```
 //Frames Example using sprites for a game
 //Lucian Novosel - 2014
 //Open sourcerer on the internet of things
 // WWW: luciannovosel.com GH: github.com/luciannovo

/*
```


```
    Frames animation library for Processing.
    Copyright (c) 2012-2013 held jointly by the individual authors.

    This file is part of Frames animation library for Processing.

    Frames animation library for Processing is free software: you can redistr
    modify it under the terms of the GNU General Public License as published
    the Free Software Foundation, either version 3 of the License, or
    (at your option) any later version.

    Frames animation library for Processing is distributed in the hope that i
    useful, but WITHOUT ANY WARRANTY; without even the implied warranty of
    MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
    GNU General Public License for more details.

    You should have received a copy of the GNU General Public License
    along with Frames animation library for Processing.  If not, see
    <http://www.gnu.org/licenses/>.
```


```
*/
import java.util.List;

import org.dishevelled.processing.frames.Animation;
import org.dishevelled.processing.frames.Frames;

Frames frames;

Animation metalGearSoldier_loop;
Animation megaMan_loop;
Animation cat_loop;
Animation catRun_loop;
Animation catWalk_loop;

Animation currentAnimation;
int px = width / 2;
int py = height / 2;
boolean right = true;
boolean infected = false;
```

```
boolean deceased = false;


void setup()
{
  size(16*30, 9*30);
  frameRate(8);
  background(20);
  frames = new Frames(this);

//  List<PImage> metalGearSoldier_frames = frames.createFrameList("goldenarmy
//  metalGearSoldier_loop = frames.createLoopedAnimation(metalGearSoldier_fra
//
//  List<PImage> megaMan_frames = frames.createFrameList("cobalt.png", 0, 0,
//  megaMan_loop = frames.createLoopedAnimation(megaMan_frames);

  List<PImage> cat_frames = frames.createFrameList("Grizzo_Cat.png", 0, 0, 42
  cat_loop = frames.createLoopedAnimation(cat_frames);
//  cat_loop_left = frames.createLoopedAnimation(hatDeceasedFrames);

  List<PImage> catWalk_frames = frames.createFrameList("Grizzo_Cat.png", 0, 6
  catWalk_loop = frames.createLoopedAnimation(catWalk_frames);
//  cat_loop_left = frames.createLoopedAnimation(hatDeceasedFrames);

  List<PImage> catRun_frames = frames.createFrameList("Grizzo_Cat.png", 0, 0,
  catRun_loop = frames.createLoopedAnimation(catRun_frames);
//  cat_loop_left = frames.createLoopedAnimation(hatDeceasedFrames);

  currentAnimation = cat_loop;
}

void draw()
{
  fill(20);
  rect(0, 0, width, height);

  updateAnimation();
  currentAnimation.advance();
  image(currentAnimation.getCurrentFrame(), px, py);
}


void keyPressed() {
  if (key == CODED) {
    switch (keyCode) {
```

```
      case DOWN:  currentAnimation = cat_loop; break;
      case UP:    currentAnimation = cat_loop; break;
      case LEFT:  currentAnimation = catWalk_loop; break;
      case RIGHT: currentAnimation = catRun_loop; break;
    }
  }
}


void mousePressed()
{
  if (!infected)
  {
    infected = true;
  }
  else if (!deceased)
  {
    deceased = true;
  }
}

void mouseMoved()
{
  if (!deceased)
  {
    if ((mouseX - 12) > px)
    {
      right = true;
    }
    else
    {
      right = false;
    }
    px = mouseX - 12;
    py = mouseY - 24 - 8;
  }
}

void updateAnimation()
{}
```