**Filled in the Sprite’s pixels with the sheet’s pixels in the loadGrey class, also had to change the Gray word to Grey. Also linked the method loadGrey to the constructor variables in the GreySpriteBuilder class**

**package** sonar.gamestates.states.levels.stages.entities;

**import** java.awt.image.BufferedImage;

**import** java.io.IOException;

**import** javax.imageio.ImageIO;

**public** **interface** SpriteBuilder

{

**int** getWidth();

**int** getHeight();

**int**[] getPixels();

}

**class** HUDSpriteBuilder **implements** SpriteBuilder

{

**private** SpriteSize size;

HUDSpriteBuilder(SpriteSize size, **int** colour)

{

**this**.size = size;

loadHUD(size, colour);

}

**private** **void** loadHUD(SpriteSize size, **int** colour)

{

**for**(**int** y = 0; y < size.getHeight(); y++)

{

**for**(**int** x = 0; x < size.getWidth(); x++)

{

size.getPixels()[x + y \* size.getWidth()] = colour;

}

}

}

**public** **int** getWidth(){**return** size.getWidth();}

**public** **int** getHeight(){**return** size.getHeight();}

**public** **int**[] getPixels(){**return** size.getPixels();}

}

**class** ColourSpriteBuilder **implements** SpriteBuilder

{

**private** SpriteSize size;

ColourSpriteBuilder(SpriteSize size, SpriteLocation location)

{

**this**.size = size;

loadColour(size, location);

}

**private** **void** loadColour(SpriteSize size, SpriteLocation location)

{

**int** xw = location.getX() \* size.getWidth();

**int** yh = location.getY() \* size.getHeight();

**for**(**int** y = 0; y < size.getHeight(); y++)

{

**for**(**int** x = 0; x < size.getWidth(); x++)

{

size.getPixels()[x + y \* size.getWidth()] = location.getSheet().getPixels()[(x + xw) + (y + yh) \* location.getSheet().getSize()];

}

}

}

**public** **int** getWidth(){**return** size.getWidth();}

**public** **int** getHeight(){**return** size.getHeight();}

**public** **int**[] getPixels(){**return** size.getPixels();}

}

**class** GreySpriteBuilder

{

**private** **int**[] colours;

**private** SpriteSize size;

GreySpriteBuilder(SpriteSize size, SpriteLocation location, **int**... colours)

{

**this**.size = size;

**this**.colours = colours;

loadGrey(size, location);

}

**private** **void** loadGrey(SpriteSize size, SpriteLocation location)

{

**int** xw = location.getX() \* size.getWidth();

**int** yh = location.getY() \* size.getHeight();

**for**(**int** y = 0; y < size.getHeight(); y++)

{

**for**(**int** x = 0; x < size.getWidth(); x++)

{

size.getPixels()[x + y \* size.getWidth()] = location.getSheet().getPixels()[(x + xw) + (y + yh) \* location.getSheet().getSize()];

}

}

}

}

**class** SpriteLocation

{

**private** **int** x, y;

**private** SpriteSheet sheet;

SpriteLocation(**int** x, **int** y, SpriteSheet sheet)

{

**this**.x = x;

**this**.y = y;

**this**.sheet = sheet;

}

**int** getX(){**return** x;}

**int** getY(){**return** y;}

SpriteSheet getSheet(){**return** sheet;}

}

**class** SpriteSheet

{

**private** **int**[] pixels;

**private** **int** size;

SpriteSheet(String path)

{

loadSheet(path);

}

**private** **void** loadSheet(String path)

{

BufferedImage image;

**try**

{

image = ImageIO.*read*(SpriteSheet.**class**.getResource(path));

**int** width = image.getWidth();

**int** height = image.getHeight();

size = width;

pixels = **new** **int**[width \* height];

image.getRGB(0, 0, width, height, pixels, 0, width);

}

**catch** (IOException e){e.printStackTrace();}

}

**int**[] getPixels(){**return** pixels;}

**int** getSize(){**return** size;}

}

**class** SpriteSize

{

**private** **int** width, height;

**private** **int**[] pixels;

SpriteSize(**int** width, **int** height)

{

**this**.width = width;

**this**.height = height;

pixels = **new** **int**[width \* height];

}

**int** getWidth(){**return** width;}

**int** getHeight(){**return** height;}

**int**[] getPixels(){**return** pixels;}

}