

2020-1

# 객체프로그래밍 실습

송인서

공과대학 IT미디어공학과 4학년(17)

songinseo0910@duksung.ac.kr

010-9610-9779

QnA 및 공지용  slack

2020-1-dswu-it-java.slack.com



# 남은 주차 진행 계획

9장 인터페이스

10장 중첩 클래스

11장 예외 처리

12장 제네릭

13장 스트림 기반의 입출력과 통신

14장 스레드

일	월	화	수	목	금	토
	1	2	3	4	5	6
7	8	9	10	11	12	13
14	15	16	17	18	19	20
21	22	23 6~7 주차 과제 마감	24	25	26 기말 과제 마감	27
28	29	30				

# 인터페이스(Interface)

- class에 어떤 '기능' 혹은 '속성'을 부여하기 위한 규약
- 추상 메소드의 집합으로, 인터페이스 자체는 인스턴스화 해서 사용할 수 없음
- 인터페이스를 구현(implement)하는 클래스에서는 반드시 인터페이스에서 정의된 모든 추상 메소드의 바디를 작성해야 함
- 다음과 같은 구문으로 사용

```
class 클래스명 implements 인터페이스1, 인터페이스2 ... {  
    // class body  
}
```

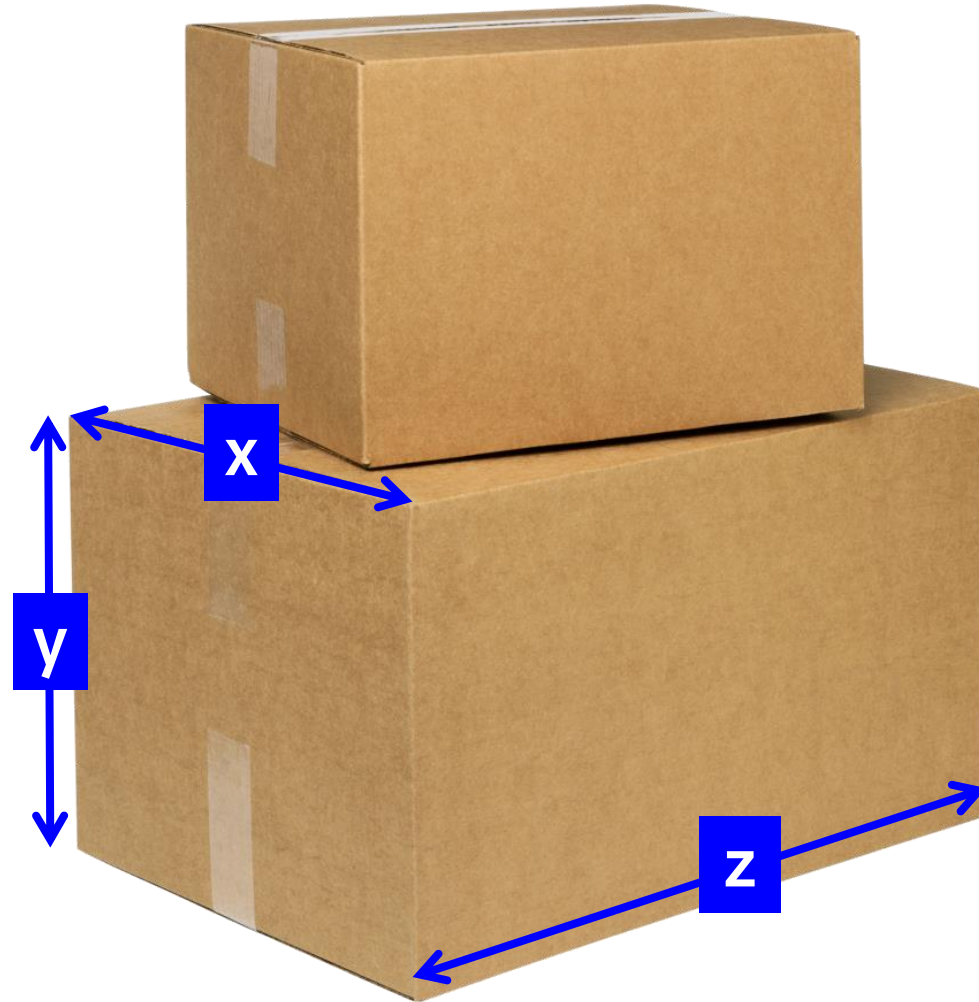
# 상속과의 비교

상속(Inheritance)	인터페이스(Interface)
<ul style="list-style-type: none"><li>• 다중 상속 불가능</li><li>• 부모 클래스의 메소드를 자식 클래스에서 반드시 재작성(overriding)하지 않고 사용해도 정상적으로 동작함</li></ul>	<ul style="list-style-type: none"><li>• 다중 구현 가능</li><li>• 상수와 추상 메소드만 멤버로 가질 수 있음 (default method라는 특수한 경우가 있긴 함)</li><li>• 사용하려는 인터페이스 내에 선언된 모든 추상 메소드를 클래스 내에서 반드시 작성해야 함</li><li>• 모든 멤버의 접근지정자를 public으로 간주함</li></ul>

# 제네릭(Generic)

- class(또는 type)를 매개 변수처럼 사용할 수 있게 하는 기능
  - 컴파일러가 타입을 체크할 수 있게 해 줌
  - 명시적 타입 변환 해 줄 필요를 줄여줌
  - 하나의 코드를 다양한 클래스에 대해 동작할 수 있게 함

## 예제 - Comparable<T>구현해보기




# interface Comparable<T>

## Method Summary



All Methods	Instance Methods	Abstract Methods
Modifier and Type	Method	Description
int	<code>compareTo(T o)</code>	Compares this object with the specified object for order.

# 예제 - Comparable구현해보기

```
class Box implements Comparable<Box>{
```

 The type Box must implement the inherited abstract method Comparable<Box>.compareTo(Box)

2 quick fixes available:

-  [Add unimplemented methods](#)
-  [Make type 'Box' abstract](#)

Press 'F2' for focus

```
    private int x;  
