Finding audio and video formats supported by the phone in Java ME

This code snippet demonstrates how to find audio and video formats supported by the phone.

Overview

The `Manager.getSupportedContentTypes(null)` method returns all supported formats as string array, with items containing their media type in the beginning.

This code snippet can be implemented without using MMAPI, by parsing the "audio.encodings" and "video.encodings" system properties.

Source file: GetSupportedFormats.java

```
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.media.Manager;
import javax.microedition.midlet.MIDlet;


public class GetSupportedFormats extends MIDlet implements CommandListener{

    private Display display;
    private Form form;

    private Command exitCommand;

    /**
     * Constructor. Constructs the object and initializes displayables.
     */
    public GetSupportedFormats() {
        form = new Form("Supported formats");
        exitCommand = new Command("Exit", Command.EXIT, 2);

        form.addCommand(exitCommand);
        form.setCommandListener(this);

        display = Display.getDisplay(this);
        display.setCurrent(form);

        form.append("Supported audio formats:\n");
        displaySupportedFormats("audio");
        form.append("Supported video formats:\n");
        displaySupportedFormats("video");

    }

    /**
     * Displays supported formats.
     * @param format media format, which need to be displayed.
     */
    private void displaySupportedFormats(String format) {

        String[] s = Manager.getSupportedContentTypes(null);
        for (int i = 0; i<s.length; i++ ) {
            if ( s[i].indexOf(format, 0) != -1 ) {
                form.append(s[i] + "\n");
            }
        }

    }

    /**
     * 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
     */
    public void destroyApp(boolean unconditional) {
        // No implementation required
    }

    /**
     * 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
     */
```

 (C) Copyright Nokia 2013. All rights reserved.

```
    public void commandAction(Command c, Displayable d) {
        if (c == exitCommand) {
            notifyDestroyed();
        }
    }

}
```

Postconditions

After the MIDlet starts, the user will see all the audio and video formats supported by the device.

Supplementary material

You can view the source file and executable application in the attached zip archive. The archive is available for download at Media:Finding audio and video formats supported by the phone in J2ME.zip.

 (C) Copyright Nokia 2013. All rights reserved.

 (C) Copyright Nokia 2013. All rights reserved.