



Home Java Frameworks Database Technology Web Development Build/Test Tools Servers PHP

Home J2me Java Platform Micro Edition

Questions: Ask Latest



Java Platform Micro Edition

Share on Google+

Advertisement

Java ME was designed by Sun Microsystems and is a replacement for a similar technology, Personal Java. Originally developed under the Java Community Process as JSR 68, the different flavors of Java ME have evolved in separate JSRs.

Java Platform Micro Edition

<—Prior Index Next—>

Java ME was designed by Sun Microsystems and is a replacement for a

similar technology, Personal Java. Originally developed under the Java Community Process as JSR 68, the different flavors of Java ME have evolved in separate JSRs. Sun provides a reference implementation of the specification, but has tended not to provide free binary implementations of its Java ME runtime environment for mobile devices, rather relying on third parties to provide their own. As of 22 December 2006, the Java ME source code is licensed under the GNU General Public License, and is released under the project name phone.

Java ME has become a popular option for creating games for cell phones, as they can be emulated on a PC during the development stage and easily uploaded to the phone. This contrasts with the difficulty of developing, testing, and loading games for other special gaming platforms such as those made by Nintendo, Sony, Microsoft, and others, as expensive system-specific hardware and kits are required.

Usage of Java ME

Java ME includes flexible user interfaces, robust security, built-in network protocols, and support for networked and offline applications that can be downloaded dynamically. Applications based on Java ME are portable across many devices; yet leverage each device?s native capabilities.

Java ME device as implement a profile, the most common of these are the Mobile Information Device Profile aimed at mobile devices, such as cell phones, and the Personal Profile aimed at consumer products and embedded devices like Set-top boxes and PDAs.

Developing with Java ME

Writing a Java ME application uses the same basics programming constructs as used with Java SE applications. Basically there are two types of configurations involved in Java ME application development, which are:

CLDC (Connected Limited Device Configuration)

CDC (Connected Device Configuration)

Architecture The Java ME Architecture comprises of three software layers:

The first layer is the configuration layer that includes the JVM, which directly interacts with the native OS. The Configuration layer also handles the interaction between the profile and the JVM.

Google Custom Search Search

Related Latest Frameworks Category

J2ME Tutorial
Access URL
A MIDlet Example to execute Alert Message with an Image
Appending Image into the J2ME Form
Appending string in J2ME Form
Arc MIDlet Example
Audio MIDlet Example
Text Example in J2ME
J2ME Calendar Application
J2ME Draw Triangle, Rectangle, Arc, Line Round Rectangle Example
>>>More Tutorials in this section...



The second layer is the profile layer, which consists of the minimum set of application programming interface (API) for the small devices.

The third layer is the Mobile Information Device profile (MIDP) layer. The MIDP layer contains java APIs for user network connections, persistence storage, and the user interface. It also has access to CLDC libraries and MIDP libraries.

The Java ME Application Development

I. System Requirements - Hardware

Minimum hardware requirements are:

- 100 MB hard disk space
- 128 MB system RAM
- 800 MHz Pentium III CPU

II. Minimal Software Requirement

- IDE ? Sun ONE Studio 4, Mobile Edition, (formerly Forte for Java)
- GUI ? Sun Java ME Wireless Toolkit 2.5.1 (WTK 2.5.1) for CLDC

For Windows: Download the Sun Java Wireless Toolkit for CLDC from <http://java.sun.com/products/sjwtoolkit/download.html> Ensure that you have installed an appropriate Java SE environment. Run the installer, `sun_java_wireless_toolkit-2_5_1-windows.exe`. Follow the instructions provided by the installer.

Download the Sun Java Wireless Toolkit for CLDC from <http://java.sun.com/products/sjwtoolkit/download.html> Ensure that you have installed an appropriate

Java SE environment

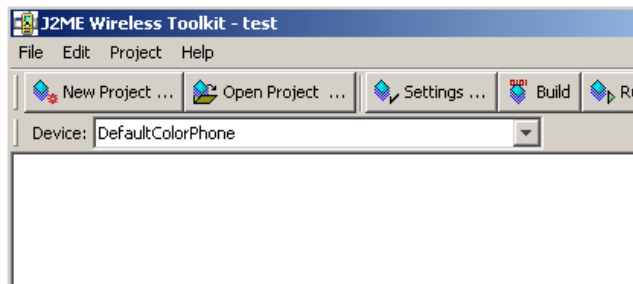
Run the installer, `sun_java_wireless_toolkit-2_5_1-linux.exe`. Follow the instructions provided by the installer.

III. install Sun Java ME Wireless Toolkit 2.5.1 (WTK 2.5.1) on the Windows platform.

- Download the installer file i.e. `netbeans_mobility-5_5_1-win.exe`
- Double Click the icon of downloaded exe.

