Task Eight: Guess The Number

Task:

In this task, you will be making a game where the user has to guess a randomly generated number. You must notify the user if he/she has guessed too high, too low, or correctly.

Good luck, and remember to ask if you have any questions.

Task Hints and Requirements:

Make at least a 4x4 grid of buttons (so 16 buttons). You are allowed to look back at your code from the previous task if you forgot how to do this.

Keep track of the number of guesses.

To generate a random number, use Math.random() \* the greatest possible number.

Have a message box pop up telling the user that they have guessed too high or too low. If the user runs out of guesses, or guesses correctly, make the appropriate message pop up, and have the program exit.

You may have noticed that every time you make a new button, you have also been making a new ActionListener. For this task, that might get somewhat difficult. Instead, it is a good idea to connect every button to the same listener.

private class Listener implements ActionListener

{

int listenX;

int listenY;

private Listener(int x, int y)

{

listenX;

listenY;

}

public void actionPerformed(ActionEvent e)

{

}

}

Now you can use listenX and listenY , which will be different for every button, but you can use them in the same listener.

Add a listener to the button (EXAMPLE):

button.addActionListener(new Listener(2, 3));

Organize the buttons by rows and columns. It might be a good idea to keep track of which row and column each row is in.