

```
import java.util.ArrayList;
public class RandomNumberList
{
    private ArrayList<Long> alNumbers = new ArrayList<>();

    public void addSeries(int count, long min, long max)
    {
        Randomizer random = new Randomizer(min, max);
        for (int i = 0; i < count; i++)
            alNumbers.add(random.getNext());
    }

    public void clear()
    {
        alNumbers.clear();
    }

    public int size()
    {
        return alNumbers.size();
    }

    public double getAverage()
    {
        return 0;
    }

    public Long getMin()
    {
        return null;
    }

    public Long getMax()
    {
        return null;
    }

    public Object[] toArray()
    {
        return alNumbers.toArray();
    }
}
```

```
public class Randomizer
{
    // Les limites
    private long min;
    private long max;

    public Randomizer(long pMin, long pMax)
    {
        setLimits(pMin, pMax);
    }

    public void setLimits(long pMin, long pMax)
    {
        // Version avancée
        min = Math.min(pMin, pMax);
        max = Math.max(pMin, pMax);
    }

    public long getNext()
    {
        return (long) (Math.random() * (max - min + 1)) + min;
    }
}
```

```

public class MainFrame extends javax.swing.JFrame
{
    private RandomNumberList numbers = new RandomNumberList();

    public MainFrame()
    {
        initComponents();
        updateView();
    }

    public void updateView()
    {
        numbersList.setListData(numbers.toArray());

        nbrLabel.setText(String.valueOf(numbers.size()));

        double avg;
        avg = numbers.getAverage();

        // Amélioration: ne pas afficher -1 ou null (qui indique une liste vide...)
        if (avg == -1)
            averageLabel.setText("-");
        else
            averageLabel.setText(String.valueOf(avg));

        Long val;

        val = numbers.getMin();
        if (val == null)
            minLabel.setText("-");
        else
            minLabel.setText(String.valueOf(val));

        val = numbers.getMax();
        if (val == null)
            maxLabel.setText("-");
        else
            maxLabel.setText(String.valueOf(val));
    }
    // Skipped: ... initComponents { ... }
    private void addButtonActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_addButtonActionPerformed
        int nbr = Integer.valueOf(numberValuesTextField.getText());
        long min = Long.valueOf(minTextField.getText());
        long max = Long.valueOf(maxTextField.getText());

        numbers.addSeries(nbr, min, max);
        updateView();

        numberValuesTextField.requestFocus();
    } //GEN-LAST:event_addButtonActionPerformed

    private void clearButtonActionPerformed(java.awt.event.ActionEvent evt) { //GEN-FIRST:event_clearButtonActionPerformed
        numbers.clear();
        updateView();
    } //GEN-LAST:event_clearButtonActionPerformed
    // Skipped: ... Look & Feel
    // Variables declaration - do not modify //GEN-BEGIN:variables
    private javax.swing.JButton addButton;
    private javax.swing.JLabel averageLabel;
    private javax.swing.JButton clearButton;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel jLabel5;
    private javax.swing.JLabel jLabel6;
    private javax.swing.JLabel jLabel7;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JLabel maxLabel;
    private javax.swing.JTextField maxTextField;
    private javax.swing.JLabel minLabel;
    private javax.swing.JTextField minTextField;
    private javax.swing.JLabel nbrLabel;
    private javax.swing.JTextField numberValuesTextField;
    private javax.swing.JList numbersList;
    // End of variables declaration //GEN-END:variables
}

```