Converting time zones in Java ME

This code snippet demonstrates how to convert time zones.

Overview

A list of all supported time zones is shown on the screen. By choosing a time zone from the list, the user can see the time offset it has. Method `TimeZone.getTimeZone()` allows the application to retrieve the chosen Time Zone and method `TimeZone.getRawOffset()` is used to retrieve the raw time offset for the chosen time zone.

Source file: ConvertingTimeZones.java

```
import java.util.TimeZone;
import javax.microedition.lcdui.Command;
import javax.microedition.lcdui.CommandListener;
import javax.microedition.lcdui.Display;
import javax.microedition.lcdui.Displayable;
import javax.microedition.lcdui.Form;
import javax.microedition.lcdui.List;
import javax.microedition.midlet.MIDlet;

public class ConvertingTimeZones extends MIDlet implements CommandListener {

    private List list;
    private Form form;
    private Display display;
    private TimeZone timeZone;
    private Command backCommand;
    private Command exitCommand;

    /**
     * Constructor. Constructs the object and initializes buffer.
     */
    public ConvertingTimeZones() {
        timeZone = TimeZone.getDefault();
        backCommand = new Command("Back", Command.BACK, 1);
        exitCommand = new Command("Exit", Command.EXIT, 0);

        form = new Form("Time zone info");
        form.addCommand(backCommand);
        form.setCommandListener(this);

        //Create list filled with all available time zone IDs.
        list = new List("Select time zone:", List.IMPLICIT,
                        timeZone.getAvailableIDs(), null);
        list.addCommand(exitCommand);
        list.setCommandListener(this);

        display = Display.getDisplay(this);
        display.setCurrent(list);
    }

    /**
     * From CommandListener.
     * Called by the system to indicate that a command has been invoked on a
     * particular displayable.
     * @param cmd the command that was invoked
     * @param displayable the displayable where the command was invoked
     */
    public void commandAction(Command cmd, Displayable displayable) {
        if (cmd == List.SELECT_COMMAND) {
            //Clear form.
            form.deleteAll();
            //Get selected time zone index.
            String timeZoneID = list.getString(list.getSelectedIndex());
            form.append("Time zone ID: " + timeZoneID + "\n");
            form.append("Time offset: ");
            //Display the GMT offset for selected time zone in hours.
            form.append(String.valueOf(
                   timeZone.getTimeZone(timeZoneID).getRawOffset() / 3600000));
            form.append(" hours");

            display.setCurrent(form);
        } else if (cmd == backCommand) {
            display.setCurrent(list);
        } else if (cmd == exitCommand) {
            notifyDestroyed();
        }
    }

    /**
     * From MIDlet.
     * Called when the MIDlet is started.
     */
    public void startApp() {
        // No implementation required.
    }

    /**
     * From MIDlet.
     * Called to signal the MIDlet to enter the Paused state.
     */
    public void pauseApp() {
        // No implementation required.
    }

    /**
     * From MIDlet.
     * Called to signal the MIDlet to terminate.
     * @param unconditional whether the MIDlet has to be unconditionally
     * terminated
```


(C) Copyright Nokia 2013. All rights reserved.

```
     */
    public void destroyApp(boolean unconditional) {
    }
}
```

Postconditions

This application allows the user to retrieve the time offset for any time zone he or she needs.

Supplementary material

The source file and executable application are available for download at Media:Converting time zones in J2ME.zip.

(C) Copyright Nokia 2013. All rights reserved.

(C) Copyright Nokia 2013. All rights reserved.