package main.java.com.wfdai.weatherforecastdai.main.KPI;

import com.mysql.jdbc.jdbc2.optional.MysqlDataSource; import java.sql.Connection;
import java.sql.ResultSet; import java.sql.SQLException; import java.sql.Statement;
import java.text.DateFormat; import java.util.Date; import java.text.SimpleDateFormat;
import java.time.LocalDate; import java.time.LocalDateTime; import
java.time.LocalTime; import java.util.ArrayList; import main.java.com.wfdai.weatherforecastdai.main.DataBas

/** * Regista e devolve dados relativos a KPI's @author daniel */ public class
RegistoKPI {

```
DateFormat dateFormat = new SimpleDateFormat("yyyy/MM/dd HH:mm:ss");

protected ArrayList<String> uptime, totalClients, activeClients, messages, subscriptions, re

public RegistoKPI() {
    this.uptime = new ArrayList<>();
    this.totalClients = new ArrayList<>();
    this.activeClients = new ArrayList<>();
    this.messages = new ArrayList<>();
    this.subscriptions = new ArrayList<>();
    this.receivedLoad5 = new ArrayList<>();
    this.receivedLoad15 = new ArrayList<>();
    this.bytesSent15 = new ArrayList<>();
    this.time = new ArrayList<>();
}

/**
 * Coloca KPI's na base de dados
 *
 * @param kpi Objeto KPI
 */
public void putKPI(KPI kpi) {
    try {
        MysqlDataSource dataSource = new MysqlDataSource();
        DataBase database = new DataBase();
        dataSource.setUser(database.getUser());
        dataSource.setPassword(database.getPassword());
        dataSource.setServerName(database.getServerName());
        try (Connection conn = dataSource.getConnection()) {
            Statement st = conn.createStatement();
            st.executeUpdate("INSERT INTO mydb.KPI (`uptime`,totalClients, activeClients, me
                    + "VALUES ('" + kpi.getUptime() + "', '" + kpi.getTotalClients() + "', '
                    + kpi.getMessages() + "', '" + kpi.getSubscriptions() + "', '" + kpi.get
                    + kpi.getBytesSent15() + "')");
        }
    } catch (SQLException e) {
```

```
        System.err.println("Got an exception! ");
        System.err.println(e.getMessage());
    }
}

/**
 * Obtem o Registo dos KPI's da base de dados e atribui-os a atributos do objeto do tipo Reg
 *
 */
public void getRegistoKPI() {
    uptime.clear();
    totalClients.clear();
    activeClients.clear();
    messages.clear();
    subscriptions.clear();
    receivedLoad5.clear();
    receivedLoad15.clear();
    bytesSent15.clear();
    time.clear();

    try {
        MysqlDataSource dataSource = new MysqlDataSource();
        DataBase database = new DataBase();
        dataSource.setUser(database.getUser());
        dataSource.setPassword(database.getPassword());
        dataSource.setServerName(database.getServerName());
        try (Connection conn = dataSource.getConnection()) {
            Statement st = conn.createStatement();
            ResultSet rs = st.executeQuery("Select * from mydb.KPI");

            while (rs.next()) {
                uptime.add("\"" + rs.getString("uptime") + "\"");
                totalClients.add(rs.getString("totalClients"));
                activeClients.add(rs.getString("activeClients"));
                messages.add(rs.getString("messages"));
                subscriptions.add(rs.getString("subscriptions"));
                receivedLoad5.add(rs.getString("receivedLoad5"));
                receivedLoad15.add(rs.getString("receivedLoad15"));
                bytesSent15.add(rs.getString("bytesSent15"));

                LocalDate datePart = LocalDate.parse(rs.getDate("time").toString());
                LocalTime timePart = LocalTime.parse(rs.getTime("time").toString());
                LocalDateTime dt = LocalDateTime.of(datePart, timePart);
                Date data = java.sql.Timestamp.valueOf(dt);
                time.add("\"" + data.toString() + "\"");
                //time.add(rs.getDate("time"));
```

```
            }
        }
    } catch (SQLException e) {
        System.err.println("Got an exception! ");
        System.err.println(e.getMessage());
    }
}

public ArrayList<String> getUptime() {
    return uptime;
}

public ArrayList<String> getTotalClients() {
    return totalClients;
}

public ArrayList<String> getActiveClients() {
    return activeClients;
}

public ArrayList<String> getMessages() {
    return messages;
}

public ArrayList<String> getSubscriptions() {
    return subscriptions;
}

public ArrayList<String> getReceivedLoad5() {
    return receivedLoad5;
}

public ArrayList<String> getReceivedLoad15() {
    return receivedLoad15;
}

public ArrayList<String> getBytesSent15() {
    return bytesSent15;
}

public ArrayList<String> getTime() {
    return time;
}

}
```