Como configurar um alarme em Java ME

O MIDlet abaixo é um simples despertador. Este objetiva mostrar um *DateField* e possibilitará a adição de dia e hora. Uma vez que o dia/hora estão acertados, a aplicação soará um alarme e mostrará uma caixa de dialógo.

```
import java.util.*;
import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;
import java.util.Timer;
import java.util.TimerTask;



public class Alarm extends MIDlet implements ItemStateListener,
        CommandListener {
    private Display display; // Referência a um objeto Display
    private Form fmMain; // O componente Form principal
    private Command cmAlarm; // Comando para inicializar o Timer
    private Command cmReset; // Reinicializa a data/hora corrente
    private Command cmExit; // Finaliza a MIDlet
    private DateField dfAlarmTime; // Intervalo de tempo
    private int dateIndex; // Index para o componente DateField no componente Form
    private Date currentTime; // Hora atual...muda ao pressionar reset
    private Timer tm; // Objeto timer - mantém a hora do sistema
    private AlarmTimer tt; // Objeto TimerTask
    private boolean dateOK = false; // A entrada do usuário foi válida?

    public Alarm() {
        display = Display.getDisplay(this);

// O Form (LCDUI) principal
        fmMain = new Form("When to sound the alarm:");

// Salva o dia atual
        currentTime = new Date();

// compoentne DateField com a data atual como valor default
        dfAlarmTime = new DateField("", DateField.DATE_TIME);
        dfAlarmTime.setDate(currentTime);

// Todos os comandos
        cmAlarm = new Command("Sleep", Command.SCREEN, 1);
        cmReset = new Command("Reset", Command.SCREEN, 1);
        cmExit = new Command("Exit", Command.EXIT, 1);

// Adiciona comandos ao componente Form e configura listeners
        dateIndex = fmMain.append(dfAlarmTime);
        fmMain.addCommand(cmAlarm);
        fmMain.addCommand(cmReset);
        fmMain.addCommand(cmExit);
        fmMain.setCommandListener(this);
        fmMain.setItemStateListener(this);
    }

    public void startApp() {
```


(C) Copyright Nokia 2014. All rights reserved.

```
        display.setCurrent(fmMain);
    }

    public void pauseApp() {
    }

    public void destroyApp(boolean unconditional) {
    }

    public void itemStateChanged(Item item) {
        if (item == dfAlarmTime) {
// Se o usuário selecionou a data ou a hora que é anterior a atual ("hoje"),
// cria-se uma flag. Utiliza-se o método getTime()
// da classe Date, o qual retorna um valor em milissegundos desde 1 de Janeiro de 1970
            if (dfAlarmTime.getDate().getTime() < currentTime.getTime())
                dateOK = false;
            else
                dateOK = true;
        }
    }

    public void commandAction(Command c, Displayable s) {
        if (c == cmAlarm) {
            if (dateOK == false) {
                Alert al = new Alert("Unable to set alarm",
                        "Please choose another date and time.", null, null);
                al.setTimeout(Alert.FOREVER);
                al.setType(AlertType.ERROR);
                display.setCurrent(al);
            } else {
// Cria um objeto Timer
                tm = new Timer();
                tt = new AlarmTimer();

// Quantidade de "tempo" restante
                long amount = dfAlarmTime.getDate().getTime() -
                        currentTime.getTime();
                tm.schedule(tt,amount);

// Remove objetos de comandos
                fmMain.removeCommand(cmAlarm);
                fmMain.removeCommand(cmReset);

// Remove o componente DateField
                fmMain.delete(dateIndex);

// Altera a mensagem do componente Form
                fmMain.setTitle("Sleeping...");
            }
        } else if (c == cmReset) {
// Reseta para o dia/hora atual
            dfAlarmTime.setDate(currentTime = new Date());
        } else if (c == cmExit) {
            destroyApp(false);
            notifyDestroyed();
        }
    }
```


(C) Copyright Nokia 2014. All rights reserved.

```
// Suporte a "timer task"
    private class AlarmTimer extends TimerTask {
        public final void run() {
            Alert al = new Alert("Time to wake up!");
            al.setTimeout(Alert.FOREVER);
            al.setType(AlertType.ALARM);
            AlertType.ERROR.playSound(display);
            display.setCurrent(al);

// Cancela o "timer task"
            cancel();
        }
    }
}
```


(C) Copyright Nokia 2014. All rights reserved.