Now we are ready to create an application with Java Platform ME. Lets create a new project with the following steps:

Step 1: Go to Windows start panel and choose" Wireless Toolkit 2.5.1? as: Start > Programs > Sun Java Wireless Toolkit 2.5.1 for CLDC > Wireless Toolkit 2.5.1. The console window appears like this.



Step 2: Now, Click the ?New Project? on the toolkit menu bar, then a new project box opens. Fill the appropriate Project name and MIDlet class Name of your choice. After that, click Create Project button.

**I Make
\$260
Every Day!**

**work from home
and make \$7,397
a Month...**

See How...

I Recomm...

Core Java V...
Horstmann C ...
INR 585.00

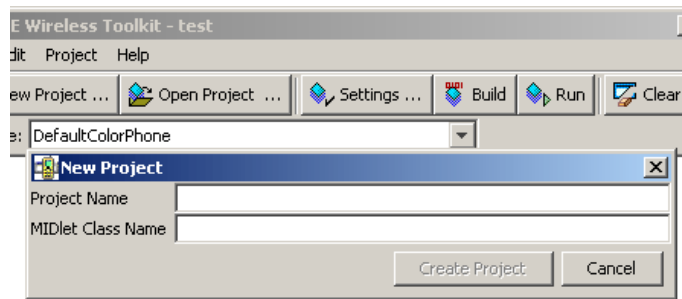
Java: The G...
O'Reilly Media

JAVA Progr...
Ralph Bravaco,...
INR 495.00

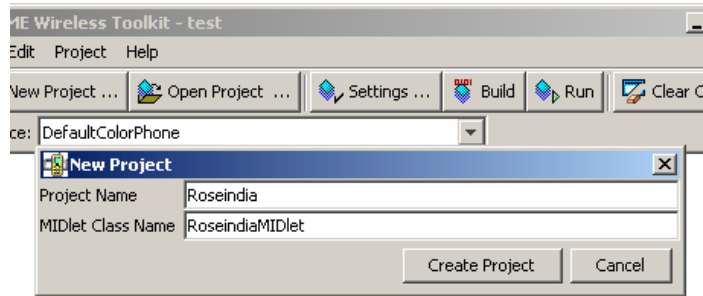
Java Books

PROGRAMM ...
E. Balagurusam...
INR 304.00

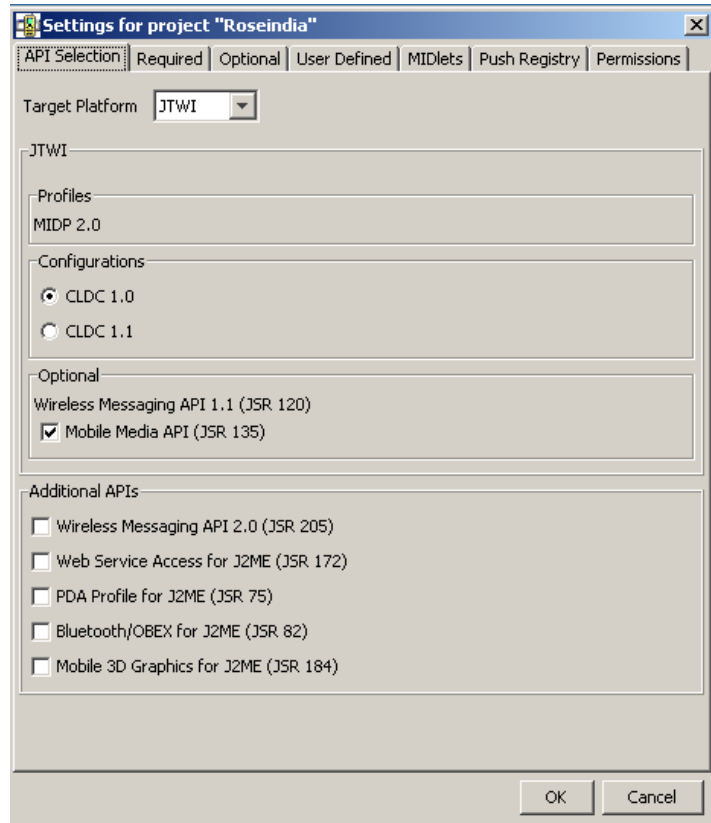
amazon.in 1 2 3 4 > Privacy



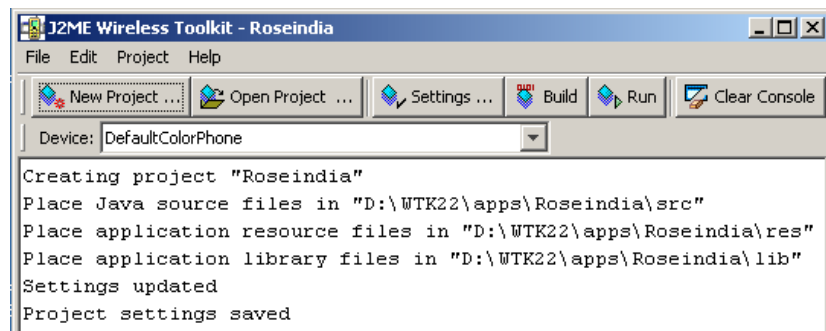
Step 3: Now, Click the ?New Project? on the toolkit menu bar, then a new project box opens. Fill the appropriate Project name (Roseindia)and MIDlet class Name(RoseindiaMIDlet) of your choice. After that, click Create Project button.



Step 4: Then a ?Settings for project? window appears. For default settings, click OK. It is



Step 5: Next appears a window indicating the updated project settings saved in the Console.