    private int y;  
    private int z;
```

```
}
```

## Method Summary

All Methods	Instance Methods	Abstract Methods
Modifier and Type	Method	Description
int	<code>compareTo(T o)</code>	Compares this object with the specified object for order.



## 예제 - Comparable구현해보기

```
class Box implements Comparable<Box>{
```

```
    private int x;
```

```
    private int y;
```

```
    private int z;
```

```
    @Override
```

```
    public int compareTo(Box arg0) {
```

```
        // TODO Auto-generated method stub
```

```
        return 0;
```

```
    }
```

```
}
```

## 예제 - Comparable구현해보기

```
class Box implements Comparable<Box>{  
  
    private int x; private int y; private int z;  
  
    public int compareTo(Box other) {  
  
        int thisV    = this.x * this.y * this.z;  
        int otherV   = other.x * other.y * other.z;  
  
        if          (thisV >  otherV)    return  1;  
        else if    (thisV == otherV)    return  0;  
        else if    (thisV <  otherV)    return -1;  
  
        return 0;  
    }  
}
```

# 참고 - context menu : Source Alt + Shift + S

```
class Box implements Comparable<Box>{
```

The context menu is open, displaying the following options:

- Toggle Comment (Ctrl+)
- Remove Block Comment (Ctrl+Shift+W)
- Generate Element Comment (Alt+Shift+J)
- Correct Indentation (Ctrl+I)
- Format (Ctrl+Shift+F)
- Format Element
- Add Import (Ctrl+Shift+M)
- Organize Imports (Ctrl+Shift+O)
- Sort Members...
- Clean Up...
- Override/Implement Methods...
- Generate Getters and Setters...
- Generate Delegate Methods...
- Generate hashCode() and equals()...
- Generate toString()...
- Generate Constructor using Fields...
- Generate Constructors from Superclass...
- Externalize Strings...

