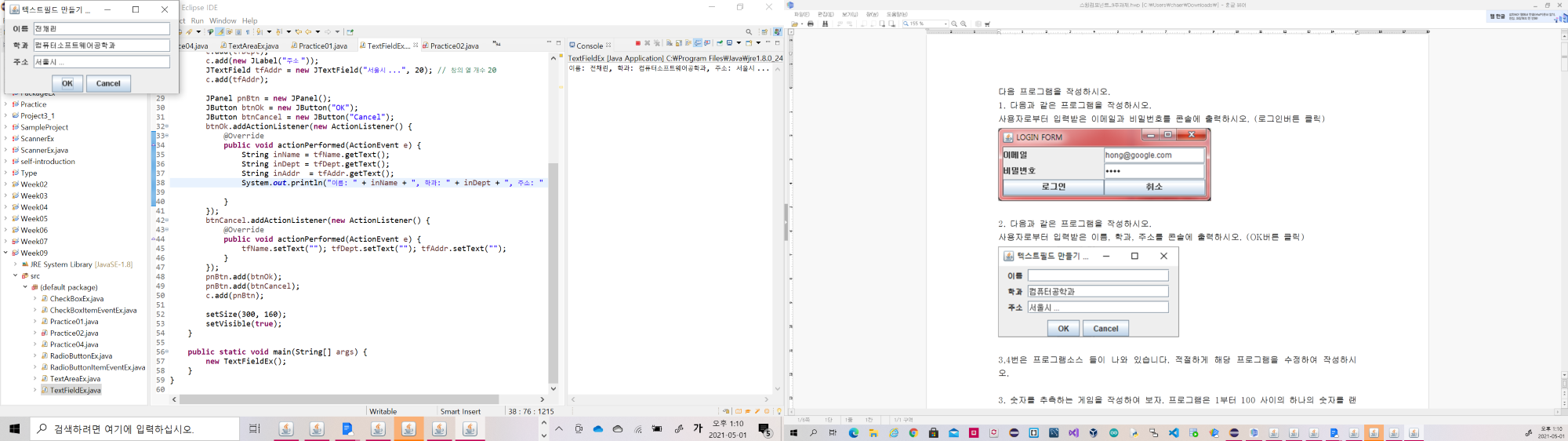
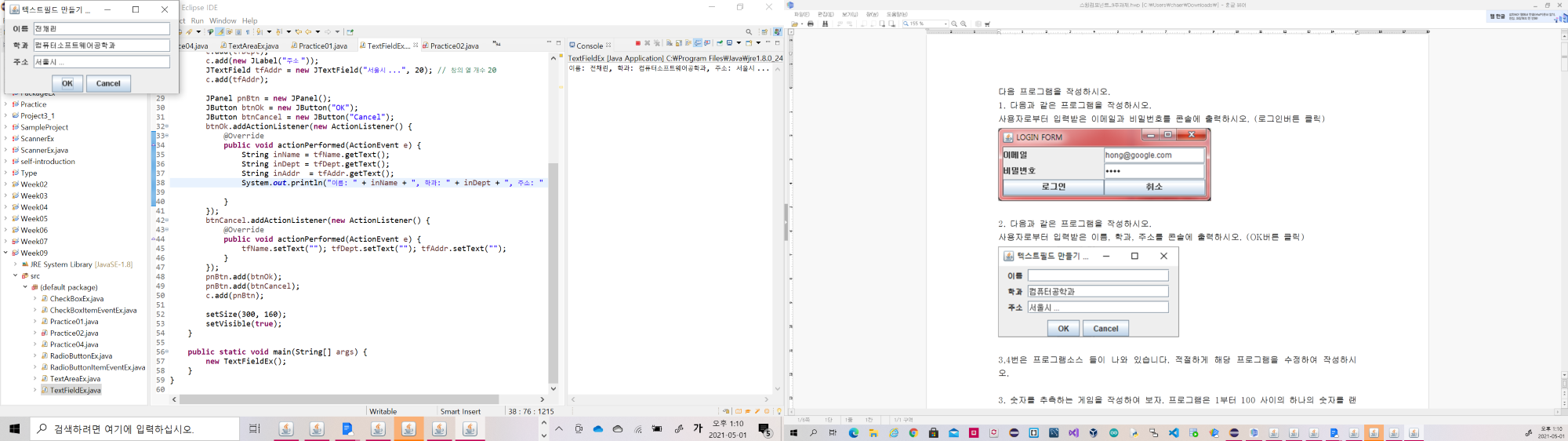
public static void main(String[] args) {

new Practice02();

}

}

[실행결과]



[문제03]

숫자를 추축하는 게임을 작성하여 보자. 프로그램은 1부터 100 사이의 하나의 숫자를 랜덤하게 생성하여 가지고 있다. 사용자는 숫자를 추측하고 컴퓨터는 사용자가 입력한 숫자가 정답보다 높은지 낮은지만을 알려준다. 다음과 같은 사용자 인터페이스를 작성한다. 숫자가 정답이 아니면, 힌트를 나타내는 텍스트 필드의 배경색을 빨간색으로 변경한다.

