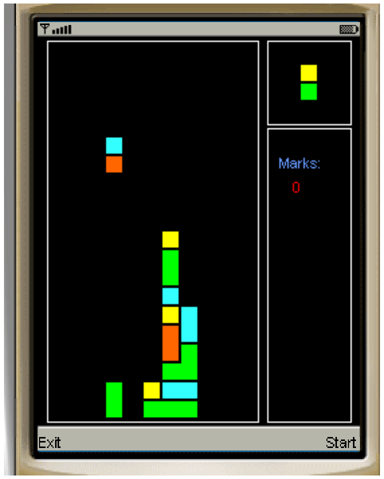
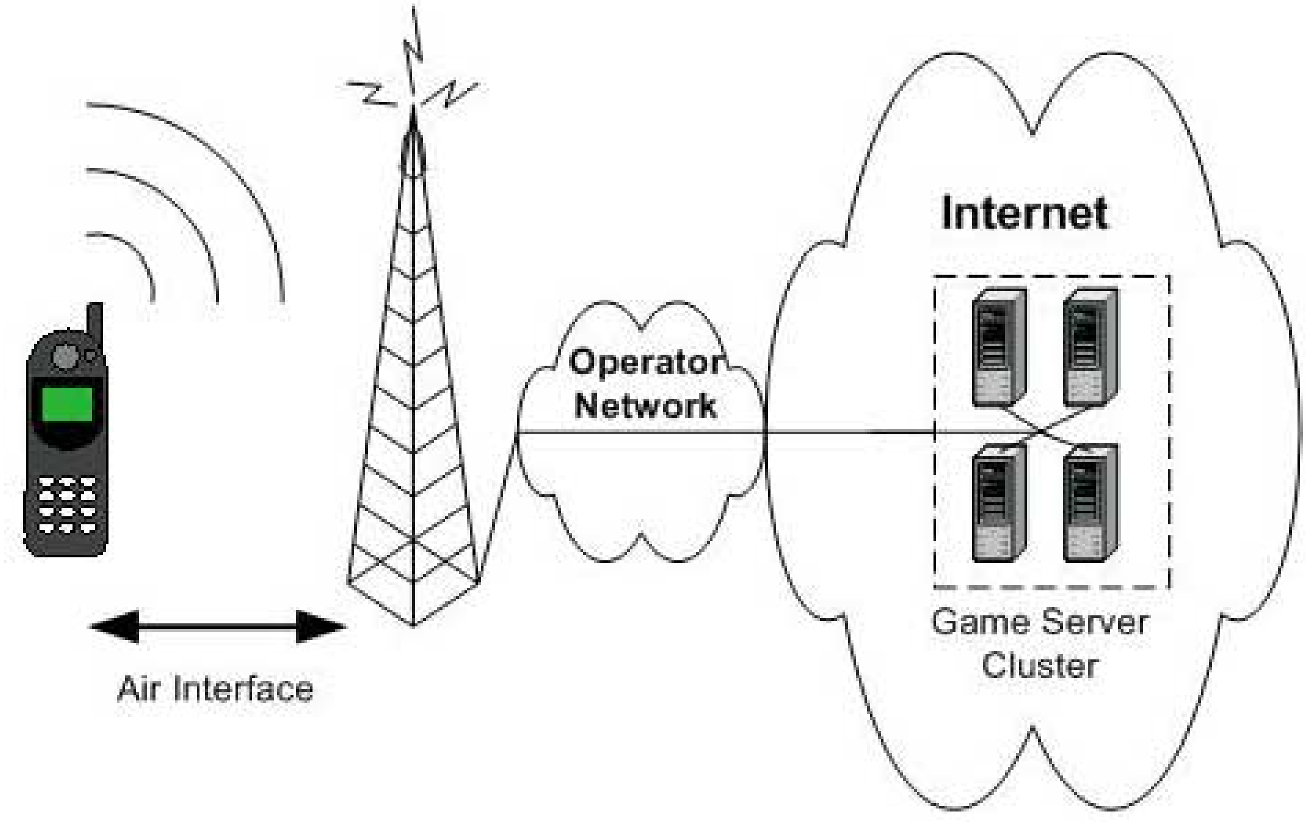
Introduction to J2ME



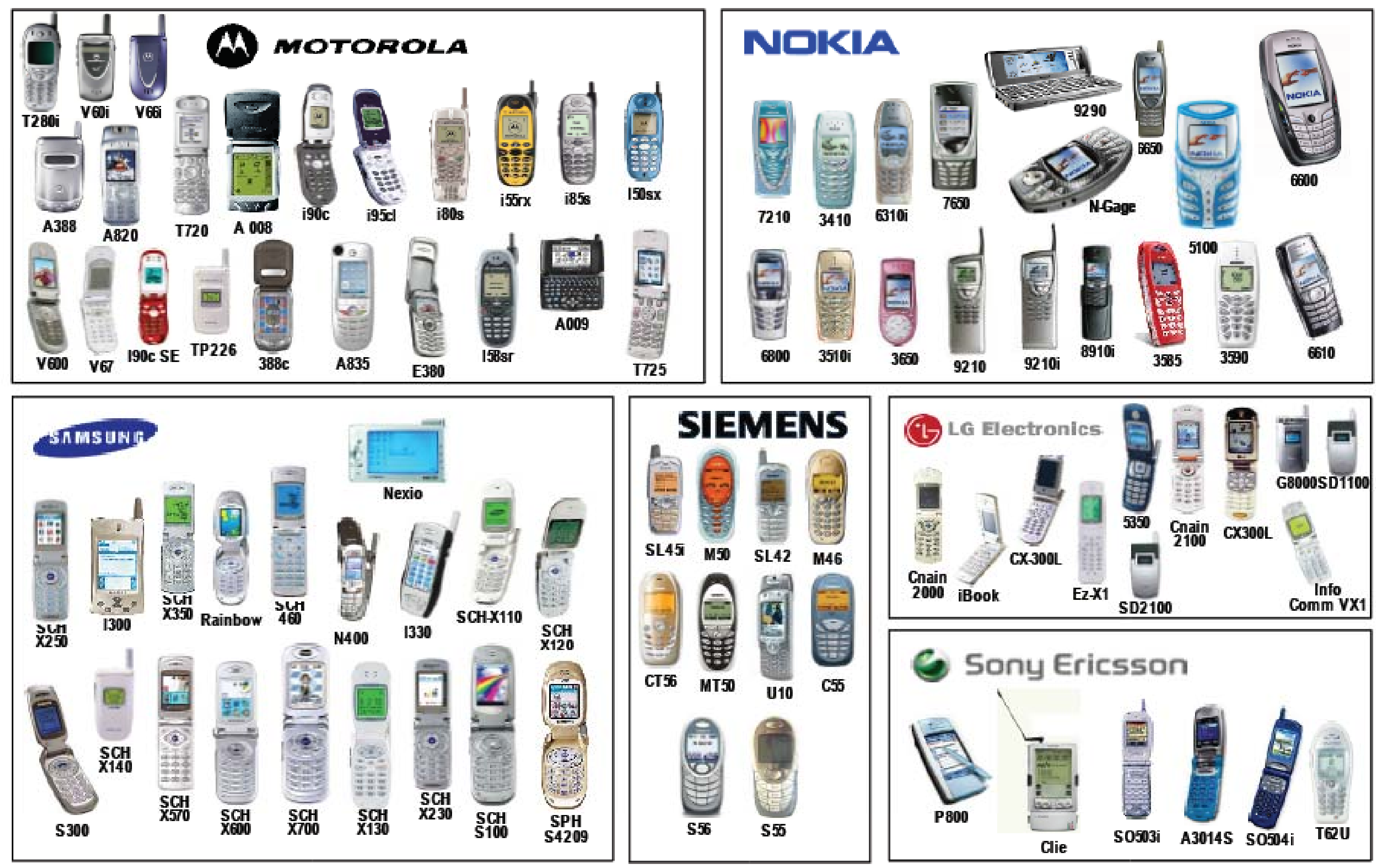
Arsitektur Jaringan dgn J2ME



Java 2 Platform Micro Edition (J2ME)

