

자바의 GUI 패키지 구버전: java.awt

- 신버전: javax.swing

상속의 키워드: extends

❖ Java 프로그램 작성법

- ✓ 메모장에서 Java 코드를 작성한 후에 *.java로 저장
- ✓ javac.exe를 사용해서 컴파일하면 *.class 파일이 생성
- ✓ java.exe를 사용해서 컴파일된 *.class 파일을 실행
- 개발자들은 대부분 이클립스 환경에서 Java를 개발

클래스 내에서 메소드의 이름이 같아도 파라미터만 다르게 하여 선언하는 방식: 메소드 오버로딩

클래스의 구성요소: 속성, 생성자, 메소드

기존 클래스를 그대로 물려받으면서 필요한 필드나 메소드 추가하기: 상속

부모 클래스의 명칭: super

자신 클래스의 명칭: this

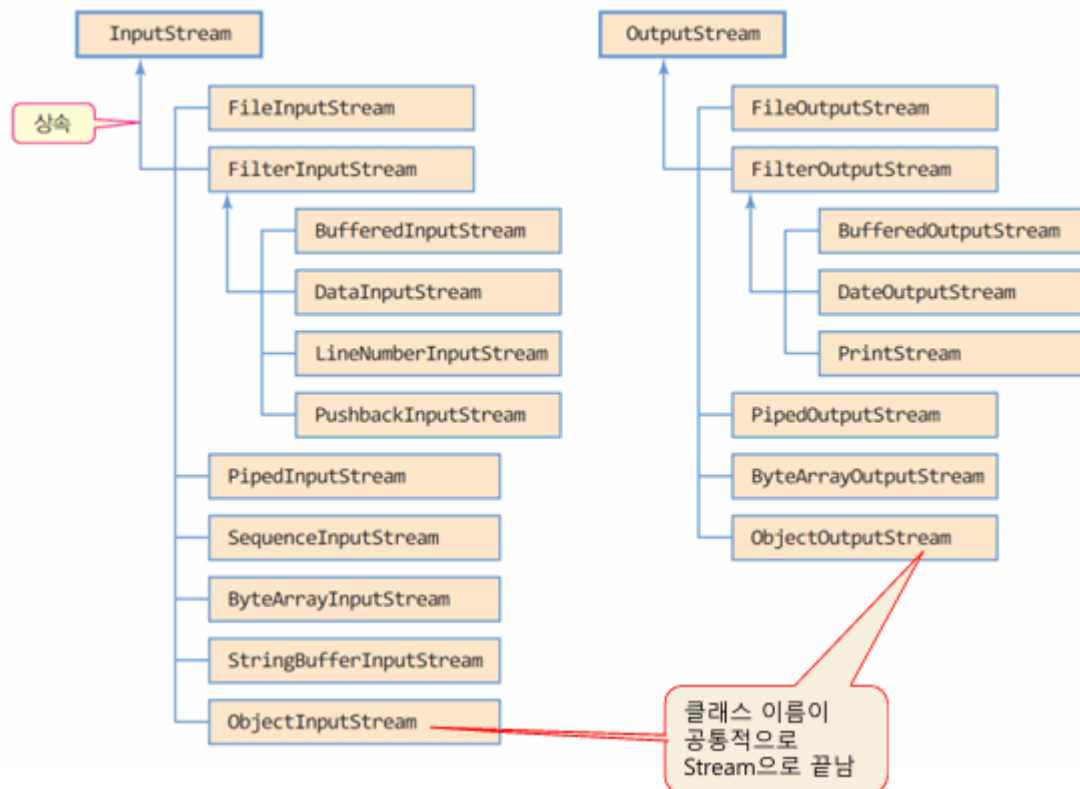
인스턴트화를 금지하는 클래스: 추상 클래스

메소드 본체가 없는 메소드: 추상메소드

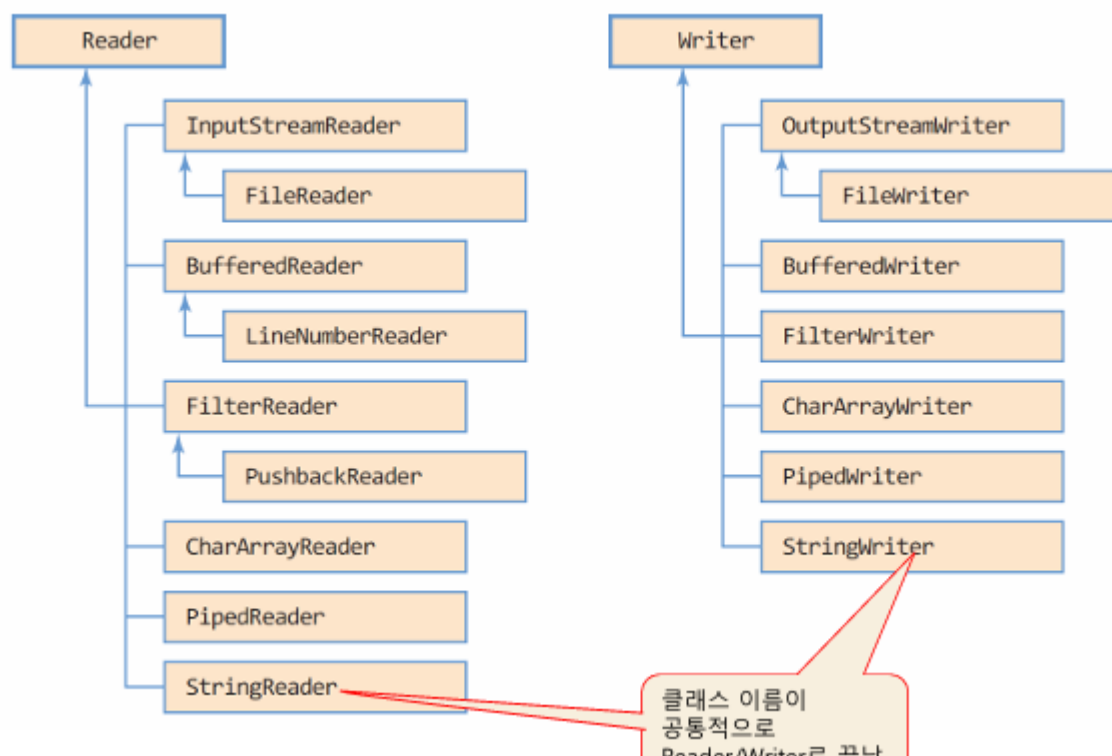
자바 코드의 묶음: 패키지

버퍼를 가지고 순차적으로 이루어지는 입출력: 스트림 입출력





위 계층 구조는 어떤 스트림의 계층 구조인가: 바이트 스트림



위 계층 구조는 어떤 스트림의 계층 구조인가: 문자스트림

스트림은 사용이 끝난 후 꼭 (close())를 해주어야 한다

스트림 사용시는 꼭 예외처리를 해주어야 한다: O, X

버퍼 스트림 사용 이유: 성능 개선

```
import java.io.*;

public class FileReadHangulSuccess {
    public static void main(String[] args) {
        InputStreamReader in = null;
        FileInputStream fin = null;
        try {
            fin = new FileInputStream("c:\\Temp\\Whangul.txt");
            in = new InputStreamReader(fin, "MS949");
            int c;

            System.out.println("인코딩 문자 집합은 " + in.getEncoding());
            while ((c = in.read()) != -1) {
                System.out.print((char)c);
            }
            in.close();
            fin.close();
        } catch (IOException e) {
            System.out.println("입출력 오류");
        }
    }
}
```

MS에서 만든 한글
완성형 문자 집합

```
import java.io.*;

public class BinaryCopyEx {
    public static void main(String[] args) {
        File src = new File("c:\\Windows\\Web\\Wallpaper\\Theme1\\Wimg1.jpg");
        File dest = new File("c:\\Temp\\copyimg.jpg");
        int c;
        try {
            FileInputStream fi = new FileInputStream(src);
            FileOutputStream fo = new FileOutputStream(dest);
            while((c = fi.read()) != -1) {
                fo.write((byte)c);
            }
            fi.close();
            fo.close();
            System.out.println(src.getPath() + "를 " +
                dest.getPath() + "로 복사하였습니다.");
        } catch (IOException e) {
            System.out.println("파일 복사 오류");
        }
    }
}
```



