**IMAGES**

Dangerous way

**File** file **= new File("1.jpg");** - getting directory **BufferedImage** image **= ImageIO.read(file);** - getting image ( can have errors)

formal way

**File** file **= new File("1.jpg");  
ImageReader** reader **= null;** - set reader null **Iterator<ImageReader>** iterator **= ImageIO.getImageReadersByFormatName("**…**"); - create iterator  
if(**iterator**.hasNext()){** reader **= iterator.next();** - set reader **}  
  
InputStream** inputStream **= new FileInputStream(new File("…"));  
ImageInputStream** imageInputStream **= ImageIO.createImageInputStream(inputStream);** - input for imagesreader**.setInput(imageInputStream, true);** - setting input **BufferedImage** image **= reader.read(reader.getNumImages(true));** - getting image **int** count **= reader.getNumThumbnails(0);** - number of thumbnails **BufferedImage** image2 **= reader.readThumbnail(0,count);** - get thumbnails

Change pixels

**BufferedImage** img **= ImageIO.read(new File("**where**"));** - getting file **WritableRaster** raster **=** img**.getRaster();** - getting raster (needed variable)

**Int[] …** = raster**.getPixel(**i**,** j**, new int[**4**])** – returns int array with 4 variables of RGBA format GETTING PIXELS

raster**.setPixel(**i**,** j**,** array**)** – setting pixels

**for (int i = 0; i <** raster**.getWidth(); i++) {** - cross the whole width **int[]** pixel **=** raster**.getPixel(**x**,** y**, new int[4]);** - get pixels by coordinates and save them in variablepixel**[0] = …;** - set red from rgb formatpixel**[1] = …;** - set green from rgb formatpixel**[2] = …;** - set blue from rgb format **raster.setPixel(**x**,** y**,** pixel**);** - set new pixel **}**img**.setData(**raster**);** - setting this raster to the img **ImageIO.write(**img**, "**format**", new File("**where and with name…**"));** - save img

Transformation and filters

Classes with methods for rendering:

**AffineTransform**

**RescaleOp**

**LookupOp**

**ColorConvertOp**

**ConvolveOp**

Example:

**BufferedImage** bufferedImage **= new BufferedImage(**img**.getWidth(),**img**.getHeight(),**img**.getType());  
AffineTransform** affineTransform **= new AffineTransform();  
BufferedImageOp** op **= new AffineTransform(affineTransform, AffineTransform. ...);**op**.filter(**image**,** bufferedImage**);**

**ImageIO.write(**image**, "png", new File("2.png"));**