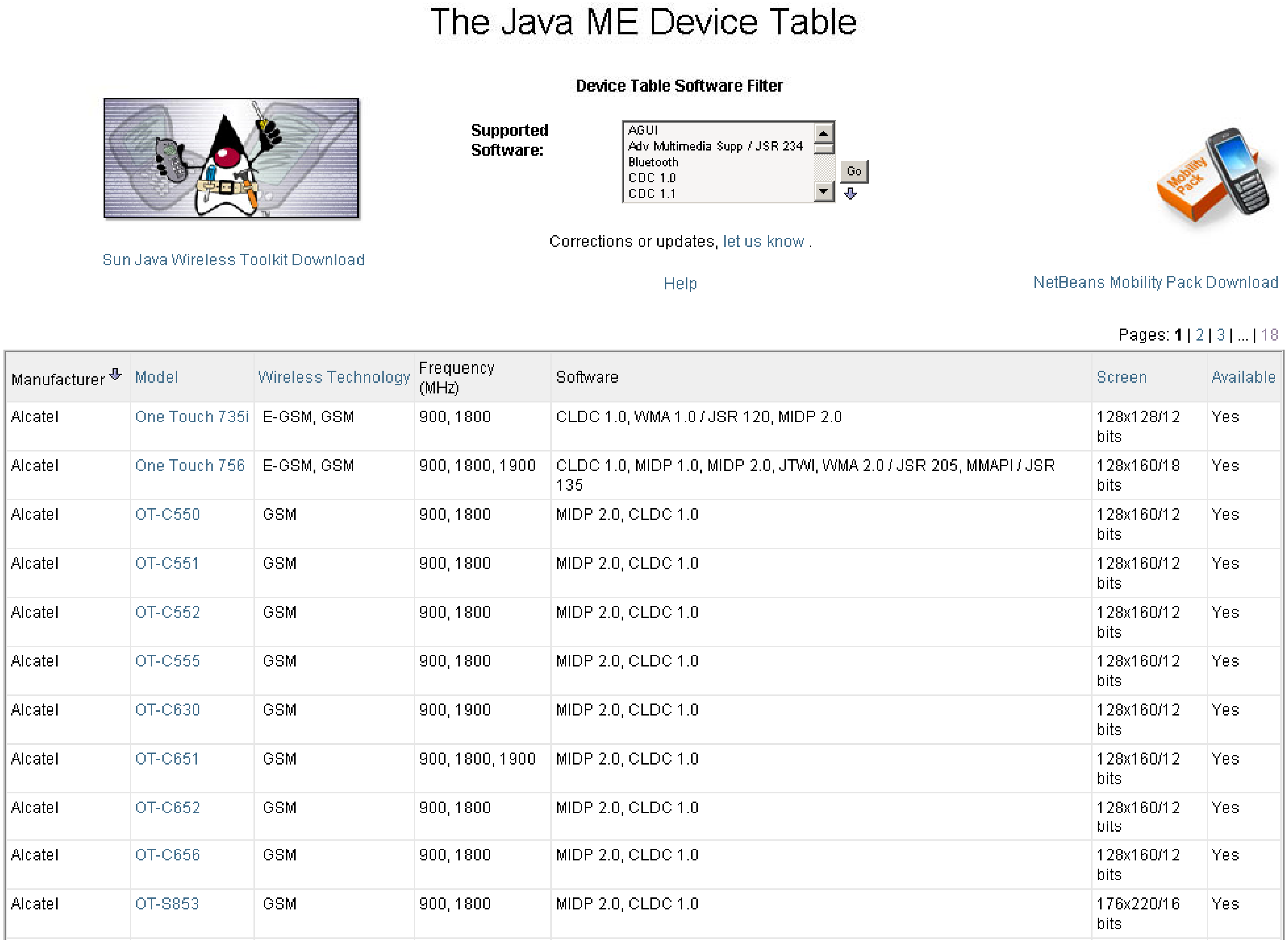
- Java platform for small devices
- A subset of J2SE
- Released mid June 1999
- Target devices:
  - Two-way pagers
  - Mobile phones, smart phones
  - PDAs (inc PocketPCs)
  - TVs, VCRs, CD players
- Almost every mobile phone support J2ME

J2ME Phones


MOTOROLA
T280i
V60i
V66i
A388
A820
T720
A 008
i90c
i95cl
i80s
i55rx
i85s
i50sx
V600
V67
i90c SE
TP226
388c
A835
E380
i58sr
A009
T725
NOKIA
9290
6650
6600
7210
3410
6310i
7650
N-Gage
5100
6800
3510i
3650
9210
9210i
8910i
3585
3590
6610
SAMSUNG
Nexio
SCH X250
i300
SCH X350
Rainbow
SCH 460
N400
i330
SCH-X110
SCH X120
S300
SCH X140
SCH X570
SCH X600
SCH X700
SCH X130
SCH X230
SCH S100
SPH S4209
SIEMENS
SL45i
M50
SL42
M46
CT56
MT50
U10
C55
S56
S55
LG Electronics
5350
Cnain 2100
CX300L
G8000
SD1100
Cnain 2000
iBook
CX-300L
Ez-X1
SD2100
Info Comm VX1
Sony Ericsson
P800
Clie
SO503i
A3014S
SO504i
T62U

J2ME Phones (Up to 15 Jan 2008)

http://developers.sun.com/mobility/device/pub/device/list.do?sort=manufacturer&filterIds=61&page=1



The Java ME Device Table

Device Table Software Filter

Supported Software:

Sun Java Wireless Toolkit Download

Corrections or updates, let us know .

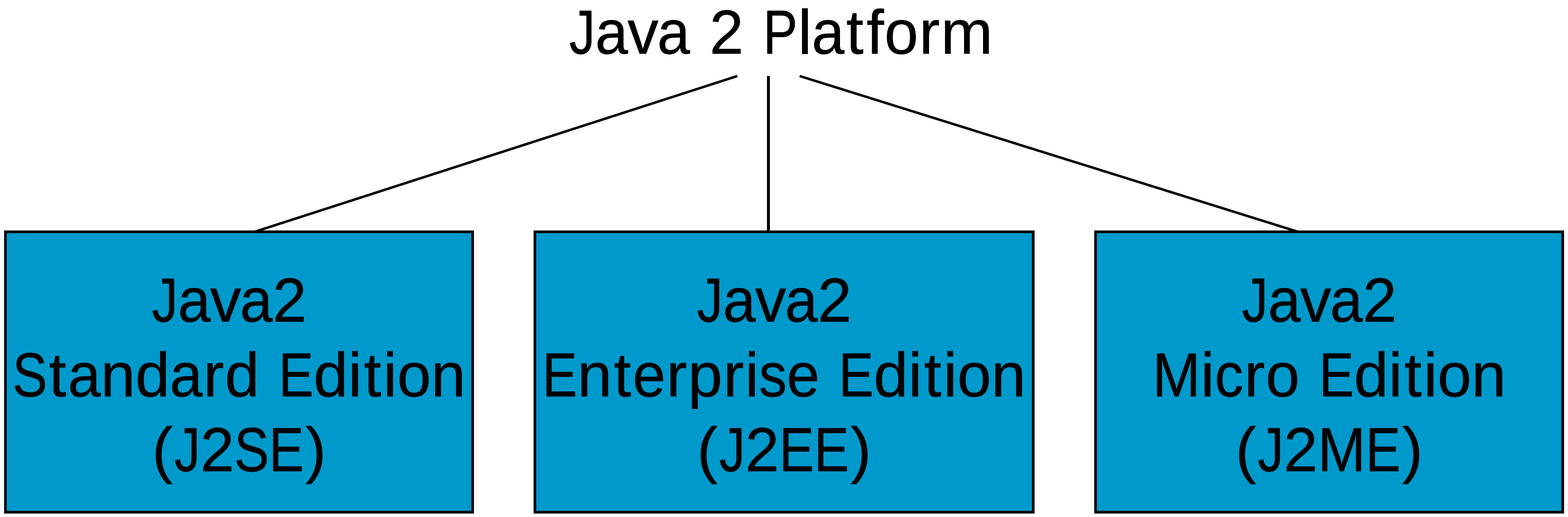
Help

NetBeans Mobility Pack Download

Pages: **1** | 2 | 3 | ... | 18

Manufacturer ⇩	Model	Wireless Technology	Frequency (MHz)	Software	Screen	Available
Alcatel	One Touch 735i	E-GSM, GSM	900, 1800	CLDC 1.0, WMA 1.0 / JSR 120, MIDP 2.0	128x128/12 bits	Yes
Alcatel	One Touch 756	E-GSM, GSM	900, 1800, 1900	CLDC 1.0, MIDP 1.0, MIDP 2.0, JTWI, WMA 2.0 / JSR 205, MMAPI / JSR 135	128x160/18 bits	Yes
Alcatel	OT-C550	GSM	900, 1800	MIDP 2.0, CLDC 1.0	128x160/12 bits	Yes
Alcatel	OT-C551	GSM	900, 1800	MIDP 2.0, CLDC 1.0	128x160/12 bits	Yes
Alcatel	OT-C552	GSM	900, 1800	MIDP 2.0, CLDC 1.0	128x160/12 bits	Yes
Alcatel	OT-C555	GSM	900, 1800	MIDP 2.0, CLDC 1.0	128x160/12 bits	Yes
Alcatel	OT-C630	GSM	900, 1900	MIDP 2.0, CLDC 1.0	128x160/12 bits	Yes
Alcatel	OT-C651	GSM	900, 1800, 1900	MIDP 2.0, CLDC 1.0	128x160/12 bits	Yes
Alcatel	OT-C652	GSM	900, 1800	MIDP 2.0, CLDC 1.0	128x160/12 bits	Yes
Alcatel	OT-C656	GSM	900, 1800	MIDP 2.0, CLDC 1.0	128x160/12 bits	Yes
Alcatel	OT-S853	GSM	900, 1800	MIDP 2.0, CLDC 1.0	176x220/16 bits	Yes