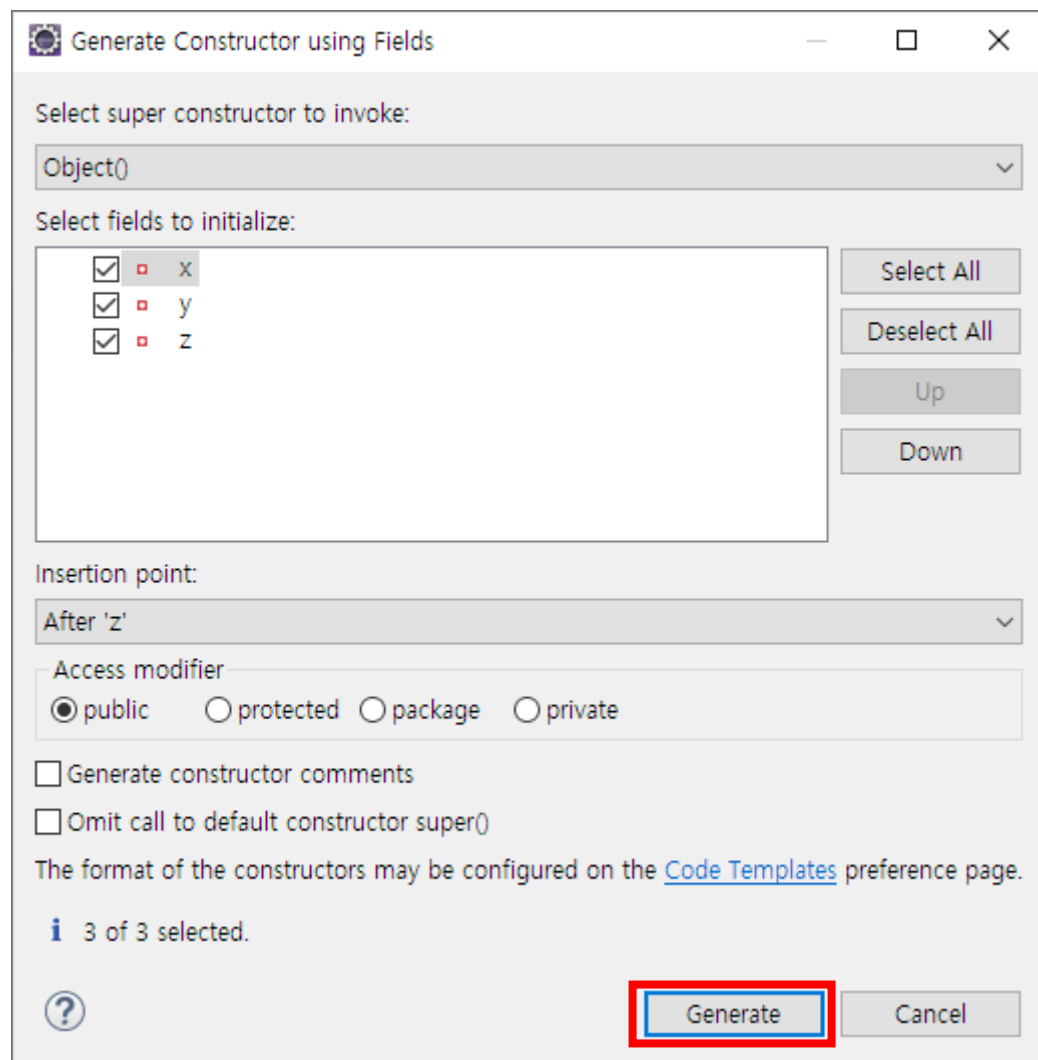
The background code snippet is as follows:

```
private int y; private int z;

@Override {
    * this.y * this.z;
    * other.y * other.z;

@Override return 1;
@Override return 0;
@Override return -1;
}
```

# 참고 – context menu Source



자동 생성 !

```
public Box(int x, int y, int z) {  
    super();  
    this.x = x;  
    this.y = y;  
    this.z = z;  
}
```

