How to make an animation loop in Java ME

This article shows how to create an animation loop and a small animation using a thread. The animation has a rectangle moving on the screen and when it collides with the screen borders it changes direction. The loop runs until the user exits the program.

The AnimationLoopMIDlet.java midlet

```
import javax.microedition.lcdui.Display;
import javax.microedition.midlet.MIDlet;

public class AnimationLoopMIDlet extends MIDlet {

 static AnimationLoopMIDlet instance;

 public AnimationLoopMIDlet()
 {
  AnimationLoopMIDlet.instance = this;
 }

 /**Overrides abstract 'startApp' method of javax.microedition.midlet.MIDlet.
  */
 public void startApp() {
  Display.getDisplay(this).setCurrent(new AnimationLoop());
 }

 /**Overrides abstract 'pauseApp' method of javax.microedition.midlet.MIDlet
  */
 public void pauseApp() {
 }

 /**Overrides abstract 'destroyApp(boolean unconditional)' method of
javax.microedition.midlet.MIDlet.
  */
 public void destroyApp(boolean unconditional) {
 }

}
```

The AnimationLoop.java class

```
import javax.microedition.lcdui.Command;
import javax.microedition.lcdui.CommandListener;
import javax.microedition.lcdui.Displayable;
import javax.microedition.lcdui.Graphics;
import javax.microedition.lcdui.game.GameCanvas;

public class AnimationLoop extends GameCanvas implements Runnable, CommandListener{

 // indicates if the animation loop is running
 private boolean running;

 // screen height
 private int screenHeight;
```


(C) Copyright Nokia 2014. All rights reserved.

```
// screen width
private int screenWidth;

// Rectancle coordinates
private float rectX;
private float rectY;
private int rectW;
private int rectH;

// Animation velocity
private float velX;
private float velY;


private Command exit;

/*
 *  AnimationLoop constructor
 */
public AnimationLoop() {

 super(false);
 setFullScreenMode(false);

 initialize();

 start();

}

/*
 *  Initialize variables
 */
protected void initialize()
{
 exit = new Command("Exit", Command.EXIT, 1);
 addCommand(exit);
 setCommandListener(this);

 screenWidth = getWidth();
 screenHeight = getHeight();

 velX = 0.1f;
 velY = 0.1f;

 rectX = 0.0f;
 rectY = 0.0f;
 rectW = 60;
 rectH = 30;
}

public void commandAction(Command command, Displayable displayable) {
 if (command == exit) {
  stop();
 }
}
```


(C) Copyright Nokia 2014. All rights reserved.

```
/*
 *  Starting of the thread
 */
void start()
{
 Thread thread = new Thread(this);
 thread.setPriority(Thread.MAX_PRIORITY);
 thread.start();
}

/*
 *  stopping the loop
 */
void stop()
{
 running = false;
}

/*
 * Loop to draw the animation
 */
public void run() {
 Graphics g = getGraphics();

 running = true;

 int dt = 0;

 long loopStartTime = System.currentTimeMillis();

 while(running){
  dt = (int) (System.currentTimeMillis()-loopStartTime);
  if(dt>0)
  {
   loopStartTime = System.currentTimeMillis();

   update(dt);
   draw(g);

   flushGraphics();
  }

 }

 quit();
}

/*
 * movement calculations
 */
private void update(int dt){
 // simple motion calculations
 float tempX = rectX + velX*dt;
 float tempY = rectY + velY*dt;

 //  collision check for horizontal screen borders
```


(C) Copyright Nokia 2014. All rights reserved.

```
   if(tempX+rectW >= screenWidth){
    tempX = screenWidth-rectW-(tempX+rectW-screenWidth);
    velX *= -1.0f;
   }else if(tempX<0){
    tempX *= -1.0f;
    velX *= -1.0f;
   }

   //  collision check for vertical screen borders
   if(tempY+rectH >= screenHeight){
    tempY =screenHeight-rectH-(tempY+rectH-screenHeight);
    velY *= -1.0f;
   }else if(tempY<0){
    tempY *= -1.0f;
    velY *= -1.0f;
   }

   rectX = tempX;
   rectY = tempY;
  }

  /*
   *  rendering the screen
   */
  private void draw(Graphics g){
   // fill background
   g.setColor(255,255,255);
   g.fillRect(0, 0, screenWidth, screenHeight);

   // draw the rectangle on screen
   g.setColor(255,0,0);
   g.fillRect((int)rectX,(int)rectY,rectW,rectH);
  }

  /*
   *  Exiting the midlet
   */
  private void quit()
  {
   AnimationLoopMIDlet.instance.notifyDestroyed();
  }

 }
```

Resources

The source files are available here: File:AnimationLoopSource.zip

The binary files are available here: File:AnimationLoopBinaries.zip

(C) Copyright Nokia 2014. All rights reserved.

(C) Copyright Nokia 2014. All rights reserved.