3 Java Platforms



Java 2 Platform

Java2
Standard Edition
(J2SE)

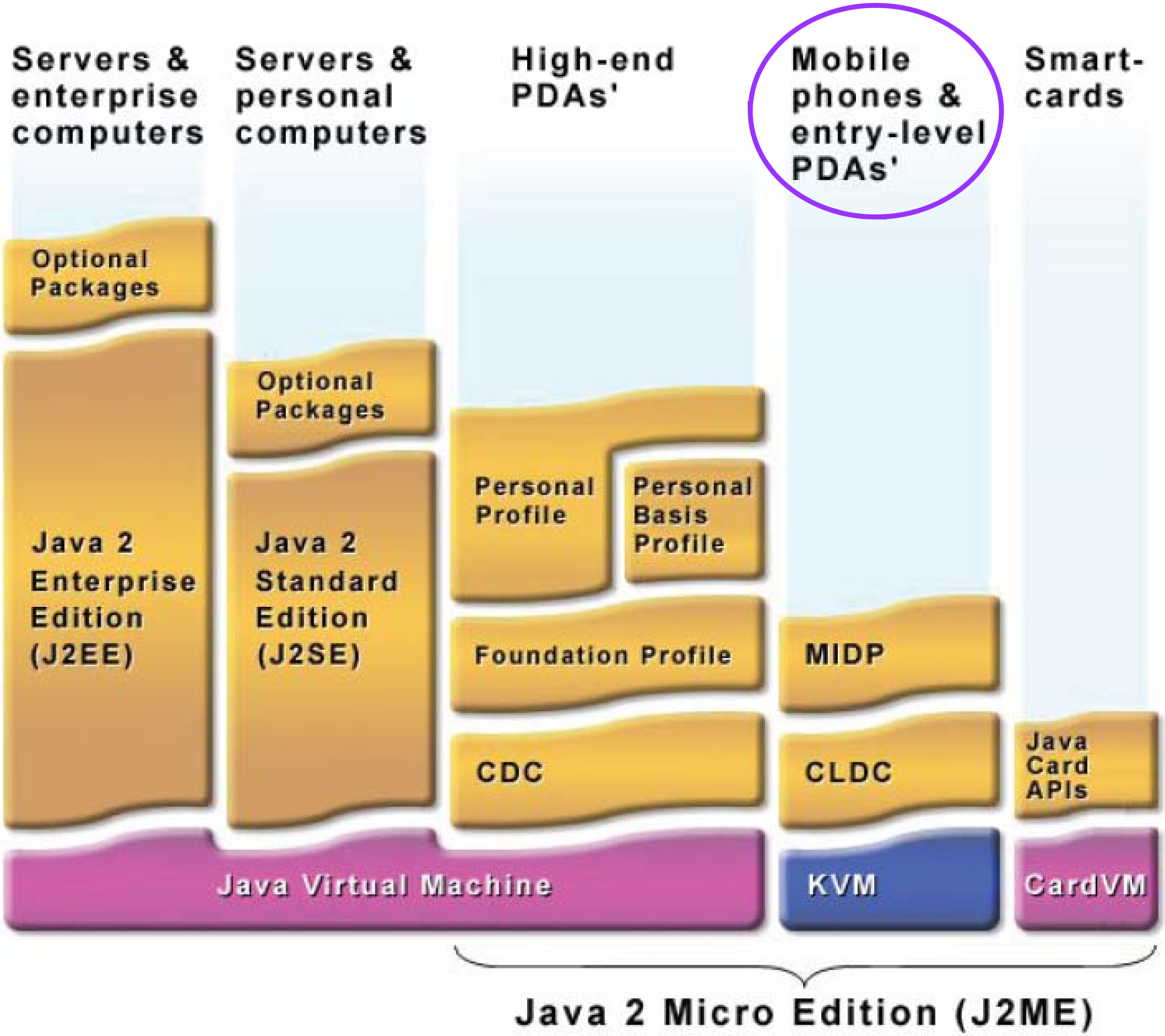
Java2
Enterprise Edition
(J2EE)

Java2
Micro Edition
(J2ME)

Standard desktop &
Workstation Applications

Heavy duty server
systems

Small & memory
Constrained devices



Servers & enterprise computers

Servers & personal computers

High-end PDAs'

Mobile phones & entry-level PDAs'

Smart-cards

Optional Packages

Java 2 Enterprise Edition (J2EE)

Optional Packages

Java 2 Standard Edition (J2SE)

Personal Profile

Personal Basis Profile

Foundation Profile

CDC

MIDP

CLDC

Java Card APIs

Java Virtual Machine

KVM

CardVM

Java 2 Micro Edition (J2ME)

J2ME Architecture

- To increase the flexibility of design, the J2ME consists of two distinct layers:

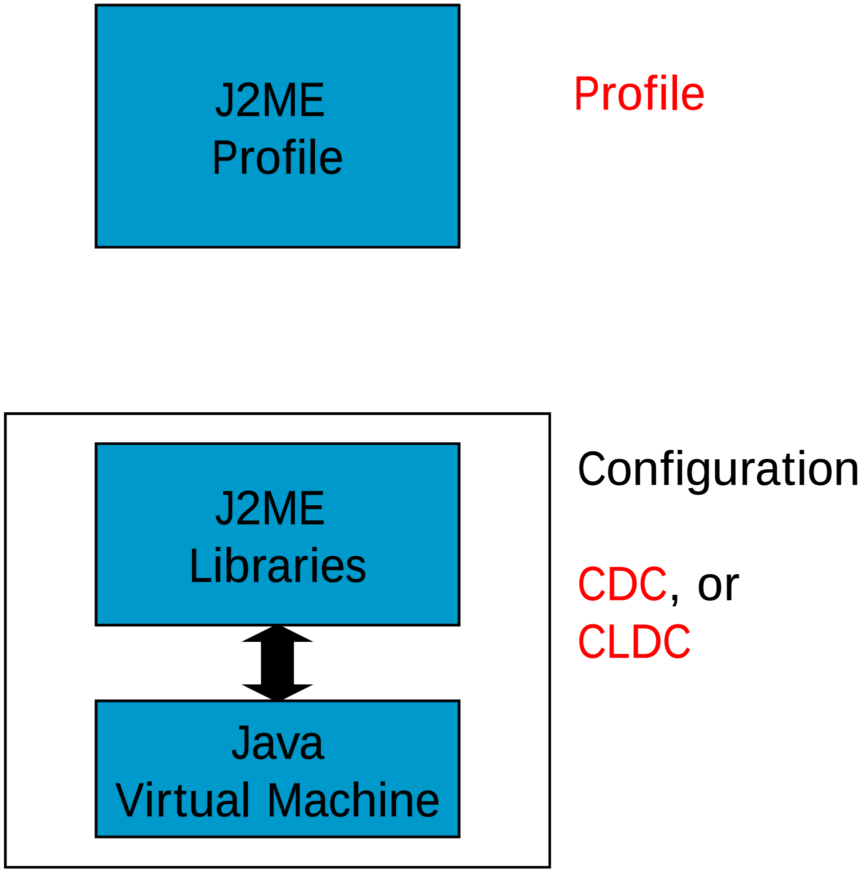
  Configurations and Profiles

- Configuration
  - Defines the minimum Java technology for a broad range of devices with similar capabilities

- Profile
  - Provides capabilities, on top of configuration, for a specific device type

J2ME Architecture

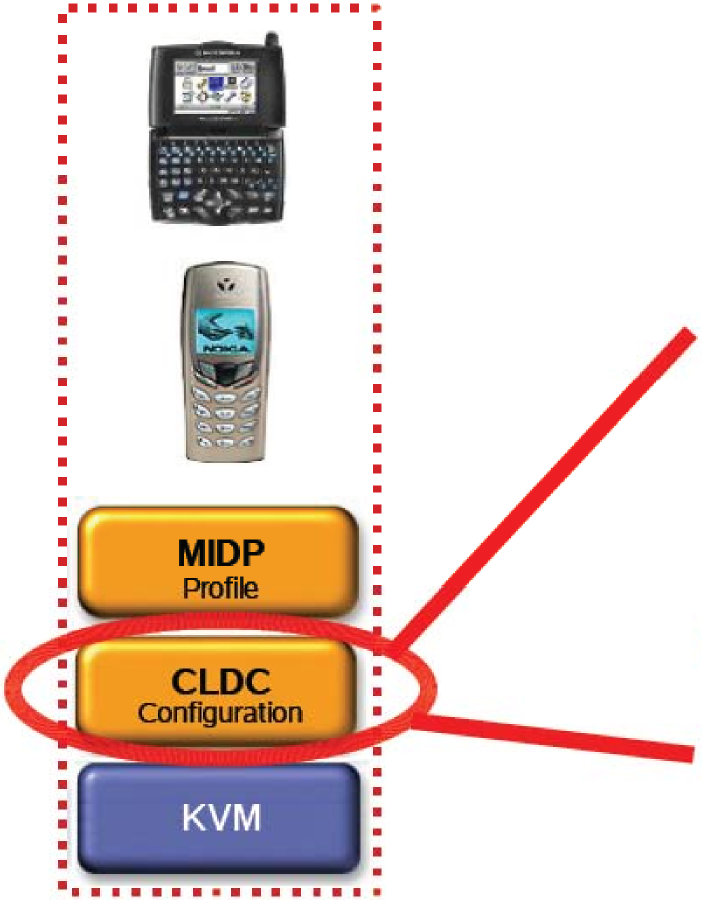
- Two types of J2ME configurations
  1. Connected Device Configuration
  2. Connected Limited Device Configuration