## 예제 - Comparable구현해보기

```
public class Week6 {  
  
    public static void main(String[] args) {  
  
        ArrayList<Box> boxList = new ArrayList<Box>();  
  
        boxList.add(new Box(1,1,1));  
        boxList.add(new Box(3,3,3));  
        boxList.add(new Box(2,2,2));  
  
        Collections.sort(boxList);  
  
        System.out.println(boxList);  
  
    }  
}
```

# javax.swing

- 쉽게 GUI 프로그램을 작성할 수 있게 해주는 라이브러리
- 버튼, 텍스트 필드 등 다양한 컴포넌트 존재

# 예제 - GUI 프로그래밍 해 보기

```
import javax.swing.*;

class Window extends JFrame implements ActionListener{

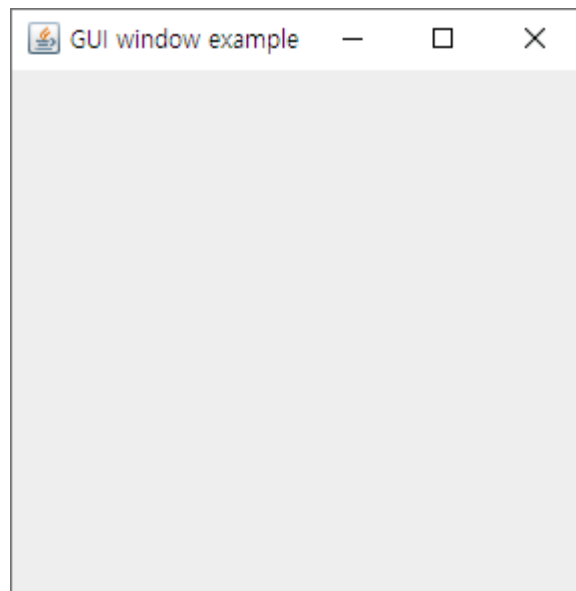
    public void run(){

        this.setTitle("GUI window example");
        setBounds(0,0,300,300);
        setVisible(true);
    }

    @Override
    public void actionPerformed(ActionEvent arg0) {
        // TODO Auto-generated method stub
    }

}
```

main method 에서  
인스턴스 생성 후 run()



# 예제 - GUI 프로그래밍 해 보기

```
private int count = 0;
private JButton btn = new JButton("click");
private JLabel label = new JLabel(count + "번 눌렀습니다.");
```

```
public void run(){
    JPanel panel = new JPanel();

    btn.addActionListener(this);

    panel.add(btn);
    panel.add(label);

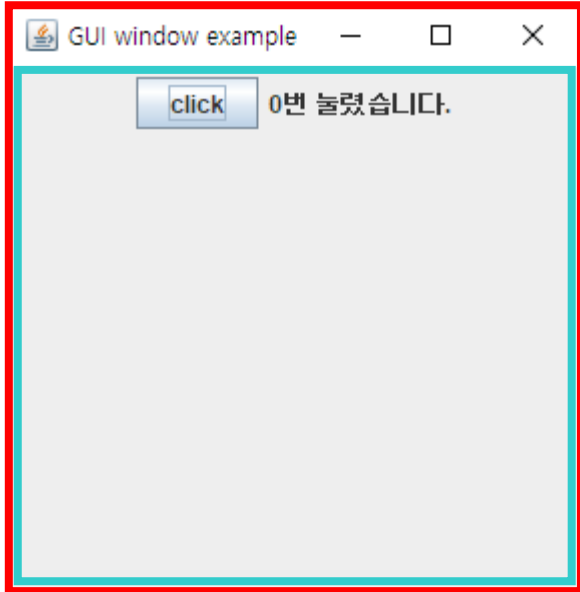
    this.add(panel);

    this.setTitle("GUI window example");
    setBounds(0,0,300,300);
    setVisible(true);
}
```

```
@Override
public void actionPerformed(ActionEvent e) {
    count++;
    label.setText(count + "번 눌렀습니다.");
}
```



# 예제 - GUI 프로그래밍 해 보기



```
@Override  
public void actionPerformed(ActionEvent e) {  
    count++;  
    label.setText(count + "번 눌렀습니다.");  
}
```

