

How to Make a GUI Grid in Java

The Grid does nothing special at this stage, but with a little bit of research, you can add action listeners and a bit of logic to make a simple 2D game like tic-tac-toe, or more complicated ones like Battleship. Note: This article uses...

Method 1 of 1:

Steps Code

1. The main class:

```
public class ButtonGrid { public static void main(String[] args) { } }
```

1. Imports:

```
import javax.swing.JFrame; import javax.swing.JButton; import java.awt.GridLayout; public class ButtonGrid { ...
```

1. Constructor Code:

```
public class ButtonGrid { public ButtonGrid(int width, int length){ } } ...
```

1. Frame Code:

```
public class ButtonGrid { JFrame frame=new JFrame(); public ButtonGrid(int width, int length){ frame.setLayout(new GridLayout(width,length)); frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); frame.pack(); frame.setVisible(true); } } ...
```

1. Button Grid Code:

```
|JFrame frame=new JFrame(); //creates frame JButton[][] grid; //names the grid of buttons public ButtonGrid(int width, int length){
```

```
//constructor with 2 parameters frame.setLayout(new GridLayout(width,length
)); //set layout of frame grid=new JButton[width][length];
//allocate the size of grid for(int y=0; ylength; y++){ for(int x=0; xwidth
; x++){ grid[x][y]=new JButton(""+x+","+y+""); frame.add(grid[x][y]);
//adds button to grid } } frame.setDefaultCloseOperation(JFrame.
EXIT_ON_CLOSE); frame.pack(); frame.setVisible(true); } ...
```

1. Adding Buttons to Frame:

```
for(int y=0; ylength; y++){ for(int x=0; xwidth; x++){ grid[x][y]=new
JButton(""+x+","+y+""); frame.add(grid[x][y]); } } ...
```

1. Making a button grid instance:

```
public static void main(String[] args) { new ButtonGrid(3,3);
//makes new ButtonGrid with 2 parameters } ...
```

1. Final Code:

```
import javax.swing.JFrame; //imports JFrame library import
javax.swing.JButton; //imports JButton library import java.awt.GridLayout;
//imports GridLayout library public class ButtonGrid { JFrame frame=new
JFrame(); //creates frame JButton[][] grid; //names the grid of buttons
public ButtonGrid(int width, int length){ //constructor frame.setLayout(new
GridLayout(width,length)); //set layout grid=new JButton[width][length];
//allocate the size of grid for(int y=0; ylength; y++){ for(int x=0; xwidth
; x++){ grid[x][y]=new JButton(""+x+","+y+""); //creates new button
frame.add(grid[x][y]); //adds button to grid } } frame.
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); frame.pack();
//sets appropriate size for frame frame.setVisible(true);
//makes frame visible } public static void main(String[] args) { new
ButtonGrid(3,3); //makes new ButtonGrid with 2 parameters } }
```

```
import javax.swing.JFrame; //imports JFrame library import javax.swing.JButton; //imports JButton library
import java.awt.GridLayout; //imports GridLayout library
```

```
public class ButtonGrid {
```

```
JFrame frame=new JFrame(); //creates frame JButton[][] grid; //names the grid of buttons
```

```
public ButtonGrid(int width, int length){ //constructor frame.setLayout(new GridLayout(width,length)); //set
layout grid=new JButton[width][length]; //allocate the size of grid for(int y=0; y
```

You finished reading the article "**How to Make a GUI Grid in Java**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.