CLDC vs CDC

- CLDC
  - 160 Kbytes to 512 Kbytes of total memory available
  - 16-bit or 32-bit processor
  - Low power consumption and often operating with battery power
  - Connectivity with limited bandwidth.
- CDC
  - 2Mbytes or more memory for Java platform
  - 32-bit processor
  - High bandwidth network connection, most often using TCP/IP

CLDC

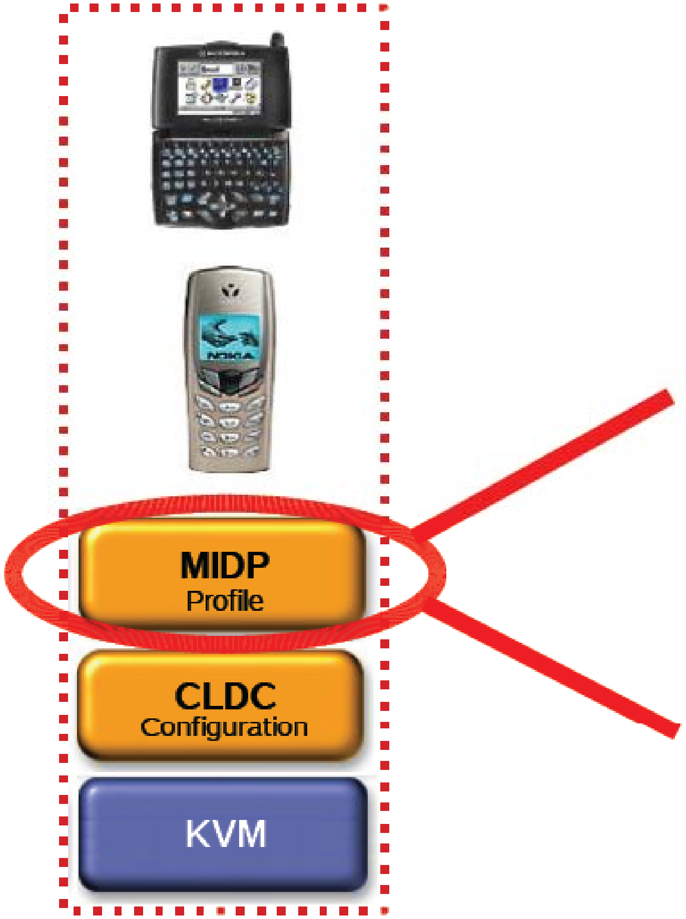


- Core JLS & JVMS features
- Core java.* libraries
- Additional I/O and networking libs
- Security features
- Internationalization

Mobile Information Device Profile (MIDP)

- Is a set of APIs that allow developers to control mobile device-specific problems
  - i.e. user interfaces, local storage and client application lifecycles etc.
- MIDlets minimum requirements
  - 96 x 54 pixels mono screen
  - two-way wireless network
  - input device (i.e. keypad)
  - 128 KB for CLDC/MIDP class and another 32 KB for the KVM
- Midlets are the most important and popular applications in the J2ME family.

MIDP

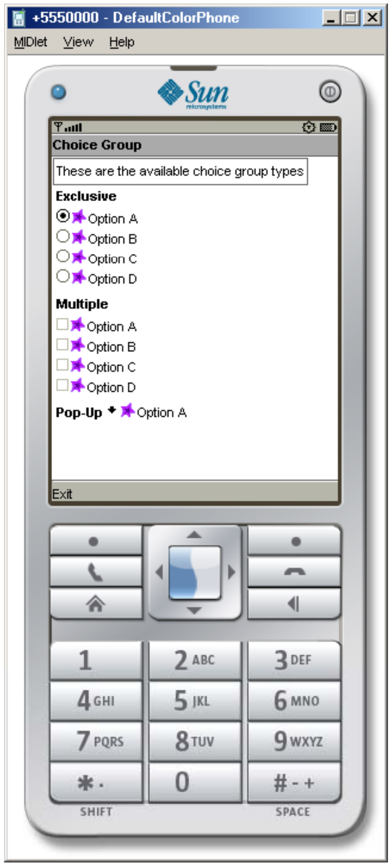


- Application model (life-cycle, packaging)
- Persistent storage (RMS APIs)
- Networking (HTTP, etc.)
- User interface (High and low level APIs)

Building J2ME Apps- Tool

- We will use **Sun Java Wireless Toolkit 2.x for CLDC** (The newest version is 2.5.2 in Jan 2008) which can be downloaded from

`http://java.sun.com/j2me/download.html`



Choice Group

These are the available choice group types

Exclusive

● Option A
○ Option B
○ Option C
○ Option D

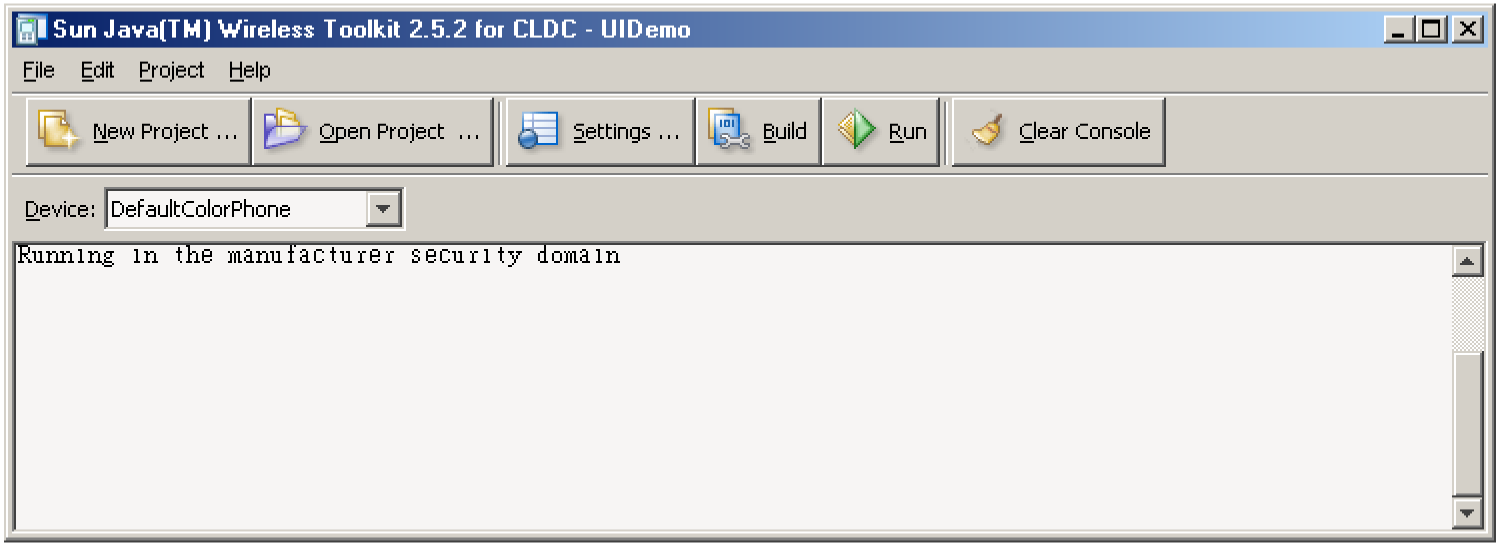
Multiple

□ Option A
□ Option B
□ Option C
□ Option D

Pop-Up Option A

J2ME Wireless Toolkit Demo

- Launch the Wireless Toolkit:
  - Start > Programs > Sun Java(TM) Wireless Toolkit 2.5.2 for CLDC