# 기말과제 - Notepad

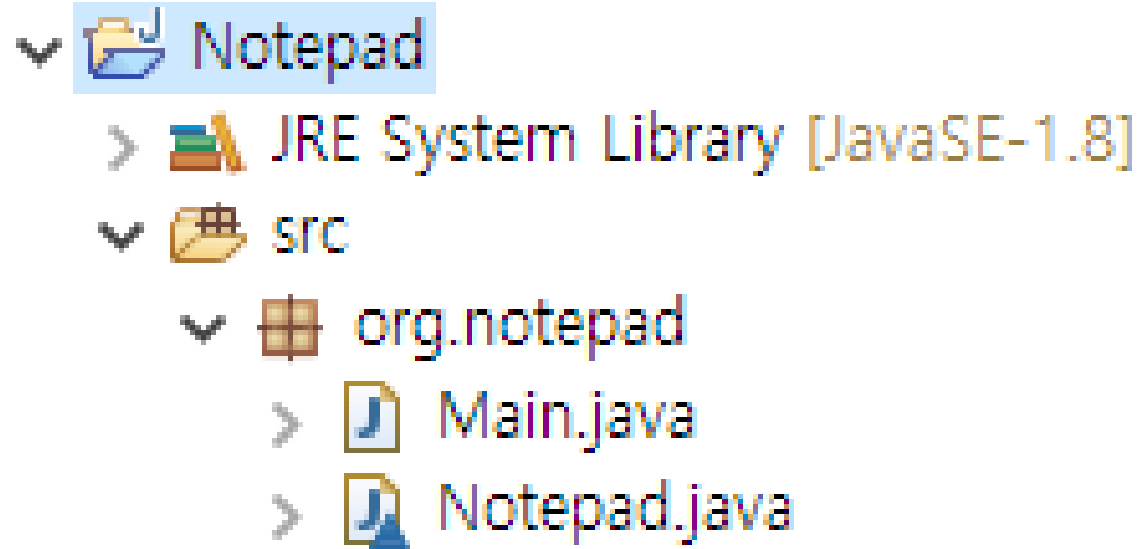
- 개인과제로 진행
- Runnable jar 생성해서 함께 제출