[소스코드]

// 03. 숫자를 추축하는 게임을 작성하여 보자.

// 프로그램은 1부터 100 사이의 하나의 숫자를 랜덤하게 생성하여 가지고 있다.

// 사용자는 숫자를 추측하고 컴퓨터는 사용자가 입력한 숫자가 정답보다 높은지 낮은지만을 알려준다.

// 다음과 같은 사용자 인터페이스를 작성한다.

// 숫자가 정답이 아니면, 힌트를 나타내는 텍스트 필드의 배경색을 빨간색으로 변경한다.

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class Practice03 extends JFrame {

private int randomNum;

private JTextField tfInput;

private JLabel lbGuide;

private JButton btnRetry;

private JButton btnExit;

public Practice03() {

// 부모 클래스 생성자 명시적 호출

super("숫자게임");

// top 패널에 들어갈 컴포넌트 구성

JLabel lbInfo = new JLabel("숫자를 추측하시오 : ");

tfInput = new JTextField(10);

tfInput.addActionListener(new ActionListener() { // TextField에 들어갈 액션 리스너

@Override

public void actionPerformed(ActionEvent e) {

try {

if(Integer.parseInt(tfInput.getText()) < randomNum) {

lbGuide.setText("너무 낮습니다!");

lbGuide.setBackground(Color.RED);

}

else if(Integer.parseInt(tfInput.getText()) > randomNum) {

lbGuide.setText("너무 높습니다!");

lbGuide.setBackground(Color.RED);

}

else {

lbGuide.setText("정답입니다!");

lbGuide.setBackground(Color.WHITE);

}

} catch(NumberFormatException ne) {

lbGuide.setText("잘못된 입력입니다!");

lbGuide.setBackground(Color.RED);

}

tfInput.selectAll();

}

});

// 텍스트필드에 액션이벤트 처리

// top 패널 구성

JPanel pnTop = new JPanel();

pnTop.add(lbInfo);

pnTop.add(tfInput);

// middle 패널에 들어갈 레이블 구성

lbGuide = new JLabel("숫자를 입력하시오.");

lbGuide.setBackground(Color.WHITE);

lbGuide.setOpaque(true); // 레이블을 불투명하게

// middle 패널 구성

JPanel pnMid = new JPanel();

pnMid.add(lbGuide);

// low 패널에 들어갈 버튼 구성

btnRetry = new JButton("새 게임");

btnExit = new JButton("종료");

btnRetry.addActionListener(new ActionListener() { //Button에 들어갈 액션 리스너

@Override

public void actionPerformed(ActionEvent e) {

setRandomNum();

lbGuide.setText("숫자를 입력하시오.");

tfInput.setText("");

}

});

btnExit.addActionListener(new ActionListener() { // Button에 들어갈 액션 리스너

@Override

public void actionPerformed(ActionEvent e) {

System.exit(0);

}

});

// low 패널 구성

JPanel pnLow = new JPanel();

pnLow.add(btnRetry);

pnLow.add(btnExit);

// 메인 패널 구성

JPanel pnMain = new JPanel();

pnMain.setLayout(new BoxLayout(pnMain, BoxLayout.Y\_AXIS));

pnMain.add(pnTop);

pnMain.add(pnMid);

pnMain.add(pnLow);

add(pnMain);

// 랜덤 넘버와 프레임 설정

setRandomNum();

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

pack();

setSize(300, 150);

setVisible(true);

