

Original em: [How to retrieve version number of Java Runtime for S60](#)

Contents

- [1 Introdução](#)
- [2 Código fonte: VersionMIDlet.java](#)
- [3 Código fonte: PlatformDetails.java](#)
- [4 Aplicação de exemplo](#)
- [5 Veja também](#)

Introdução

Os dispositivos S60 da Nokia vêm equipados com uma implementação da plataforma JavaME (Java Platform Micro Edition), por isto é chamada "Java Runtime" x.y para S60, ou abreviando JRT x.y . Começando da S60 3ª Ed. FP2 as implementações Java são divididas em versões separadamente da subjacente plataforma S60. Por exemplo, Java Runtime 1.3 for S60 (JRT 1.3) é a atual versão de lançamento com o Nokia XpressMusic (software 20.0.012 em diante). A versão exata pelo ambiente Java é 1.3.4. Haverá de ser disponibilizado nas páginas da Wiki do Fórum Nokia um documento "release note", que explica em detalhes as características da nomeação de versões. Por favor, leia a seção abaixo "veja também" para mais informações. Atualmente a versão do JRT pode ser checada a propriedade do sistema (system property) MIDP padrão "microedition.platform". Por exemplo, nos dispositivos Nokia 5800 XpressMusic S60 5ª edição este propriedade do sistema retorna:

```
NokiaN5800d-1/20.0.012/sw_platform=S60;sw_platform_version=5.0;java_build_version=1.3.4
```

Aqui abaixo está um exemplo de código MIDlet totalmente funcional. O VersionMIDlet ler as propriedades do sistema e passa os detalhes (dispositivo, versão do software, plataforma e versão da JRT) e imprimi-a na tela.

Código fonte: VersionMIDlet.java

```
import javax.microedition.lcdui.Command;
import javax.microedition.lcdui.CommandListener;
import javax.microedition.lcdui.Displayable;
import javax.microedition.lcdui.Display;
import javax.microedition.lcdui.Form;
import javax.microedition.midlet.MIDlet;

/**
 * <p>Title: VersionMIDlet</p>
 * <p>Descrição: Esta MIDlet mostra a importância da propriedade do sistema "microedition.platform"
 * e também dos detalhes dos valores contíguas. Este utiliza os métodos da classe PlatformDetails
 * para obter as informações.</p>
 */

public class VersionMIDlet extends MIDlet implements CommandListener {
    private Form form;
    private Command exitCommand;
```

Como obter o número da versão da Java Runtime para S60

```
private PlatformDetails details;
private String device = "";
private String sw = "";
private String platform = "";
private String platform_version = "";
private String java_version = "";

public void startApp() {
    form = new Form("VersionMIDlet");
    exitCommand = new Command("Exit", Command.EXIT, 1);
    form.addCommand(exitCommand);
    form.setCommandListener(this);
    details = new PlatformDetails();
    device = details.getDeviceName();
    sw = details.getSwVersion();
    platform = details.getPlatform();
    platform_version = details.getPlatformVersion();
    java_version = details.getJavaVersion();
    form.append("microedition.platform = " + details.platformString + "\n\n");
    form.append("Device: " + device + "\n");
    form.append("Software version: " + sw + "\n");
    form.append("Platform: " + platform + "\n");
    form.append("Platform version: " + platform_version + "\n");
    form.append("Java Runtime for S60: " + java_version + "\n");
    Display.getDisplay(this).setCurrent(form);
}

public void pauseApp() {
}

public void destroyApp(boolean unconditional) {
}

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

Código fonte: PlatformDetails.java