Here is the code of this program:

```
import javax.microedition.lcdui.*;
import javax.microedition.midlet.MIDlet;

public class HelloWorld extends MIDlet implements CommandListener {

    public void startApp() {
        Display display = Display.getDisplay(this);

        Form mainForm = new Form("HelloWorld");
        mainForm.append("Welcome to the world of Mobile");

        Command exitCommand = new Command("Exit", Command.EXIT, 0);
        mainForm.addCommand(exitCommand);
        mainForm.setCommandListener(this);

        display.setCurrent(mainForm);
    }

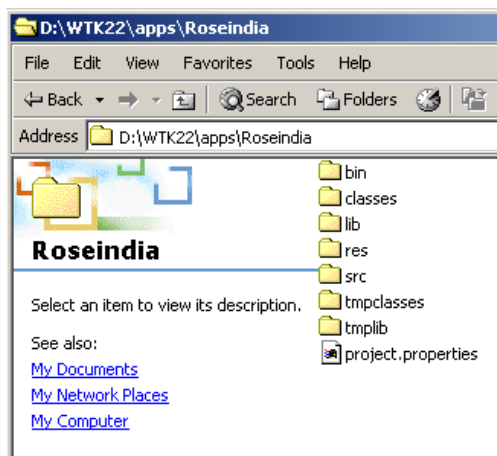
    public void pauseApp () {}

    public void destroyApp(boolean unconditional) {}

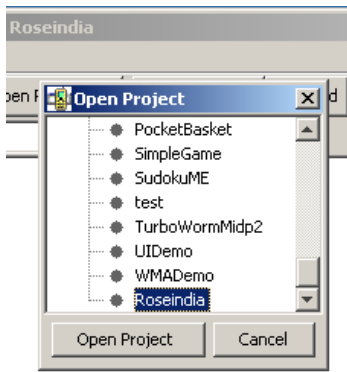
    public void commandAction(Command c, Displayable s) {
        if (c.getCommandType() == Command.EXIT)
            notifyDestroyed();
    }
}
```

[Download source code.](#)

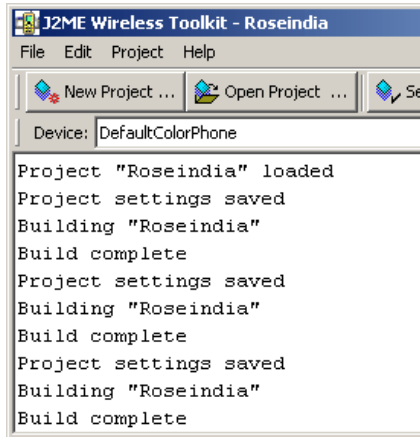
Step 6: Now we need to develop a simplest "Hello World" program in the directory structure in src folder.



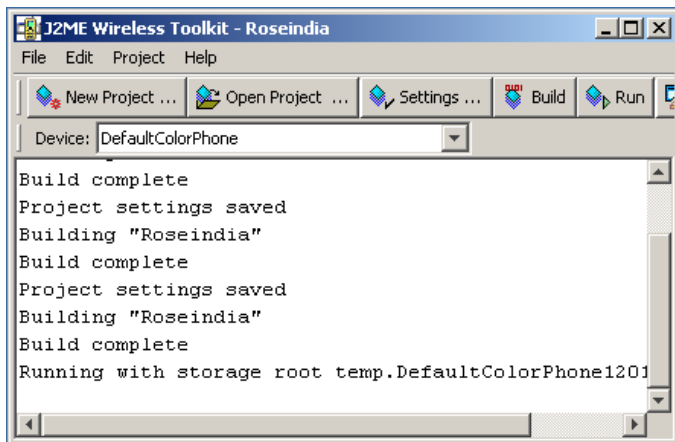
Step 7: Now, Click the ?Open Project? on the toolkit menu bar,



Step 8: Next click the ?Build? button from the toolkit menu bar. This causes the Sun Java Wireless Toolkit for CLDC to compile and preverify the Java source files. The whole build process is shown below.



