

```
public class Cistern
{
    private double maximumVolume;
    private double currentVolume;

    public Cistern(double pRadius, double pHeight)
    {
        maximumVolume = Math.PI * pRadius * pRadius * pHeight * 1000; // Attention à la conversion m3 -> litres
        currentVolume = 0;
    }

    public void add(double pVolume)
    {
        if (currentVolume + pVolume <= maximumVolume)
        {
            currentVolume = currentVolume + pVolume;
        }
        else
        {
            currentVolume = maximumVolume;
        }
    }

    public void drain(double pVolume)
    {
        if (pVolume <= currentVolume)
        {
            currentVolume = currentVolume - pVolume;
        }
        else
        {
            currentVolume = 0;
        }
    }

    public boolean addOK(double pVolume)
    {
        if (currentVolume + pVolume <= maximumVolume)
        {
            currentVolume = currentVolume + pVolume;
            return true;
        }
        else
        {
            currentVolume = maximumVolume;
            return false;
        }
    }

    public boolean drainOK(double pVolume)
    {
        if (pVolume <= currentVolume)
        {
            currentVolume = currentVolume - pVolume;
            return true;
        }
        else
        {
            currentVolume = 0;
            return false;
        }
    }

    public double getCurrentVolume()
    {
        return currentVolume;
    }

    public double getCurrentRate()
    {
        return currentVolume / maximumVolume * 100;
    }

    public String toString()
    {
        return "Fill level : " + currentVolume + " l / " + maximumVolume + " l (" + getCurrentRate() + "%)";
    }
}
```

```
public class MainFrame extends javax.swing.JFrame
{
    private Cistern cistern = new Cistern(1, 2);

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

    public void updateView()
    {
        currentVolumeLabel.setText(((int) (cistern.getCurrentVolume() * 100) / 100.0) + " litres");
        currentRateLabel.setText(((int) (cistern.getCurrentRate() * 100) / 100.0) + "%");
        currentRateProgressBar.setValue((int) cistern.getCurrentRate());
    }
// Skipped: ... initComponents { ... }
    private void addButtonActionPerformed(java.awt.event.ActionEvent evt)//GEN-FIRST:event_addButtonActionPerformed
    {//GEN-HEADEREND:event_addButtonActionPerformed
        double v = Double.valueOf(volumeTextField.getText());
        cistern.add(v);
        updateView();
    }//GEN-LAST:event_addButtonActionPerformed

    private void drainButtonActionPerformed(java.awt.event.ActionEvent evt)//GEN-FIRST:event_drainButtonActionPerformed
    {//GEN-HEADEREND:event_drainButtonActionPerformed
        cistern.drain(Double.valueOf(volumeTextField.getText()));
        updateView();
    }//GEN-LAST:event_drainButtonActionPerformed
// Skipped: ... Look & Feel
    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton addButton;
    private javax.swing.JLabel currentRateLabel;
    private javax.swing.JProgressBar currentRateProgressBar;
    private javax.swing.JLabel currentVolumeLabel;
    private javax.swing.JButton drainButton;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JTextField volumeTextField;
    // End of variables declaration//GEN-END:variables
}
```