```
/**
 * <p>Título: PlatformDetails</p>
 * <p>Descrição: Esta classe contém os métodos para obter os detalhes do valor da propriedade
 * "microedition.platform". Nos dispositivos S60 3a edição FP2 (e os mais novos)
 * existem informações sobre o nome do dispositivo, versão do software, plataforma e sua versão,
 * e a versão do Java.</p>
 * <p> Por exemplo, no dispositivo N85 S60 3.2, uma requisição a propriedade do sistema retornari
 * NokiaN5800d-1/20.0.012/sw_platform=S60;sw_platform_version=5.0;java_build_version=1.3.4</p>
 * <p>Nos dispositivos S60 mais antigos (e Serie 40) o valor retornado é normalmente
 * como segue (disponibilizando o modelo do dispositivo e a versão do sw):
 * NokiaE71-1/0.07.76</p>
 * <p>Neste caso, os valores retornados para outros detalhes são
 * "N/A".</p>
 */
public class PlatformDetails {
    protected String platformString = System.getProperty("microedition.platform");
    private int length = platformString.length();

    public String getDeviceName() {
        String device = "";
    }
}
```

Como obter o número da versão da Java Runtime para S60

```
String string = "";
int index = 0;
char c = 'a';
while (device.equals("")) {
    c = platformString.charAt(index);
    if (c != '/') {
        string += c;
        index++;
    }
    else device = string;
}
return device;
}

public String getSwVersion() {
    String sw = "";
    String string = "";
    int index = 0;
    char c = 'a';
    while (true) {
        c = platformString.charAt(index);
        if (c != '/') index++;
        else break;
    }
    index++;
    c = 'a';
    while (c != '/') {
        c = platformString.charAt(index);
        string += c;
        if (index < length) index++;
        if (index == length) {
            sw = string;
            break;
        }
    }
    if (sw.equals("")) sw = string.substring(0, string.length()-1);
    return sw;
}

public String getPlatform() {
    String platform = "";
    String sw_platform = "sw_platform";
    String test = "";
    String s60 = "";
    int sw_length = sw_platform.length();
    int index = 0;
    while (platform.equals("")) {
        test = platformString.substring(index, index + sw_length);
        if(test.equalsIgnoreCase(sw_platform)) {
            index = index + sw_length + 1;
            s60 = platformString.substring(index, index + 3);
            if (s60.equalsIgnoreCase("s60")) platform = s60;
            else platform = "N/A";
        }
        if (index < (length-sw_length)) index++;
        else platform = "N/A";
    }
    return platform;
}

public String getPlatformVersion() {
    String platformVersion = "";
```

Como obter o número da versão da Java Runtime para S60

```
int index = platformString.length()-1;
char c = 'a';
char temp_c = 'a';
while (platformVersion.equals("")) {
    c = platformString.charAt(index);
    if (c == ';') {
        String temp = "";
        int temp_i = index-1;
        while (temp.equals("")) {
            temp_c = platformString.charAt(temp_i);
            if (temp_c == '=') temp = platformString.substring(temp_i + 1, index);
            if (temp_i > 0) temp_i--;
        }
        platformVersion = temp;
    }
    if (index > 0) index--;
    else platformVersion = "N/A";
}
return platformVersion;
}

public String getJavaVersion() {
    String version = "";
    String java = "java";
    String test = "";
    int index = length;
    while (version.equals("")) {
        test = platformString.substring(index - 4, index);
        if(test.equalsIgnoreCase(java)) {
            int c_index = 0;
            int temp = 0;
            char c;
            for (int i = 1; i<12; i++) { // the version format is xxx.y.zzzzz
                temp = length - i;
                c = platformString.charAt(temp);
                if (c == '=') c_index = temp;
            }
            version = platformString.substring(c_index + 1);
        }
        if (index > 4) index--;
        else version = "N/A";
    }
    return version;
}
}
```

Há também os arquivos VersionMIDlet.jad e VersionMIDlet.jar disponíveis aqui.

Aplicação de exemplo

- [VersionMIDlet.zip](#) contendo o VersionMIDlet.jad e o VersionMIDlet.jar

Veja também

- [Nomeação de versões da "Java runtime" para S60](#)