Step 9: Next click the ?Run? button from the toolkit menu bar. This executes the compiled Java class files on the emulator.

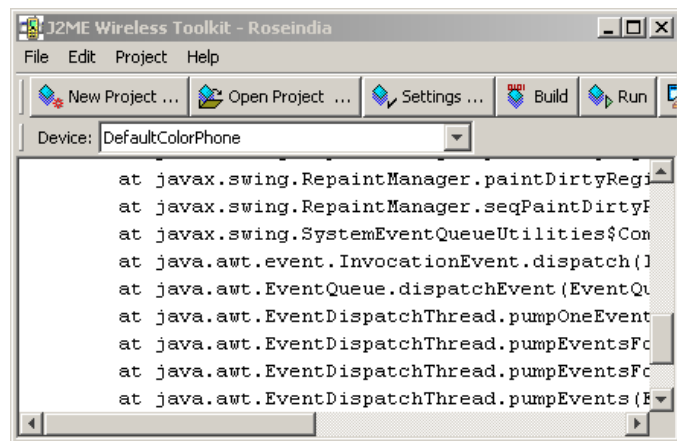


Step 10: Output of project Roseindia at emulator is lick this

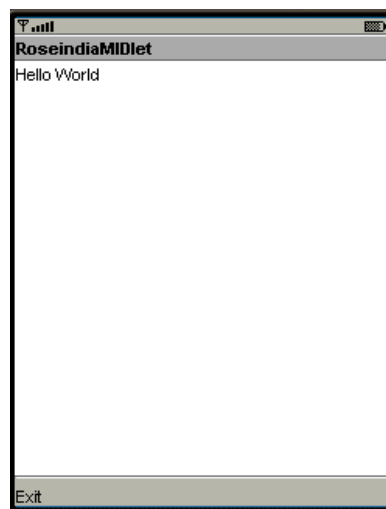


Step 11: Execution of the compiled Java class files on the emulator gives the following customized

output. This output window have



Output: Hello World" program



[Download source code.](#)

Advertisement

[«Previous](#)

[Index](#)

[Next»](#)


Follow us on [Twitter](#), or add us on [Facebook](#) or [Google Plus](#) to keep you updated with the recent trends of Java and other open source platforms. [Connect Me on Google+](#)

We're on  [Follow](#)

  +741

Recommend the
tutorial



 6,314 people like
this. Be the first
of your friends.

 +1 { 1

[Ask Questions?](#) [Discuss: Java Plateform Micro Edition](#) [View All Comments](#)

Post your Comment

Your Name (*) :

Your Email :

Subject (*) :

Your Comment (*) :



SUBMIT

COMMENTS



siva

June 6, 2011

hwo to upload source code into mobile?

hai... This tutorial is very useful to me. what ever you given in this example. i follwed those instruction iam getting the output. Now my question How to run this program in mobile. please replay me...



suneesh

June 20, 2011

j2mee - how to connect pc and mobile

am doing a project to send commands from the mobile and control the pc eg. when u send a command like shutdown the pc has to get shutdown.....could u pls help me how to send data from mobile to pc and control the pc...



sajjad

June 25, 2011

problem

in step 11 when i press the launch button a message shown in the screen that is "java.lang.class not found" i can run any java program. what the solution. i

install sun_java_wireless_toolkit-2.5.2_01-win.



Dinesh

October 9, 2011

j2me

I am able to understand j2me programming but it's hard for me theory of it. I had read lots of text to understand MIDP, CLDC, CDC etc. How can I understand it!



mannika

November 10, 2011

Where to write this code?

hey! i generally use eclipse or netbeans but where have you written this code?



kakada

March 28, 2012

Nice

Thank you very much, i get many thing about it

Services

Software Solutions

JSF Development
Outsourcing
ERP
M-Commerce
Flex Development

Website Development

Web Designing
Web Redesigning
Web Development
Logo Design
Web Design Packages
Domain Registration

Web Promotion Services

SEO Services
Search Eng. Optimization
Search Eng. Submission
SEO Tips
SEO Portfolio
Web Promotion Plans

Content Development

Content Development
Article Writing
Blog Writing
News Writing
SEO Copywriting
Technical Documentation
Article Marketing

Web Hosting Services

ASP.NET Hosting
Unix Hosting
E-Commerce Hosting
Windows Hosting
Hosting Plan

E-Commerce Solutions

CRM

Home **Privacy Policy**

© All Rights are Reserved for Rose India