

```
public class UniversalTime
{
    // localtime: UTC+2
    private int hours;
    private int minutes;

    public UniversalTime(int pHours, int pMinutes)
    {
        hours = pHours;
        minutes = pMinutes;
    }

    public void setToSystemTime()
    {
        Calendar cal = Calendar.getInstance();
        hours = cal.get(Calendar.HOUR_OF_DAY);
        minutes = cal.get(Calendar.MINUTE);
    }

    public int getTimeInMinutes()
    {
        return hours * 60 + minutes;
    }

    private String valueOf(int pHours, int pMinutes)
    {
        String h, m;
        // On veut un format 08:03h
        // et pas 8:3h
        h = String.valueOf(pHours);
        if (pHours < 10)
            h = "0" + h;
        m = String.valueOf(pMinutes);
        if (pMinutes < 10)
            m = "0" + m;

        return h + ":" + m + "h";
    }

    public String getLocalTime()
    {
        return valueOf(hours, minutes);
    }

    public String toString()
    {
        return getLocalTime() + " (" + getTimeInMinutes() + " minutes)";
    }

    public String getUTCTime()
    {
        return valueOf((24 + hours - 2) % 24, minutes);
    }

    public String getNewYorkTime()
    {
        return valueOf((24 + hours - 6) % 24, minutes);
    }

    public String getTallinTime()
    {
        return valueOf((24 + hours + 1) % 24, minutes);
    }

    public String getLosAngelesTime()
    {
        return valueOf((24 + hours - 9) % 24, minutes);
    }

    public String getBangkokTime()
    {
        return valueOf((24 + hours + 5) % 24, minutes);
    }

    public String getMoscowTime()
    {
        return valueOf((24 + hours + 1) % 24, minutes);
    }

    public String getDelhiTime()
    {
        int m = minutes + 30;
        int h = 24 + hours + 3;
        if (m >= 60)
        {
            m = m - 60;
            h++;
        }
        return valueOf(h % 24, m);
    }
}
```

```

public class MainFrame extends javax.swing.JFrame
{
    /**
     * Le temps courant.
     */
    private UniversalTime uTime = null;

    public MainFrame()
    {
        initComponents();
    }
// Skipped: ... initComponents { ... }
    private void setTimeButtonActionPerformed(java.awt.event.ActionEvent evt)//GEN-FIRST:event_setTimeButtonActionPerformed
    { //GEN-HEADEREND:event_setTimeButtonActionPerformed
        int h = Integer.valueOf(hoursTextField.getText());
        int m = Integer.valueOf(minutesTextField.getText());
        uTime = new UniversalTime(h, m);
        localTimeLabel.setText(uTime.toString());
        timeLabel.setText("-");
    } //GEN-LAST:event_setTimeButtonActionPerformed

    private void systemTimeButtonActionPerformed(java.awt.event.ActionEvent evt)//GEN-FIRST:event_systemTimeButtonActionPerformed
    { //GEN-HEADEREND:event_systemTimeButtonActionPerformed
        uTime = new UniversalTime(0,0);
        uTime.setToSystemTime();
        localTimeLabel.setText(uTime.toString());
        timeLabel.setText("-");
    } //GEN-LAST:event_systemTimeButtonActionPerformed

    private void utcButtonActionPerformed(java.awt.event.ActionEvent evt)//GEN-FIRST:event_utcButtonActionPerformed
    { //GEN-HEADEREND:event_utcButtonActionPerformed
        if (uTime != null)
            timeLabel.setText(uTime.getUTCTime());
    } //GEN-LAST:event_utcButtonActionPerformed

    private void nyButtonActionPerformed(java.awt.event.ActionEvent evt)//GEN-FIRST:event_nyButtonActionPerformed
    { //GEN-HEADEREND:event_nyButtonActionPerformed
        if (uTime != null)
            timeLabel.setText(uTime.getNewYorkTime());
    } //GEN-LAST:event_nyButtonActionPerformed

    private void tallinnButtonActionPerformed(java.awt.event.ActionEvent evt)//GEN-FIRST:event_tallinButtonActionPerformed
    { //GEN-HEADEREND:event_tallinButtonActionPerformed
        if (uTime != null)
            timeLabel.setText(uTime.getTallinTime());
    } //GEN-LAST:event_tallinButtonActionPerformed

    private void losAngelesButtonActionPerformed(java.awt.event.ActionEvent evt)//GEN-FIRST:event_losanjelesButtonActionPerformed
    { //GEN-HEADEREND:event_losanjelesButtonActionPerformed
        if (uTime != null)
            timeLabel.setText(uTime.getLosAngelesTime());
    } //GEN-LAST:event_losanjelesButtonActionPerformed

    private void bangkokButtonActionPerformed(java.awt.event.ActionEvent evt)//GEN-FIRST:event_bangkokButtonActionPerformed
    { //GEN-HEADEREND:event_bangkokButtonActionPerformed
        if (uTime != null)
            timeLabel.setText(uTime.getBangkokTime());
    } //GEN-LAST:event_bangkokButtonActionPerformed

    private void moscowButtonActionPerformed(java.awt.event.ActionEvent evt)//GEN-FIRST:event_moscowButtonActionPerformed
    { //GEN-HEADEREND:event_moscowButtonActionPerformed
        if (uTime != null)
            timeLabel.setText(uTime.getMoscowTime());
    } //GEN-LAST:event_moscowButtonActionPerformed

    private void delhiButtonActionPerformed(java.awt.event.ActionEvent evt)//GEN-FIRST:event_delhiButtonActionPerformed
    { //GEN-HEADEREND:event_delhiButtonActionPerformed
        if (uTime != null)
            timeLabel.setText(uTime.getDelhiTime());
    } //GEN-LAST:event_delhiButtonActionPerformed
// Skipped: ... Look & Feel
    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton bangkokButton;
    private javax.swing.JButton delhiButton;
    private javax.swing.JTextField hoursTextField;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel localTimeLabel;
    private javax.swing.JButton losAngelesButton;
    private javax.swing.JTextField minutesTextField;
    private javax.swing.JButton moscowButton;
    private javax.swing.JButton nyButton;
    private javax.swing.JButton setTimeButton;
    private javax.swing.JButton systemTimeButton;
    private javax.swing.JButton tallinnButton;
    private javax.swing.JLabel timeLabel;
    private javax.swing.JButton utcButton;
    // End of variables declaration//GEN-END:variables
}

```