

## TD10 Correction

Matière : DEVELOPPEMENT MOBILE AVANCE

Classes : SEM31

**" JFCameraIp "**

```
public class JFCameraIp extends JFrame {
    private void lancerThreadServeur() {
        Runnable r = new Runnable() {
            @Override
            public void run() {
                demarrerServeur();
            }
        };
        Thread th = new Thread(r);
        th.start();
    }
    private void demarrerServeur() {
        try {
            ss = new ServerSocket(PORT);
            s = ss.accept();
            br = new BufferedReader(new InputStreamReader(s.getInputStream()));
            while (true) {
                String cmd;
                cmd = br.readLine();
                execterCommande(cmd);
            }
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
    private void executerCommande(String commande) {
        String[] t = commande.split(":");
        if (t.length > 1) {
            int type = Integer.parseInt(t[0]);
            int valeur = Integer.parseInt(t[1]);
            switch (type) {
                case 0:    tournerPanoram(valeur);    break;
                case 1:    tournerVertical(valeur);   break;
                case 2:    zoom(valeur);               break;
            }
        }
    }
}
```

## "MainActivity"

```
public class MainActivity extends Activity {
    private void lancerThreadClient() {
        Runnable r = new Runnable() {
            @Override
            public void run() {
                demarrerClient();
            }
        };
        Thread th = new Thread(r);
        th.start();
    }
    protected void demarrerClient() {
        try {
            InetAddress i = (InetAddress) InetAddress.getByName(edAdresse.getText().toString());
            s = new Socket(i, Integer.parseInt(edPort.getText().toString()));
            pw = new PrintWriter(new BufferedWriter(new OutputStreamWriter
                (s.getOutputStream())), true);
        } catch (UnknownHostException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
    private void envoyer(final String cmd) {
        Runnable r = new Runnable() {
            @Override
            public void run() {
                if (pw != null) {
                    pw.println(cmd);
                }
            }
        };
        Thread th = new Thread(r);
        th.start();
    }
    protected void tournerPanoram() {
        envoyer("0:" + seekPanoram.getProgress());
    }
    protected void tournerVertical() {
        envoyer ("1:" + seekVertical.getProgress());
    }
    protected void zoom() {
        envoyer ("2:" + seekZoom.getProgress());
    }
}
```