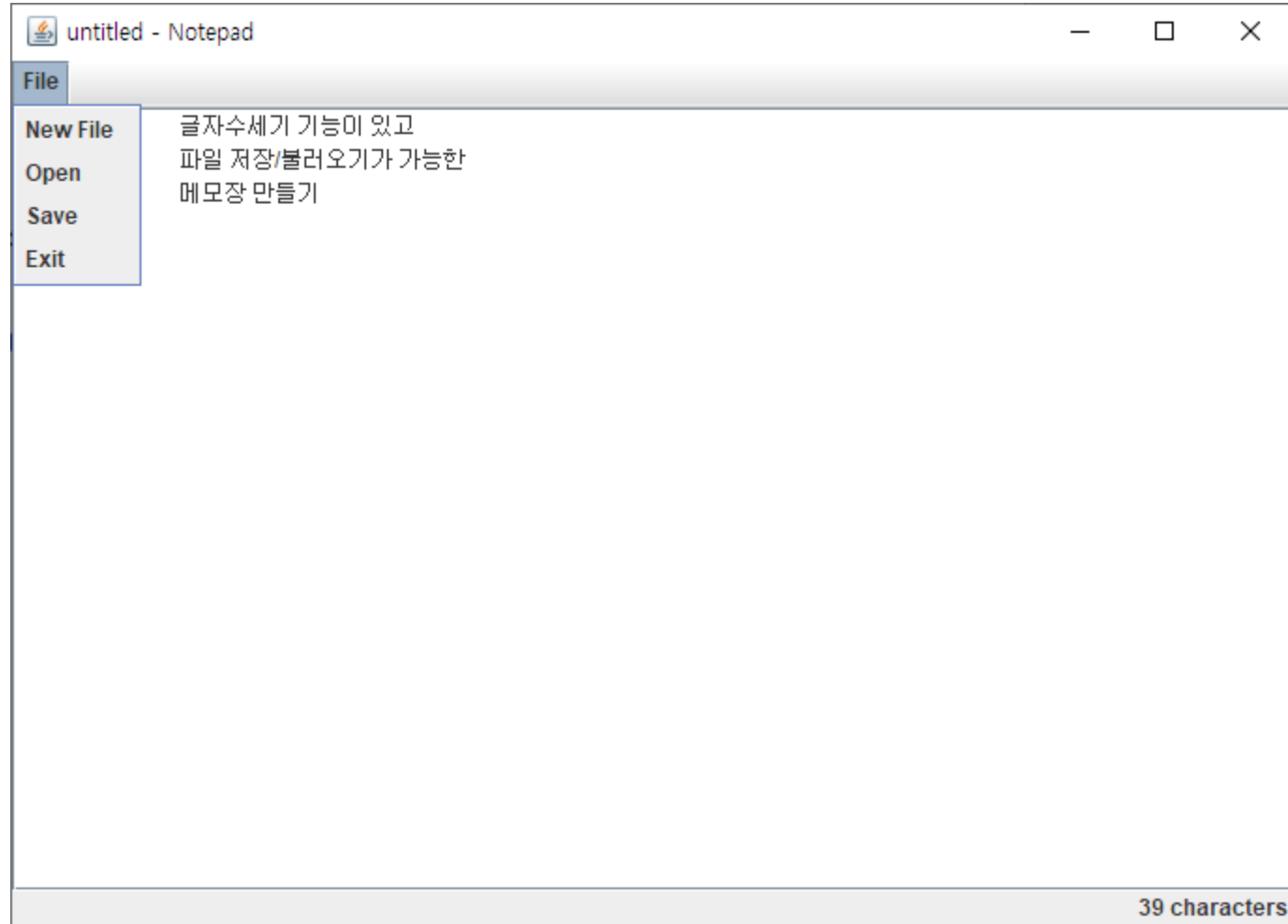
- 다음을 포함해야 함
  - 실행 스크린샷
  - Java project
  - 보고서(추가로 구현한 것이 있다면 기록)

