



제28장

AWT-GUI의 기초_Part2



Swing 이란

- AWT 기술을 기반으로 작성된 자바 라이브러리
 - : 모든 AWT 기능 + 추가된 풍부하고 화려한 고급 컴포넌트
 - : AWT 컴포넌트에 J자가 덧붙여진 이름의 클래스
 - : J 자로 시작하는 클래스
- javax.swing 패키지
- Swing 컴포넌트는 운영체제에 의존하지 않음
- 경량 컴포넌트(Light weight components)





Swing GUI 어플리케이션 만들기

1. 프레임 만들기
2. 프레임에 스윙 컴포넌트 붙이기
3. main() 메소드 작성

스윙 프로그램을 작성하기 위한 import문

```
import java.awt.*; // 그래픽 처리를 위한 클래스들의 경로명
import java.awt.event.*; // AWT 이벤트 사용을 위한 경로명
import javax.swing.*; // 스윙 컴포넌트 클래스들의 경로명
import javax.swing.event.*; // 스윙 이벤트를 위한 경로명
```



Swing 컴포넌트의 공통메소드

컴포넌트의 모양과 관련된 메소드

```
void setForeground(Color) 전경색 설정  
void setBackground(Color) 배경색 설정  
void setOpaque(boolean) 불투명성 설정  
void setFont(Font) 폰트 설정  
Font getFont() 폰트 리턴
```

컴포넌트의 위치와 크기에 관련된 메소드

```
int getWidth() 폭 리턴  
int getHeight() 높이 리턴  
int getX() x 좌표 리턴  
int getY() y 좌표 리턴  
Point getLocationOnScreen() 스크린 좌표상에서의 컴포넌트 좌표  
void setLocation(int, int) 위치 지정  
void setSize(int, int) 크기 지정
```



Swing 컴포넌트의 공통메소드

컴포넌트의 상태와 관련된 메소드

void *setEnabled(boolean)* 컴포넌트 활성화/비활성화
void *setVisible(boolean)* 컴포넌트 보이기/숨기기
boolean *isVisible()* 컴포넌트의 보이는 상태 리턴

컨테이너를 위한 메소드

Component add(Component) 자식 컴포넌트 추가
void remove(Component) 자식 컴포넌트 제거
void removeAll() 모든 자식 컴포넌트 제거
Component[] getComponents() 자식 컴포넌트 리스트 리턴
Container getParent() 부모 컨테이너 리턴
Container getTopLevelAncestor() 최상위 부모 컨테이너 리턴



버튼 컴포넌트

버튼 모양의 컴포넌트

버튼은 클릭될 때 Action 이벤트를 발생시킴

버튼 컴포넌트 생성

`.JButton()`

빈 버튼 생성

`JButton(Icon icon)`

이미지 아이콘만 가진 버튼 생성

`JButton(String text)`

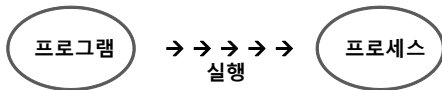
텍스트만 가진 버튼 생성

`JButton(String text, Icon icon)`

텍스트와 이미지 아이콘 모두 가진 버튼 생성



Process(프로세스)와 Thread(쓰레드)

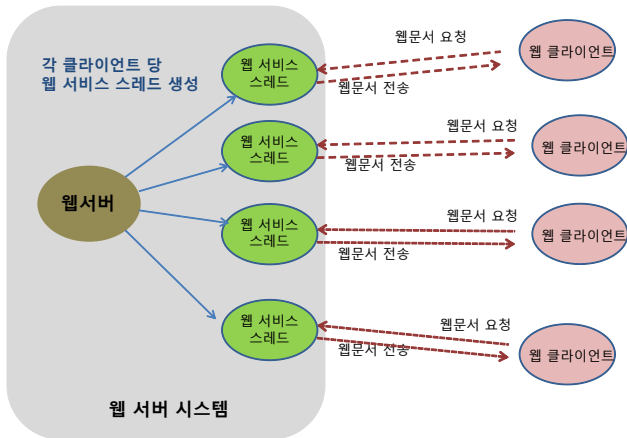


- ▶ 프로그램 : 실행 가능한 파일(HDD)
- ▶ 프로세스 : 실행 중인 프로그램(메모리)

쓰레드 : 프로세스 내에서 실제 작업을 수행.
모든 프로세스는 하나 이상의 쓰레드를 가지고 있다.
경량 프로세스(light-weight process)



Multi-Thread(멀티쓰레드)





스레드 만드는 2 가지 방법

java.lang.Thread 클래스를 이용하는 경우

java.lang.Runnable 인터페이스를 이용하는 경우



- 생성자
 - Thread()
 - Thread(Runnable target)
 - Thread(String name)
 - Thread(Runnable target, String name)
- 스레드 시작시키기
 - void start()
- 스레드 코드
 - void run()
- 스레드 잠자기
 - static void sleep(long mills)
- 다른 스레드 죽이기
 - void interrupt()
- 다른 스레드에게 양보
 - static void yield()
 - 현재 스레드의 실행을 중단하고 다른 스레드가 실행될 수 있도록 양보한다.



- 다른 스레드가 죽을 때까지 기다리기
 - `void join()`
- 현재 스레드 객체 알아내기
 - `static Thread currentThread()`
- 스레드 ID 알아내기
 - `long getId()`
- 스레드 이름 알아내기
 - `String getName()`
- 스레드 우선순위값 알아내기
 - `int getPriority()`
- 스레드의 상태 알아내기
 - `Thread.State getState()`

감사합니다!