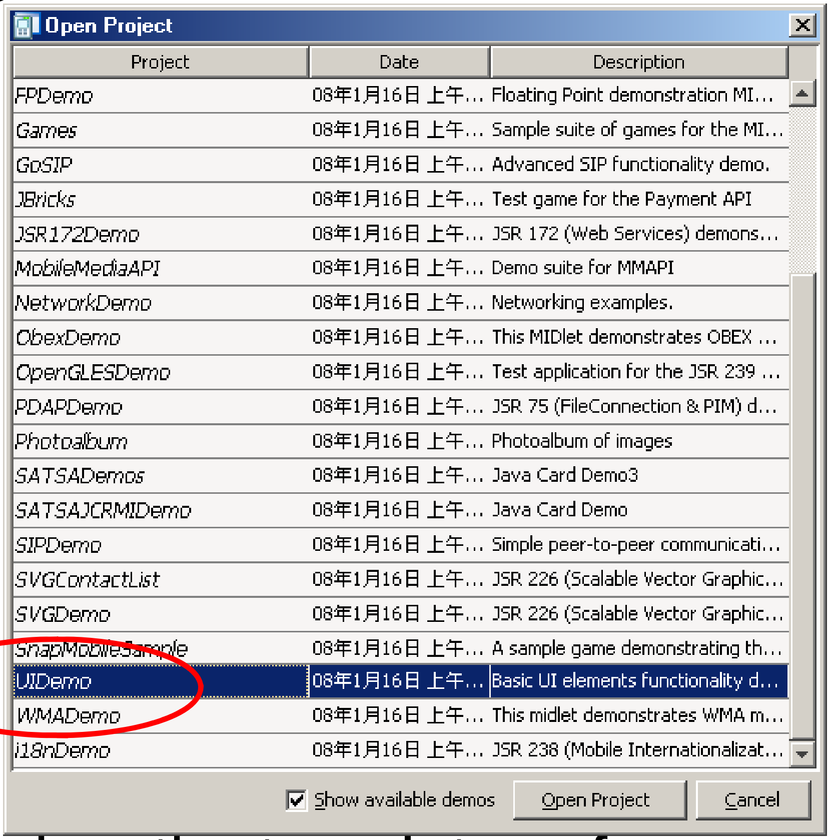
Device: DefaultColorPhone

```
Running in the manufacturer security domain
```

- WTK already includes a set of demo programs ready to run.

J2ME Wireless Toolkit Demo

- Select menu item
  `File` > **`Open Project ...`**
- Select **`UIDemo`** and click **`Open Project`**.



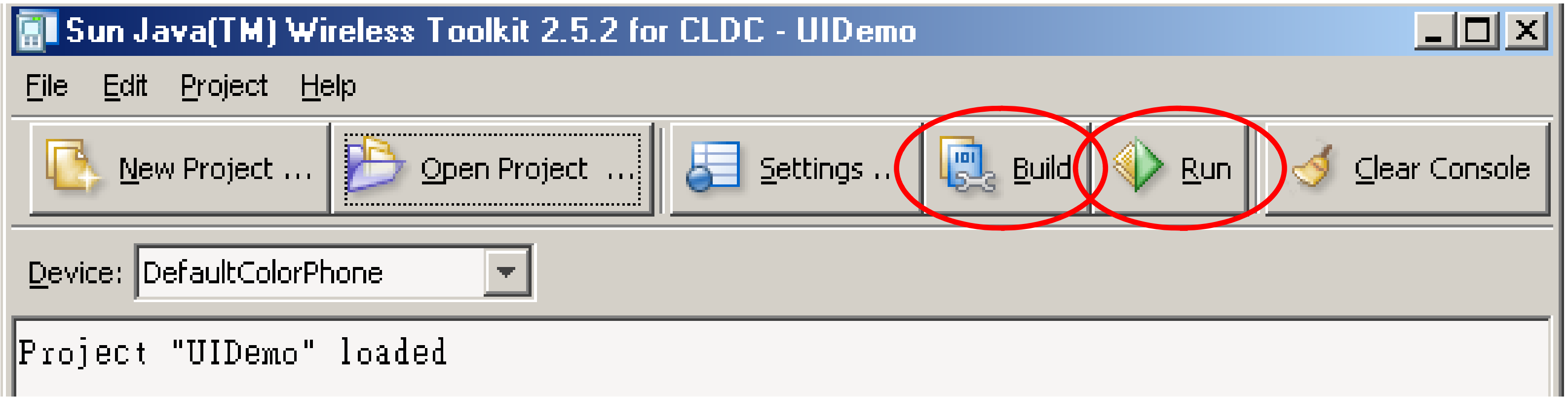
Project	Date	Description
FPDemo	08年1月16日 上午...	Floating Point demonstration MI...
Games	08年1月16日 上午...	Sample suite of games for the MI...
GoSIP	08年1月16日 上午...	Advanced SIP functionality demo.
JBricks	08年1月16日 上午...	Test game for the Payment API
JSR172Demo	08年1月16日 上午...	JSR 172 (Web Services) demons...
MobileMediaAPI	08年1月16日 上午...	Demo suite for MMAPI
NetworkDemo	08年1月16日 上午...	Networking examples.
ObexDemo	08年1月16日 上午...	This MIDlet demonstrates OBEX ...
OpenGLESDemo	08年1月16日 上午...	Test application for the JSR 239 ...
PDAPDemo	08年1月16日 上午...	JSR 75 (FileConnection & PIM) d...
Photoalbum	08年1月16日 上午...	Photoalbum of images
SATSADemos	08年1月16日 上午...	Java Card Demo3
SATSAJCRMIDemo	08年1月16日 上午...	Java Card Demo
SIPDemo	08年1月16日 上午...	Simple peer-to-peer communicati...
SVGContactList	08年1月16日 上午...	JSR 226 (Scalable Vector Graphic...
SVGDemo	08年1月16日 上午...	JSR 226 (Scalable Vector Graphic...
SnapMobileSample	08年1月16日 上午...	A sample game demonstrating th...
UIDemo	08年1月16日 上午...	Basic UI elements functionality d...
WMADemo	08年1月16日 上午...	This midlet demonstrates WMA m...
i18nDemo	08年1月16日 上午...	JSR 238 (Mobile Internationalizat...

☑ Show available demos

- The projects can be used as the templates of your applications.

J2ME Wireless Toolkit Demo

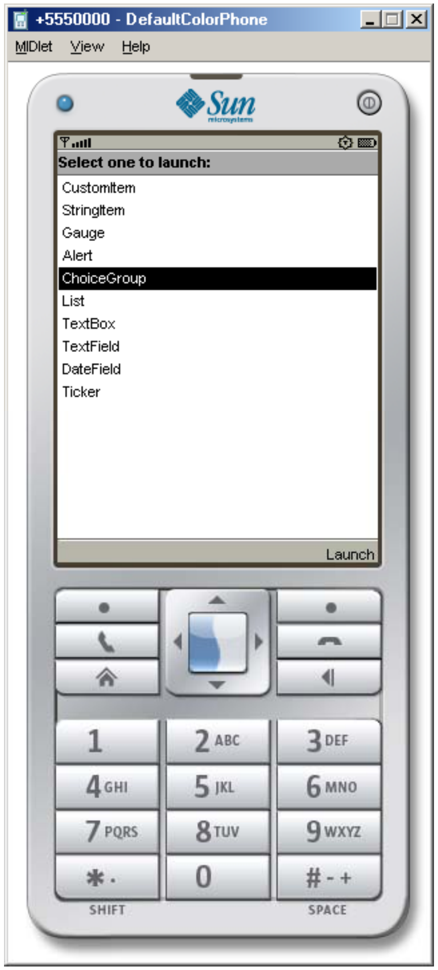
- Click the **Build** and then the **Run** buttons.



```
Project "UIDemo" loaded
```

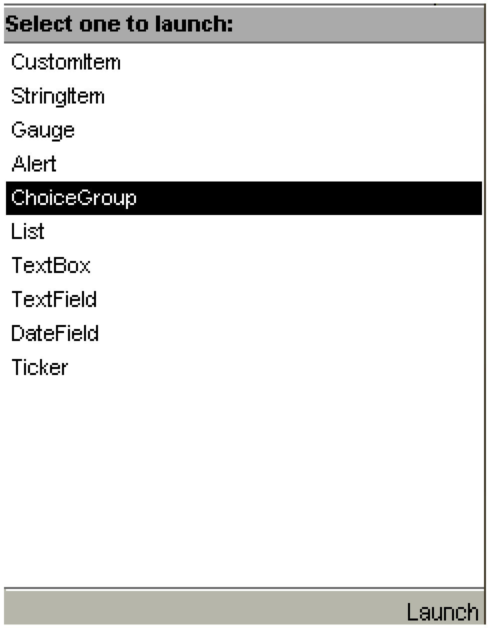
J2ME Wireless Toolkit Demo

- The main menu screen is shown up. You can choose a program and select Launch to start the program.