# 예제 – Notepad

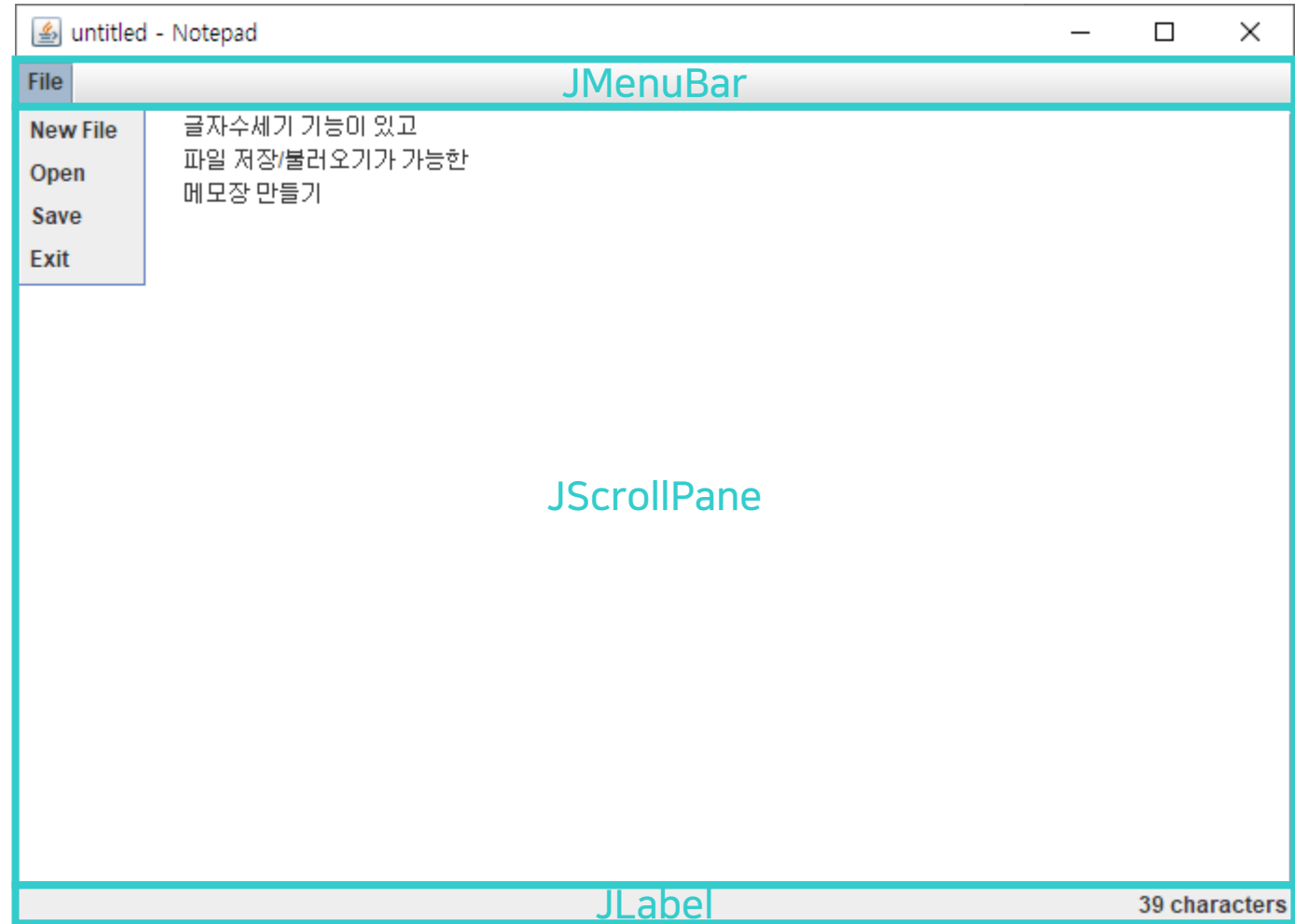
- 다음과 같이 새 Java project 생성



# 예제 – Notepad



# 예제 - Notepad



# 예제 – Notepad

org.notepad.Notepad (Notepad.java)

```
@SuppressWarnings("serial")
class Notepad extends JFrame{

    private JTextArea fileContents = new JTextArea();
    private JLabel fileStatus = new JLabel("0 characters ");
    private String fileName = "untitled";

//  private Notepad thisWindow = this;

//  private File file;
//  private JFileChooser jfc = new JFileChooser(".");
```

# 예제 – Notepad

org.notepad.Notepad (Notepad.java) public void run() (1/2)

```
public void run() {  
  
    this.setTitle(fileName+" - Notepad");  
  
    JMenuBar menuBar = new JMenuBar();  
    JMenu menuFile = new JMenu("File");  
  
        JMenuItem newFile = new JMenuItem("New File");  
        JMenuItem open = new JMenuItem("Open");  
        JMenuItem save = new JMenuItem("Save");  
        JMenuItem exit = new JMenuItem("Exit");  
  
    menuFile.add(newFile);  
    menuFile.add(open);  
    menuFile.add(save);  
    menuFile.add(exit);  
}
```

# 예제 – Notepad

org.notepad.Notepad (Notepad.java) public void run() (2/2)

```
menuBar.add(menuFile);
this.add(menuBar);
setJMenuBar(menuBar);

fileStatus.setHorizontalAlignment(SwingConstants.RIGHT);

JScrollPane scrollPane = new JScrollPane(fileContents);

this.add(scrollPane);
this.add("South", fileStatus);

/* JFrame 크기 지정 */
setBounds(300,300,700,500);

/* JFrame 화면에 띄움 */
setVisible(true);
}
```



# 예제 – Notepad

org.notepad.Main (Main.java)

```
package org.notepad;

public class Main {

    public static void main(String[] args) {

        Notepad myNp = new Notepad();
        myNp.run();

    }
}
```