

Java講座

第6回

情報科学部コンピュータ科学科
2年 竹中 優

今回の内容

- ◆ アプレットについて
 - ◆ Appletクラスの仕組み
- ◆ アプレットに描画してみよう
 - ◆ Graphicsクラス

タブレットについて

アプレットの仕組み

- ◆ Webブラウザ上で動くプログラムはアプレット (applet) と呼ばれている。
- ◆ アプレットに対して、これまで作成してきたようなコマンドプロンプトから起動するプログラムを、アプリケーションと呼ぶ。
(本講座ではコマンドプロンプトからの実行は説明していない)

アプレットの仕組み

- ◆ 今回は、Webブラウザを利用することが目的ではないが、描画系の動作や処理がもともと備わっていたり、比較的簡単に出来るので、アプレットを紹介する。

アプレットの仕組み

- ◆ アプレットを作成するには、Appletクラスを拡張したクラスを宣言することになっている。
- ◆ 次頁に最も基本的なアプレットのコードを示す。

アプレットの仕組み

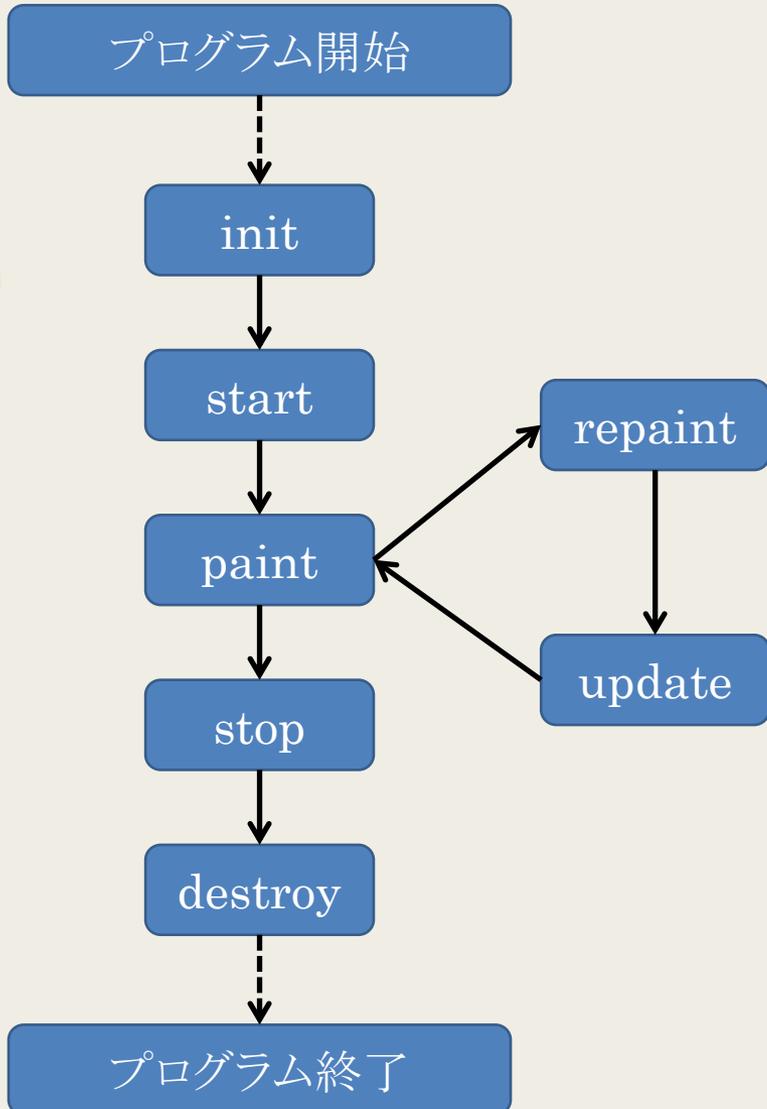
```
public class Sample_Applet01 extends Applet{  
    public void paint(Graphics g){  
        g.drawString("Hello world!", 10, 50);  
    }  
}
```

- ◆ インポートは、[Ctrl+Shift+o]で行おう。
- ◆ このプログラムは画面に”Hello world!”と表示するアプレットである。
- ◆ 実行のやり方はEclipseの場合、
メニュー→実行→Javaアプレット