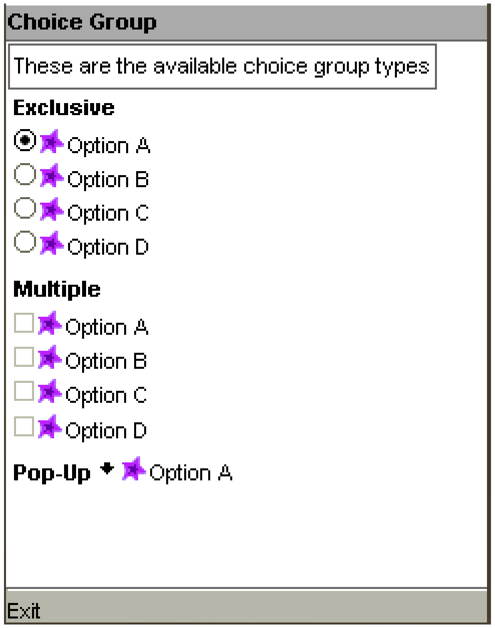


Select one to launch:

CustomItem
StringItem
Gauge
Alert
ChoiceGroup
List
TextBox
TextField
DateField
Ticker


Select one to launch:

CustomItem
StringItem
Gauge
Alert
ChoiceGroup
List
TextBox
TextField
DateField
Ticker


Choice Group

These are the available choice group types

Exclusive

● Option A
○ Option B
○ Option C
○ Option D

Multiple

□ Option A
□ Option B
□ Option C
□ Option D

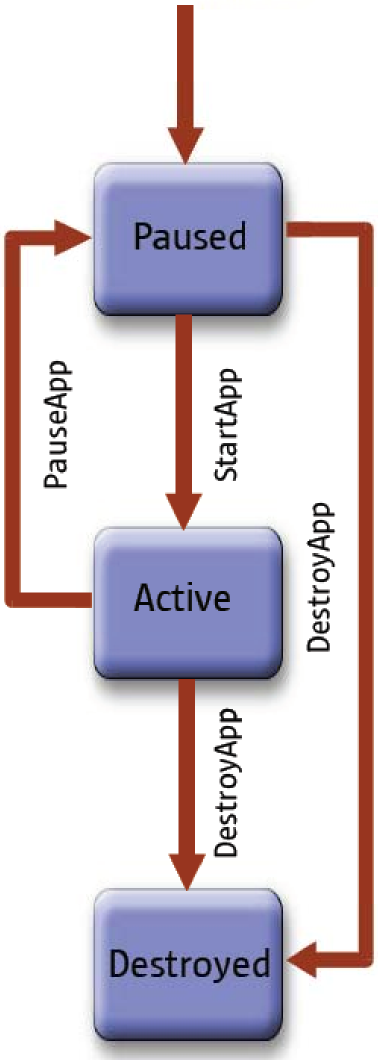
Pop-Up Option A

MIDlet Programming

- Any MIDP application must extends **MIDlet**
- This is the MIDP equivalent of an applet, where starting/stopping is under the control of the environment
- Like Java applets, MIDlets have an application life cycle while running on a mobile device.

MIDlet Transition States

- Specifically, a MIDlet can be in one of three states as shown:



Why do we need a **Paused** state?

Midlet Skeleton

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

public class MyApp extends MIDlet {
        public void startApp() {
            // start up code
        }

        public void pauseApp() {
            // we aren't showing any more
        }

        public void destroyApp(boolean unconditional) {
            // clean up
        }
}
```

Note that startApp(), pauseApp() and destroyApp() are abstract methods.

You Midlet program **must** override these 3 methods even though you are not do anything in it.

Two Level API

- There are two areas the API which you should be concerned with - the high and low-level API.
- High-Level Provides input elements such as,
  - text fields, choices, and form
- Low-level is for drawing on Canvases and capturing keyed events
- All MIDlet applications need to import the necessary midlet and lcdui packages:
  - **import javax.microedition.midlet.*;**
  - **import javax.microedition.lcdui.*;**

Displaying Objects

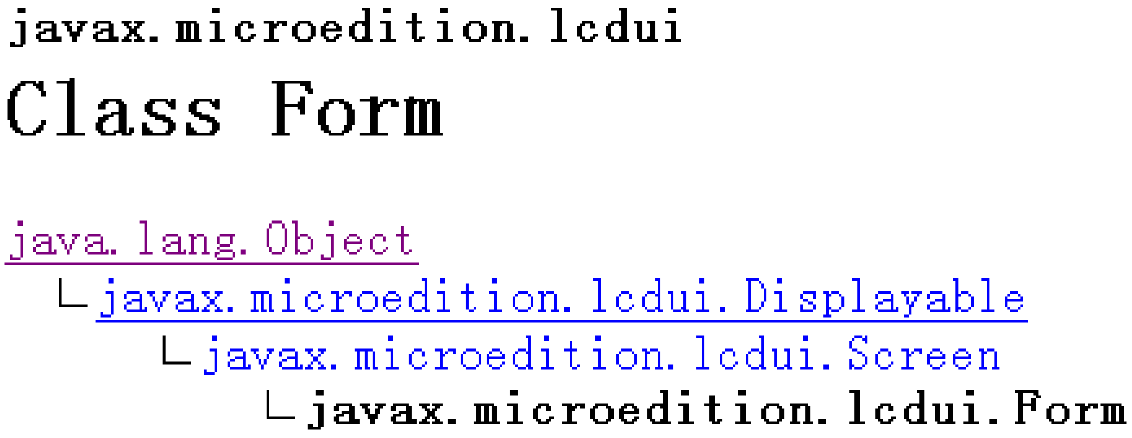
- High-level Screens have a base class called Displayable.
- To show something on a MIDP device, you need to obtain the device's display
  - javax.microedition.lcdui.Display class.
- This Display class is the one and only display manager for each active MIDlet and provides information about the device's display capability.
- Subclassed Displayable classes will fill the whole screen

Displaying Objects

- To show a Displable object you must use the setCurrent() method on the Display object.

```
Form mainForm = new Form ("First Program ");
Display display = Display.getDisplay(this);
display.setCurrent (mainForm);
```

Note that Form is a Displayable subclass.

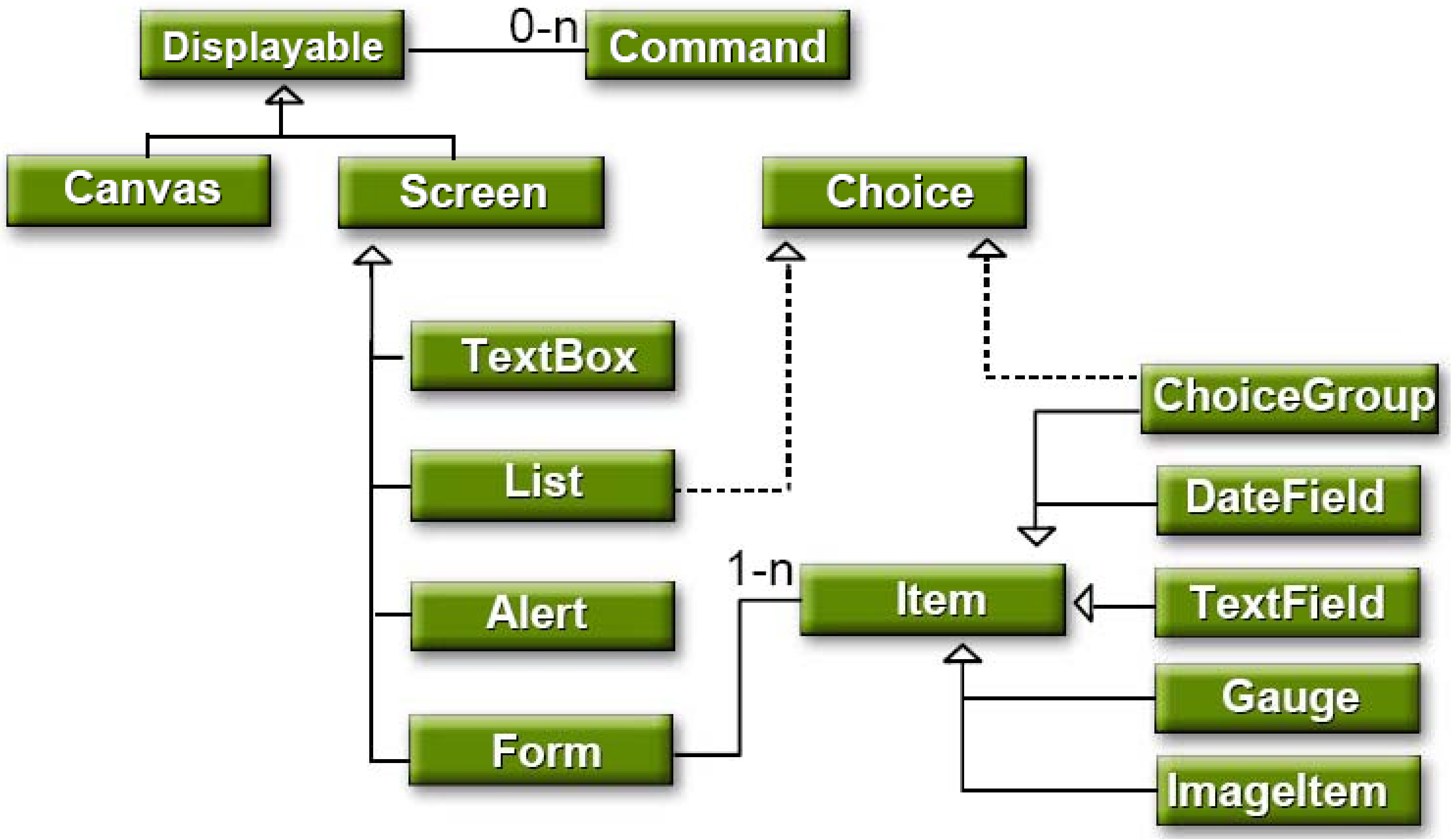


javax.microedition.lcdui

Class Form