}

// 접근자

public int getRandomNum() { return randomNum; }

// 설정자

public void setRandomNum() { randomNum = (int)(Math.random()\*100)+1; }

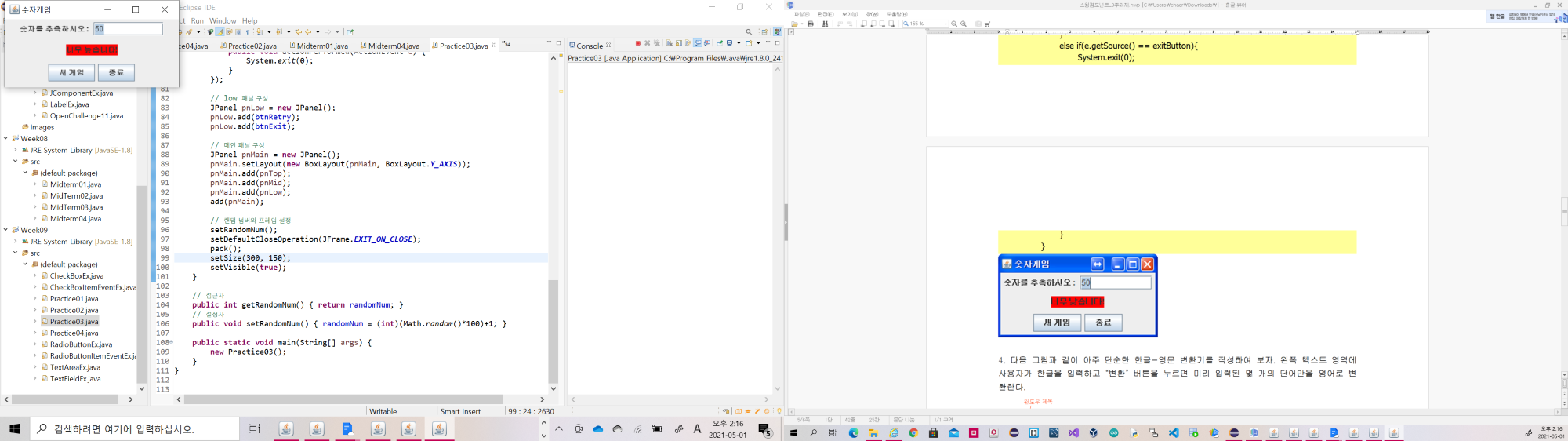
public static void main(String[] args) {

new Practice03();

}

}

[실행결과]



[문제04]

다음 그림과 같이 아주 단순한 한글-영문 변환기를 작성하여 보자. 왼쪽 텍스트 영역에 사용자가 한글을 입력하고 "변환" 버튼을 누르면 미리 입력된 몇 개의 단어만을 영어로 변환한다. 일단 텍스트 필드에서 getText()를 이용하여서 텍스트를 얻은 후에, 텍스트에서 단어를 분리하려면 Scanner 클래스의 next() 메소드를 사용하라.

[소스코드]

// 04. 다음 그림과 같이 아주 단순한 한글-영문 변환기를 작성하여 보자.

// 왼쪽 텍스트 영역에 사용자가 한글을 입력하고 "변환" 버튼을 누르면 미리 입력된 몇 개의 단어만을 영어로 변환한다.

// 일단 텍스트 필드에서 getText()를 이용하여서 텍스트를 얻은 후에,

// 텍스트에서 단어를 분리하려면 Scanner 클래스의 next() 메소드를 사용하라.

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class Practice04 extends JFrame {

JButton btnConvert;

JButton btnCancel;

JTextArea taIn;

JTextArea taOut;

public Practice04() {

super("텍스트 변환");

// 텍스트 에어리어

taIn = new JTextArea("텍스트가 입력됩니다. 몇 개의 단어를 영어로 변환합니다.", 10, 14);

taOut = new JTextArea(10, 14);

taIn.setLineWrap(true); // 자동 줄바꿈

taOut.setLineWrap(true);

taOut.setEnabled(false); // 비활성화

// 텍스트 에어리어를 관리할 패널

JPanel pnTextArea = new JPanel(new GridLayout(1, 2, 20, 20));

pnTextArea.add(taIn);

pnTextArea.add(taOut);

// 버튼

btnConvert = new JButton("변환");

btnCancel = new JButton("취소");

// 버튼 이번트처리

btnConvert.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

taOut.setText("");

String result = toEnglish(taIn.getText());

taOut.append(result);

}

});

btnCancel.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

taIn.setText("");

taOut.setText("");

}

});

// 버튼 패널

JPanel pnButton = new JPanel(new FlowLayout());

pnButton.add(btnConvert);

pnButton.add(btnCancel);

// 메인 패널

JPanel pnMain = new JPanel();

pnMain.setLayout(new BorderLayout());

pnMain.add(BorderLayout.CENTER, pnTextArea);

pnMain.add(BorderLayout.SOUTH, pnButton);

// 프레임 설정

setLayout(new FlowLayout(FlowLayout.CENTER, 20, 20));

add(pnMain);

pack();

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setVisible(true);

}

// 영어를 한국어로 변환하는 메소드

private String toEnglish(String korean) {

String result = korean;

result = result.replace("텍스트", "Text");

result = result.replace("영어", "English");

return result;

}

public static void main(String[] args) {

new Practice04();

}

}

[실행결과]