다른 GUI 컴포넌트를 포함할 수 있는 컴포넌트: 컨테이너

Swing Container 종류들: JPanel, JFrame, JApplet, JDialog, JWindow

다른 컨테이너에 속하지 않고 독립적으로 출력 가능한 컨테이너: 최상위 컨테이너

- 종류: JFrame, JDialog, JApplet

컨테이너에 포함되어야 화면에 출력될 수 있는 것: 컴포넌트

- 어떤 것을 상속 받나: javax.swing.JComponent

프레임 종료시 파일 종료 메소드: this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

배치 관리자 대표 4 유형: FlowLayout, BorderLayout, GridLayout, CardLayout

순차적 쌓임: FlowLayout

가로 새로 크기를 지정하여 순차적: GridLayout

위로 스택: CardLayout

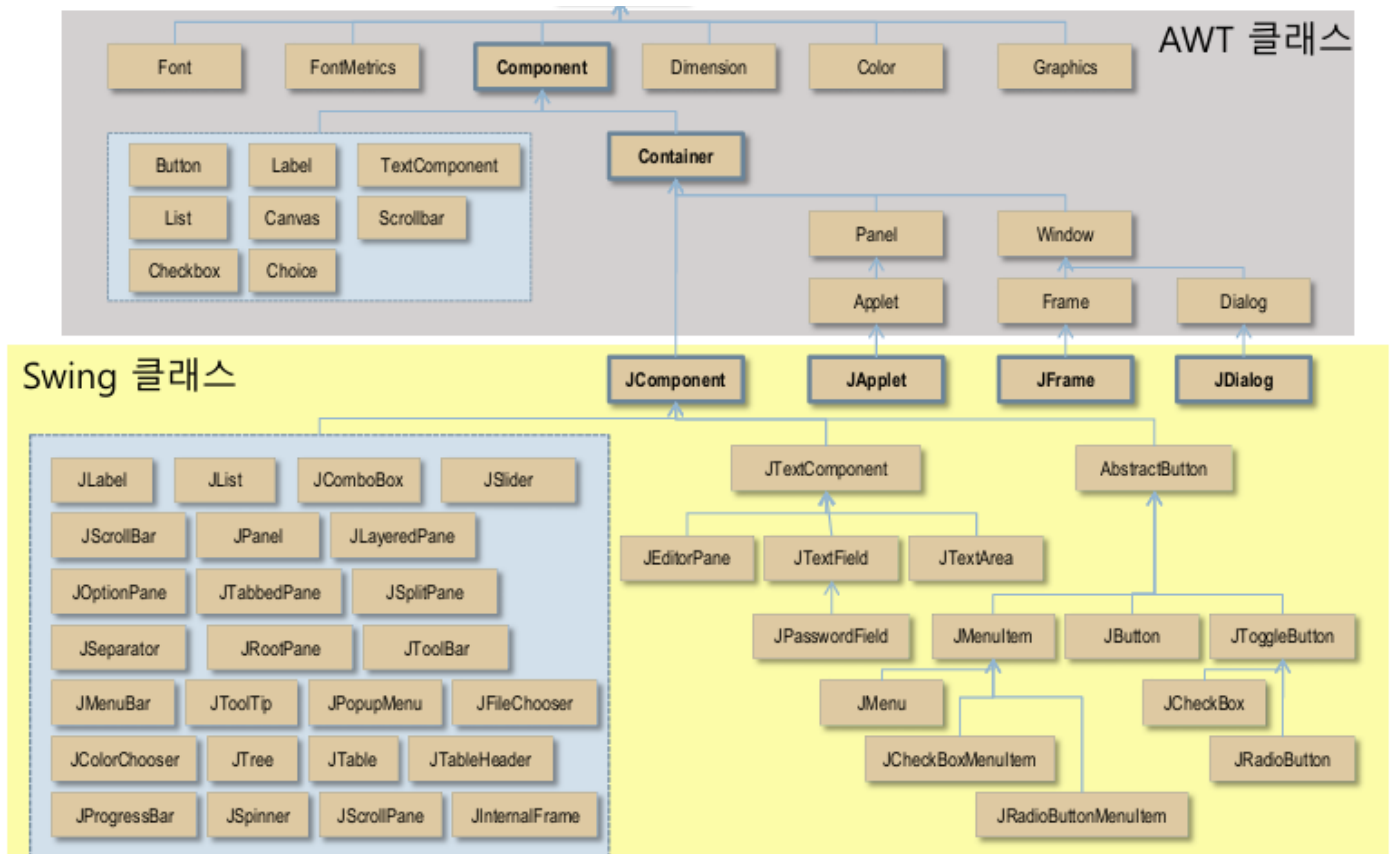
동서남북중앙: BorderLayout

레이아웃 설정하는 메소드: `this.setLayout(new 레이아웃());`



해당 레이아웃 생성 코드: `this.setLayout(GridLayout(4,3,5,5));`

대표적인 스윙 컴포넌트: JButton, JCheckBox, JTextField, JFrame, JPasswordField, JList 등



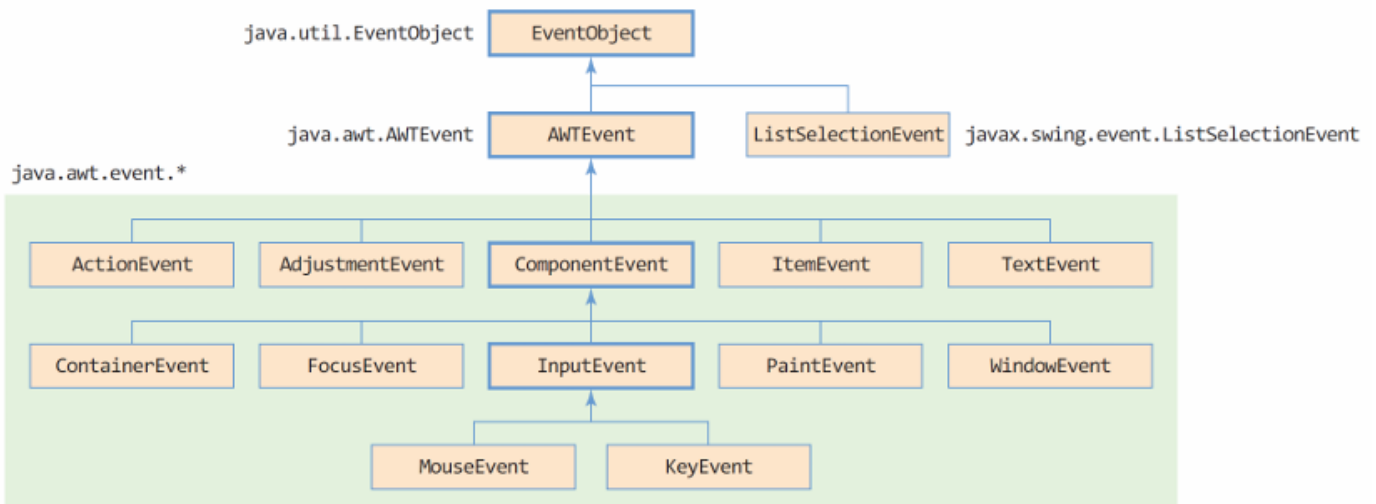
이벤트의 종류 대표적인거: 마우스 드래그, 마우스 클릭, 키보드 누름 등

이벤트 처리 순서: 이벤트 발생 -> 이벤트 객체 생성 -> 이벤트 리스너 찾기 -> 이벤트 리스너 호출 -> 이벤트 리스너 실행

이벤트를 발생시킨 GUI 컴포넌트: 이벤트 소스

발생한 이벤트에 대한 정보: 이벤트 객체

이벤트를 처리하는 코드: 이벤트 리스너



이벤트 객체	이벤트 소스	이벤트가 발생하는 경우
ActionEvent	JButton	마우스나 <Enter> 키로 버튼 선택
	JMenuItem	메뉴 아이템 선택
	TextField	텍스트 입력 중 <Enter> 키 입력
ItemEvent	CheckBox	체크박스의 선택 혹은 해제
	JRadioButton	라디오버튼의 선택 상태가 변할 때
	CheckBoxMenuItem	체크박스 메뉴 아이템의 선택 혹은 해제
ListSelectionEvent	JList	리스트에서 선택된 아이템이 변경될 때