```
java.lang.Object
  └javax.microedition.lcdui.Displayable
      └javax.microedition.lcdui.Screen
          └javax.microedition.lcdui.Form
```

Major classes in the lcdui package

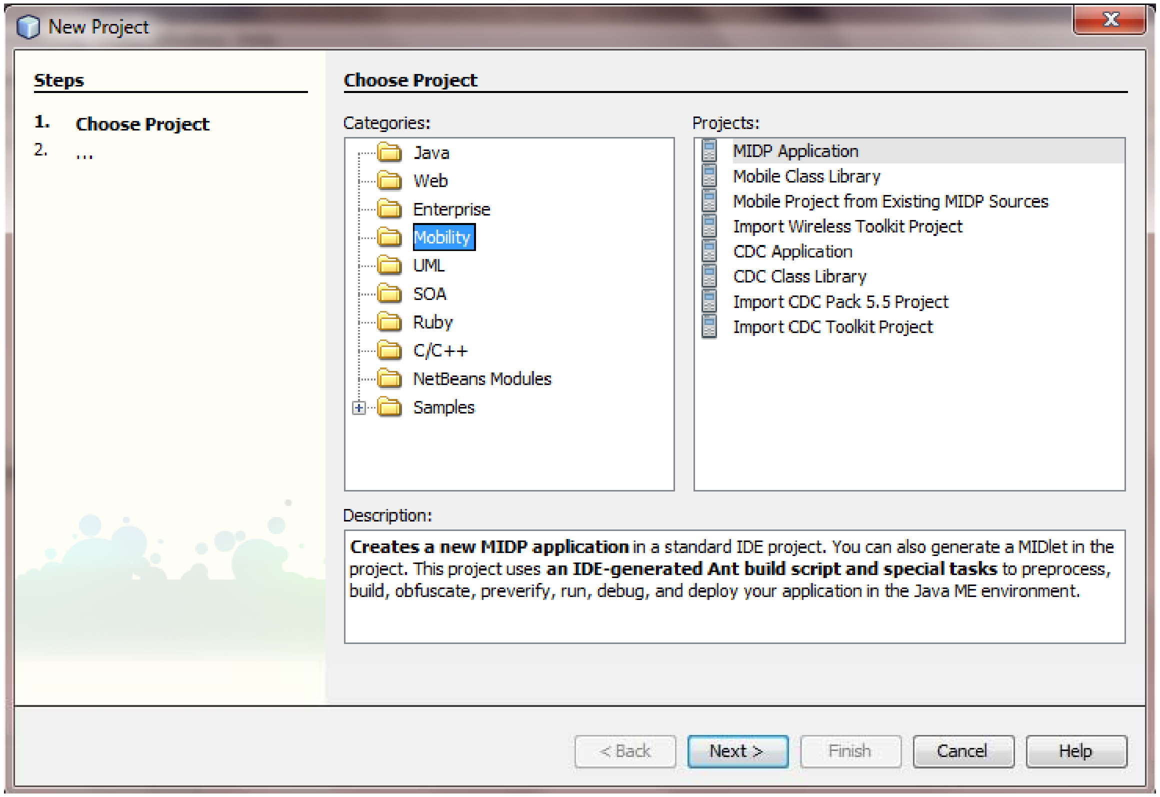


First Example - HelloWorld

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

public class HelloWorld extends MIDlet {

   public HelloWorld() {
   }

   public void startApp() {

      Form form = new Form( "First Program" );
      form.append( "Hello World"  );
      Display.getDisplay(this).setCurrent( form );
   }

   public void pauseApp() {
   }

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

Building the MIDlet using Netbeans 6

- Run the program Netbeans IDE 6.0, **File>New Project**
- Create new Project, Project Name : **FirstProgram**
- Categories : **Mobility,** Projects : **MIDP Application**



New Project

Steps

1. **Choose Project**
2. ...

Choose Project

Categories:

- Java
- Web
- Enterprise
- Mobility
- UML
- SOA
- Ruby
- C/C++
- NetBeans Modules
- Samples

Projects:

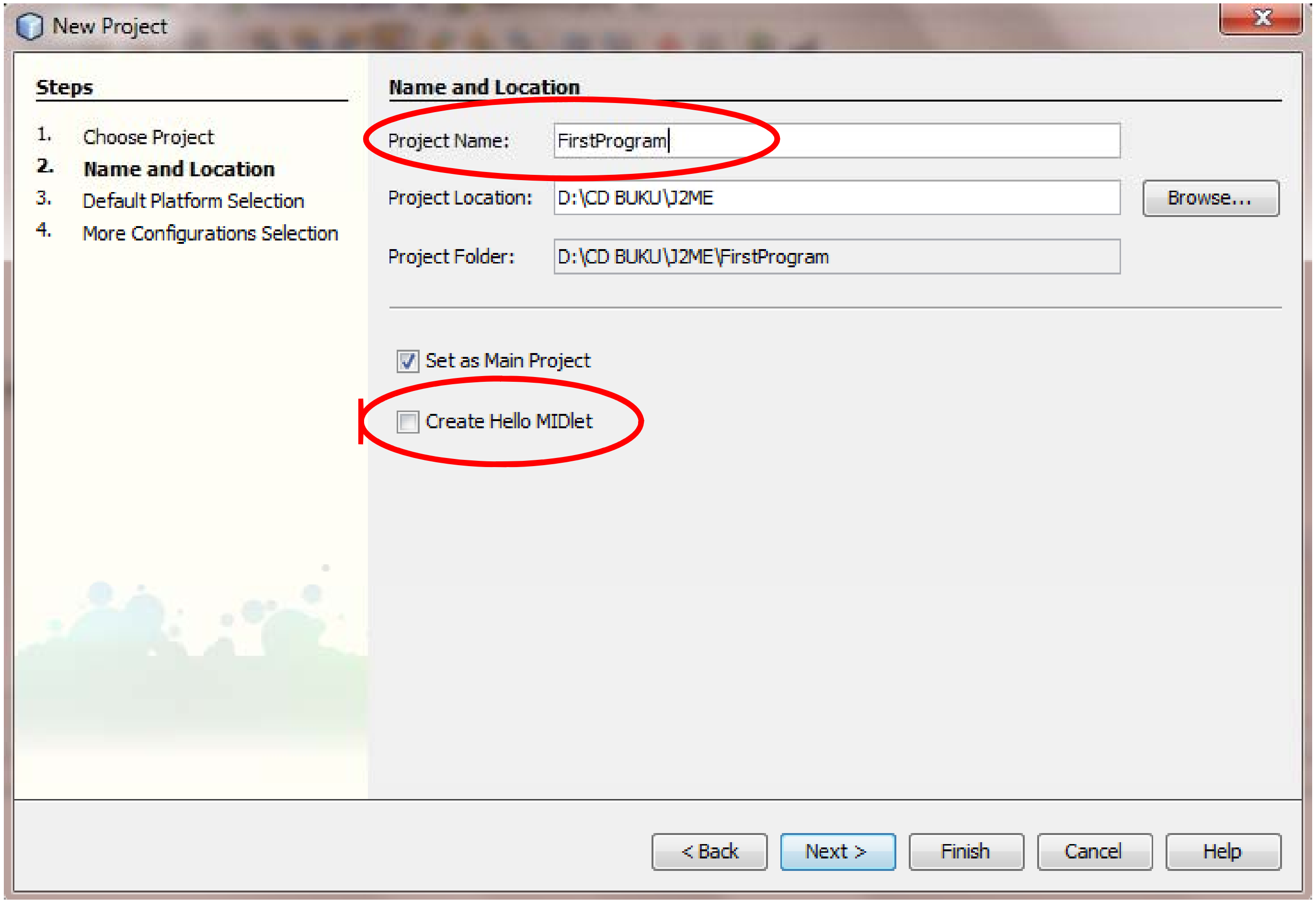
- MIDP Application
- Mobile Class Library
- Mobile Project from Existing MIDP Sources
- Import Wireless Toolkit Project
- CDC Application
- CDC Class Library
- Import CDC Pack 5.5 Project
- Import CDC Toolkit Project

Description:

