import java.io.\*;

import java.util.\*;

import java.sql.\*;

public class JavaTerm

{

SimpleSerial m\_SerialPort = null;

int m\_PortIndex;

boolean m\_IsNative = true;

String inputString;

static final String m\_PrefsFileName = new String("JavaTerm.pref");

Connection con;

Statement st;

ResultSet rs;

int ch = 0;

static String rfidData = "";

public static void main(String[] argh){

new JavaTerm();

}

private void initSerialPort() throws IOException{

if (m\_SerialPort != null){

m\_SerialPort.close();

m\_SerialPort = null;

}

if (m\_IsNative){

m\_SerialPort = new SimpleSerialNative(m\_PortIndex);

}

if (!m\_SerialPort.isValid()){

throw (new IOException("Serial port not opened"));

}

}

public JavaTerm(){

int ii;

try{

DataInputStream prefs = new DataInputStream(new FileInputStream(m\_PrefsFileName));

m\_PortIndex = prefs.readInt();

m\_IsNative = prefs.readBoolean();

if (m\_PortIndex < 0 ){

throw new IOException(m\_PrefsFileName + " is corrupt");

}

initSerialPort();

}catch(IOException e){

System.out.println("preferences file 'JavaTerm.pref' not found / didn't open or there was a problem opening serial port. Searching for serial port");

int m=4;

find\_open\_serial\_port:

for (ii = 0; ii < m; ii++){

try{

m\_PortIndex = ii + 1;

initSerialPort();

System.out.println("Opening serial port Comm" + m\_PortIndex);

break find\_open\_serial\_port;

}catch (IOException ee){

if (ii == m - 1){

System.out.println("Couldn't open any serial ports");

System.exit(0);

}

}

}

}

StringTokenizer stringToken = null;

for (;true;){

if (m\_SerialPort != null){

inputString = m\_SerialPort.readString();

try{

if(inputString.trim().intern() == ""){

System.out.println("No data found");

}else {

System.out.println("Value"+inputString.trim());

if(inputString.trim().startsWith("I")){

SendSMS.sendMessageToMobile("Breaking the DOOR...","9941703017");}

}

Thread.sleep(1000);

}catch(Exception ex){ex.printStackTrace();System.out.println(ex); }}}}

**ALERT SMS:**

import com.objectxp.msg.GsmSmsService;

import com.objectxp.msg.MessageException;

import com.objectxp.msg.SmsMessage;

import com.objectxp.msg.SmsService;

import com.objectxp.msg.StatusReportMessage;

import java.io.File;

import java.io.IOException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.\*;

class SendSMS{

static String smsMessage="",mnum;

public static void sendMessageToMobile(String message, String mobile){

smsMessage = "Alert"+message;

System.out.println(smsMessage);

System.out.println(mobile);

SmsService service = new GsmSmsService();

try {

service.init(new File("C:\\Program Files\\Apache Software Foundation\\Tomcat 5.5\\webapps\\BuildingWeb\\WEB-INF\\classes\\jsms.conf"));

String serviceName = service.getServiceName();

System.out.println(serviceName);

SmsMessage msg = new SmsMessage();

msg.setRecipient(mobile);

msg.setMessage(smsMessage);

service.connect();

service.sendMessage(msg);

service.disconnect();

}catch(IOException em){

System.out.println("IO : "+em);

Logger.getLogger(SendSMS.class.getName()).log(Level.SEVERE, null, em);

}catch (MessageException ex){

System.out.println("ME : "+ex);

Logger.getLogger(SendSMS.class.getName()).log(Level.SEVERE, null, ex);

}finally{

service.destroy();

}

}

}

**IDENTIFICATION OF PORT:**

import java.io.\*;

import java.util.\*;

import javax.comm.\*;

public class SimpleWrite

{

static Enumeration portList;

static CommPortIdentifier portId;

static String messageString;

static SerialPort serialPort;

static OutputStream outputStream;

static String data;

public void writeData (String data)

{

this.data = data;

if(data!=null) {System.out.println("Data : "+data);}

portList = CommPortIdentifier.getPortIdentifiers();

while (portList.hasMoreElements())

{

portId = (CommPortIdentifier) portList.nextElement();

if (portId.getPortType() == CommPortIdentifier.PORT\_SERIAL)

{

if (portId.getName().equals("COM1"))

{

try

{

serialPort = (SerialPort)

portId.open("SimpleWriteApp", 2000);

} catch (PortInUseException e) {}

try

{

outputStream = serial

Port.getOutputStream();

} catch (IOException e) {}

try

{

serialPort.setSerialPortParams(9600,

SerialPort.DATABITS\_8,

SerialPort.STOPBITS\_1,

SerialPort.PARITY\_NONE);

} catch (UnsupportedCommOperationException e) {}

try

{

outputStream.write(data.getBytes());

} catch (IOException e) {}

if(serialPort!=null)

{

serialPort.close();

}

}

}

}

}

}

**RUNNING TOMCAT:**

import java.io.\*;

import javax.servlet.\*;

import javax.servlet.http.\*;

import java.sql.\*;

import java.util.\*;

import java.lang.\*;

public class DeviceServlet extends HttpServlet

{

public void doGet(HttpServletRequest request,HttpServletResponse response) throws IOException, ServletException

{

response.setContentType("text/html");

PrintWriter out = response.getWriter();

String message="",value="";

value=request.getParameter("Request");

System.out.println(value);

out.println(value);

new ReadWrite(value);

out.println("<center><hr><h2><b>YOU HAVE SUCCESFULLY CONTROLLED DEVICE </b></h2>");

}

}

**DOOR OPENING:**

import java.io.\*;

import javax.servlet.\*;

import javax.servlet.http.\*;

import java.sql.\*;

import java.util.\*;

import java.lang.\*;

public class LoginServlet extends HttpServlet{

String uname = "" , password = "",message = "";

private Connection con = null;

private Statement stmt = null;

public void init(ServletConfig config){

try{

Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");

con = DriverManager.getConnection("jdbc:odbc:Driver={SQL Server};Database=dooropening;Server=.;UID=sa");

stmt = con.createStatement();

}catch(Exception ex){

ex.printStackTrace();

System.out.println(ex);

}

}

public void doGet(HttpServletRequest request,HttpServletResponse response) throws ServletException{

try{

response.setContentType("text/html");

PrintWriter out = response.getWriter();

String requestString = request.getParameter("Request");

StringTokenizer stringTokenizer = new StringTokenizer(requestString,"$");

stringTokenizer.nextToken();

uname = (String)stringTokenizer.nextToken();

password = (String)stringTokenizer.nextToken();

ResultSet rs = stmt.executeQuery("select \* from userRegisteration where uname='"+uname+"' and pass='"+password+"'");

if(rs.next()){

message = "VALID";

}else{

message = "INVALID";

}

out.println(message);

}catch(Exception ex){

ex.printStackTrace();

System.out.println(ex);

}

}

}