KeyEvent	Component	키가 눌러지거나 눌러진 키가 떼어질 때
MouseEvent	Component	마우스 버튼이 눌러지거나 떼어질 때, 마우스 버튼이 클릭될 때, 컴포넌트 위에 마우스가 올라갈 때, 올라간 마우스가 내려올 때, 마우스가 드래그될 때, 마우스가 단순히 움직일 때
FocusEvent	Component	컴포넌트가 포커스를 받거나 잃을 때
WindowEvent	Window	Window를 상속받는 모든 컴포넌트에 대해 윈도우 활성화, 비활성화, 아이콘화, 아이콘에서 복구, 윈도우 열기, 윈도우 닫기, 윈도우 종료
AdjustmentEvent	JScrollBar	스크롤바를 움직일 때
ComponentEvent	Component	컴포넌트가 사라지거나, 나타나거나, 이동, 크기 변경 시
ContainerEvent	Container	Container에 컴포넌트의 추가 혹은 삭제

버튼에 리스너 추가하는 코드: JButton.addActionListener();



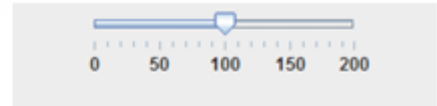
JButton



JCheckBox



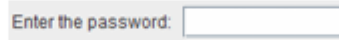
JRadioButton



JSlider



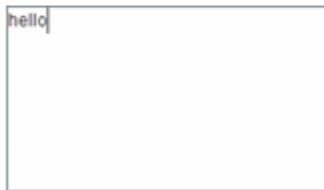
JTextField



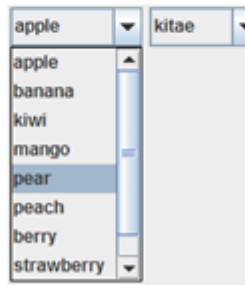
JPasswordField



JSpinner



JTextArea



JComboBox



JList

```

- btn.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        JButton b = (JButton)e.getSource();
        if(b.getText().equals("Action"))
            b.setText("액션");
        else
            b.setText("Action");
        setTitle(b.getText());
    }
});

```

Jframe 활용시 필요한 필수 코드

setTitle("title")

setDefaultCloseAction(JFrame.EXIT_ON_CLOSE);

setSize(x, y);

setVisible(true);