アプレットの仕組み

- ◆ アプリケーションと最も異なるのは、mainメソッドが存在しないことだろう。
アプレットはpaintメソッドが(内部で勝手に)呼ばれることで画面に描画が行われる。
- ◆ Graphicsオブジェクトは描画に関する様々なメソッドを保有し、このオブジェクトを利用することで、描画が行える。
- ◆ 次頁から詳しくAppletクラスを見ていこう。

アプレットの仕組み



アプレットの重要なメソッド

```
public void init()
public void start()
public void repaint()
public void update(Graphics g)
public void paint(Graphics g)
public void stop()
public void destroy()
```

アプレットの仕組み

◆ initメソッド

最初に一度だけ呼び出される。初期化処理を行うと良い。またスレッドの作成もここで行う。もとのAppletクラスでは何も行わない。

◆ startメソッド

Webブラウザ上でアプレットが有効になるたびに呼ばれる。もとのAppletクラスでは何も行わない。

アプレットの仕組み

- ◆ paintメソッド
Graphicsオブジェクトを引数として受け取り、画面に描画を行う。
- ◆ updateメソッド
Graphicsオブジェクトを引数として受け取り、paintメソッドを呼び出す。
- ◆ repaintメソッド
Graphicsオブジェクトを取得して、updateメソッドを呼び出す。オーバーライドする必要は無い。

アプレットの仕組み

- ◆ stopメソッド
アプレットを実行しているWebページの代わりに別のWebページがされた時か、アプレットが終了する直前に呼び出される。
もとのAppletクラスでは何も行わない。
- ◆ destroyメソッド
アプレットが終了する際にstopメソッドの次に呼び出される。
もとのAppletクラスでは何も行わない。

アプレットの仕組み

- ◆ サンプルコード: `Sample_Applet.java` を実行してみよう。

タブレットに描画してみよう

Graphicsクラス

- ◆ 画面に描画するGraphicsクラスを使用するため、そのメソッドを見てみよう。

Graphicsクラスのメソッド

- ◆ `void drawString(String str, int x, int y)`
(x, y)から右に文字列strを描画する
- ◆ `void drawLine(int sx, int sy, int ex, int ey)`
(sx, sy)から(ex, ey)へ線を引く
- ◆ `void drawRect(int x, int y, int width, int height)`
(x, y)を左上の点とし、幅width、高さheightの四角形を描画する
- ◆ `void fillRect(int x, int y, int width, int height)`
`drawRect(x, y, width, height)`の四角形の内側を塗りつぶしたものを描画する

Graphicsクラスのメソッド

- ◆ `void drawOval(int x, int y, int width, int height)`
`drawRect(x, y, width, height)`の四角形にぴったり収まる楕円を描画する
- ◆ `void fillOval(int x, int y, int width, int height)`
`drawOval(x, y, width, height)`の楕円に塗りつぶしを行ったものを描画する
- ◆ `void setColor(Color color)`
色を変更する。`Color.RED`などのクラス定数を渡す。
- ◆ `void setFont(Font font)`
描画する文字列のフォントを指定する

Appletクラスのメソッド

- ◆ `int getX()`
画面の原点のx座標を返す
- ◆ `int getY()`
画面の原点のy座標を返す
- ◆ `int getWidth()`
画面の幅を返す
- ◆ `int getHeight()`
画面の高さを返す
- ◆ `Color getBackground()`
背景色を返す
- ◆ `void setSize(int w, int h)`
画面の幅、高さを設定する

アニメーションについて

- ◆ プログラムでアニメーションを行いたい場合、一定間隔で画面の再描画を行う必要がある。
- ◆ 例えば、0.1秒ごとに再描画を行う場合。
画面を表示→描画→(0.1秒待機)→
画面内容を更新→再描画→(0.1秒待機)→...
という感じ。
- ◆ アニメーションを行うにはスレッドを利用する必要がある。
サンプルコード：
`java_lec06.samples.Sample_Runnable.java`

問題1

- ◆ 画面上でボールが跳ね回るアニメーションをアプレットで作成せよ。

アプレットを継承するクラスはMovingBallとし、ボールはBallクラスを作成する。

座標 x , y と、再描画ごとに移動する距離 dx , dy を全てdouble型のフィールドで宣言せよ。

スレッドについてはSample_Applet.javaを参考にせよ。

終わり