# How to add buttons in Gdx

rst I create my button image, add it to the assets folder and load the texture region. Now I make a sprite out of it.

Sprite button1=new Sprite(myTextureRegion);

To check if the button is touched I can use the rectangle from the sprite to check if you touched the image. In your touchUp method you will do something like

if(button1.getBoundingRectangle.contains(screenX,screenY))

# [java - merge adiacent rectangles into a polygon](http://stackoverflow.com/questions/20997254/java-merge-adiacent-rectangles-into-a-polygon)

[**Rectangle**](https://libgdx.badlogicgames.com/nightlies/docs/api/com/badlogic/gdx/math/Rectangle.html#Rectangle-float-float-float-float-)(x start, y start, float width, float height)

Constructs a new rectangle with the given corner point in the bottom left and dimensions.

[**merge**](https://libgdx.badlogicgames.com/nightlies/docs/api/com/badlogic/gdx/math/Rectangle.html#merge-com.badlogic.gdx.math.Rectangle-)([**Rectangle**](https://libgdx.badlogicgames.com/nightlies/docs/api/com/badlogic/gdx/math/Rectangle.html) rect)

Merges this rectangle with the other rectangle.

Drawing the I shape

1. Create an l shape
2. Create the Tshape

Drawing the L shape

1. Create an l shape
2. Create an - shape
3. Merge I with – but do we merge precicely. Check your positioning!
4. You now have L
5. Rotate it around
6. Rotate up() to get 7
7. Rotate down to get l-
8. Rotate again to get –l

Drawing the square

1. Make height and width the same
2. No rotations needed

Drawing the

# Drawing the grid we will use to play

If you want the grid in buttons instead.

OrthoGraphicCamera camera = new OrthoGraphicCamera();

camera.setToOrtho(false, yourViewportWidth, yourViewportHeight);

camera.translate(xPos, yPos);

Stage stage = new Stage(your wanted stage width, your wanted stage height, false, batch);

stage.setCamera(camera);

for(int y = 0; y < buttonsOnHeight; y++){

for(int x = 0; x < buttonsOnWidth; x++){

stage.addActor(new TextButton("" + x + y \* buttonsOnWidth, textButtonStyle);

}

}

The render

float buttonWidth = camera.viewportWidth / buttonsOnWidth;

float buttonHeight = camera.viewportHeight / buttonsOnHeight;

for(int y = 0; y < buttonsOnHeight; y++){

for(int x = 0; x < buttonsOnWidth; x++){

TextButton button = stage.getActors().get(x + y \* buttonsOnWidth);

button.setX(x \* buttonWidth);

button.setY(y \* buttonHeight);

button.setWidth(buttonWidth);

button.setHeight(buttonHeight);

}

}

Then draw the stage, note that you should stop any batch that's currently running because stage has it's own batch.begin() and batch.end(). You could start your batch again after stage.draw();

stage.act(delta);

stage.draw();

http://stackoverflow.com/questions/21444209/libgdx-create-grid-for-board-game