Creates a new MIDP application in a standard IDE project. You can also generate a MIDlet in the project. This project uses **an IDE-generated Ant build script and special tasks** to preprocess, build, obfuscate, preverify, run, debug, and deploy your application in the Java ME environment.

Building the MIDlet using Netbeans 6

- Project Name : **FirstProgram**
- Uncheck : **Create Hello Midlet**



New Project

Steps

1. Choose Project
2. **Name and Location**
3. Default Platform Selection
4. More Configurations Selection

Name and Location

Project Name: FirstProgram

Project Location: D:\CD BUKU\J2ME

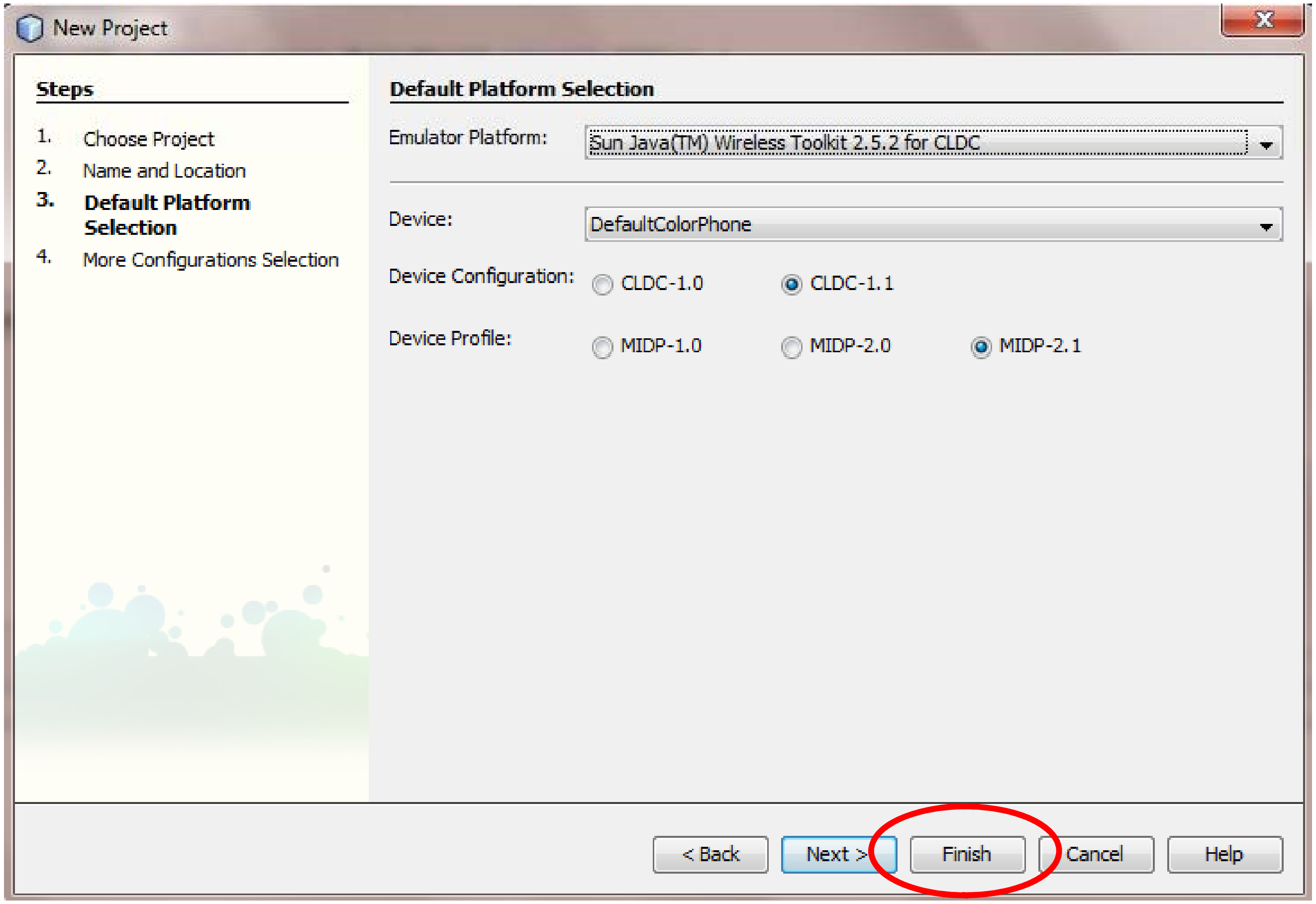
Project Folder: D:\CD BUKU\J2ME\FirstProgram

☑ Set as Main Project

□ Create Hello MIDlet

Building the MIDlet using Netbeans 6

- Device Configuration : **CLDC-1.1**
- Device Profile : **MIDP-2.1**
- Click : **Finish**



New Project

Steps

1. Choose Project
2. Name and Location
3. **Default Platform Selection**
4. More Configurations Selection

Default Platform Selection

Emulator Platform: Sun Java(TM) Wireless Toolkit 2.5.2 for CLDC

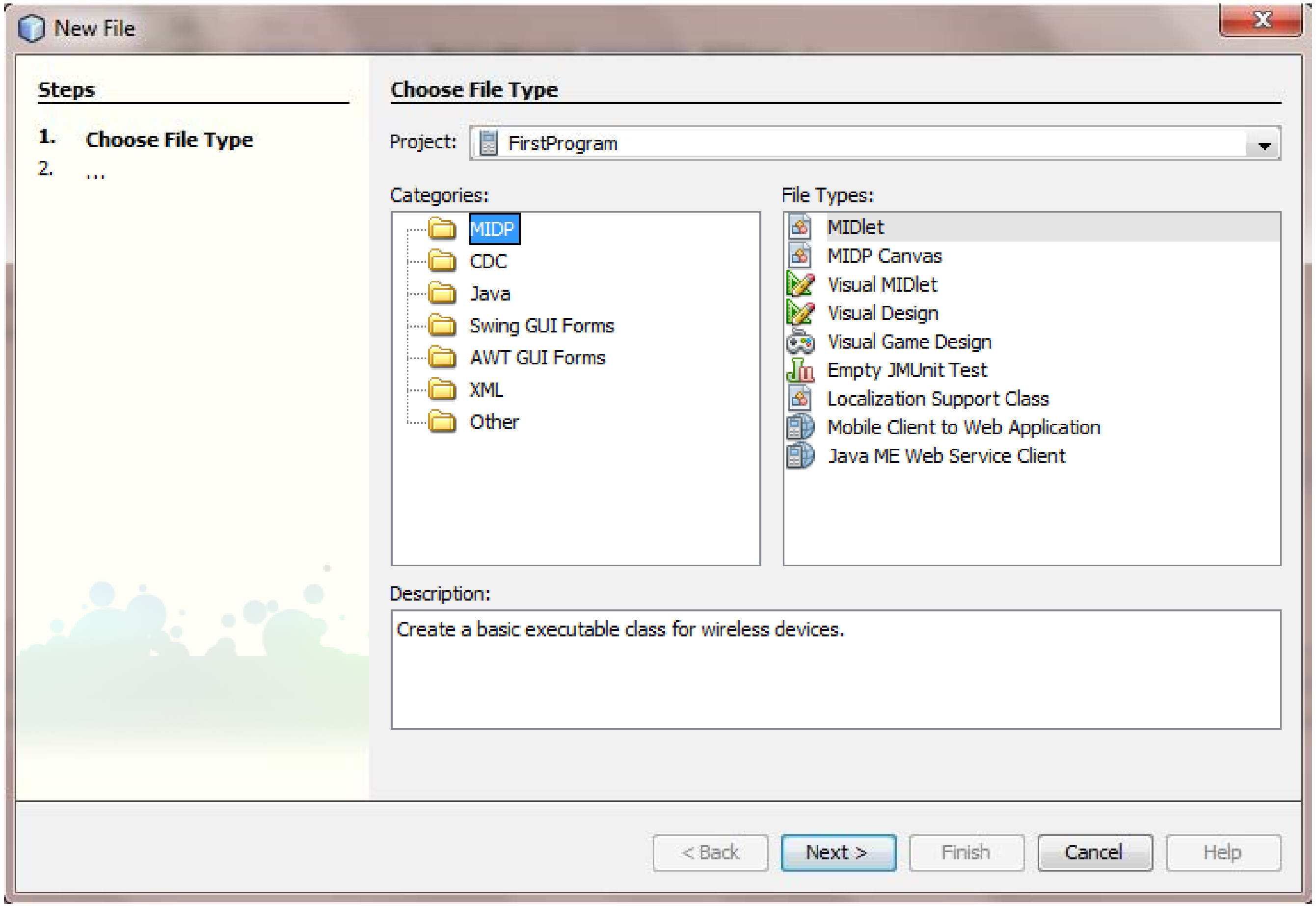
Device: DefaultColorPhone

Device Configuration: ○ CLDC-1.0 ● CLDC-1.1

Device Profile: ○ MIDP-1.0 ○ MIDP-2.0 ● MIDP-2.1

Building the MIDlet using Netbeans 6

- **File > New File…**
- Categories : **MIDP**
- File Types: **MIDlet**



New File

Steps

1. Choose File Type
2. …

Choose File Type

Project: FirstProgram

Categories: MIDP, CDC, Java, Swing GUI Forms, AWT GUI Forms, XML, Other

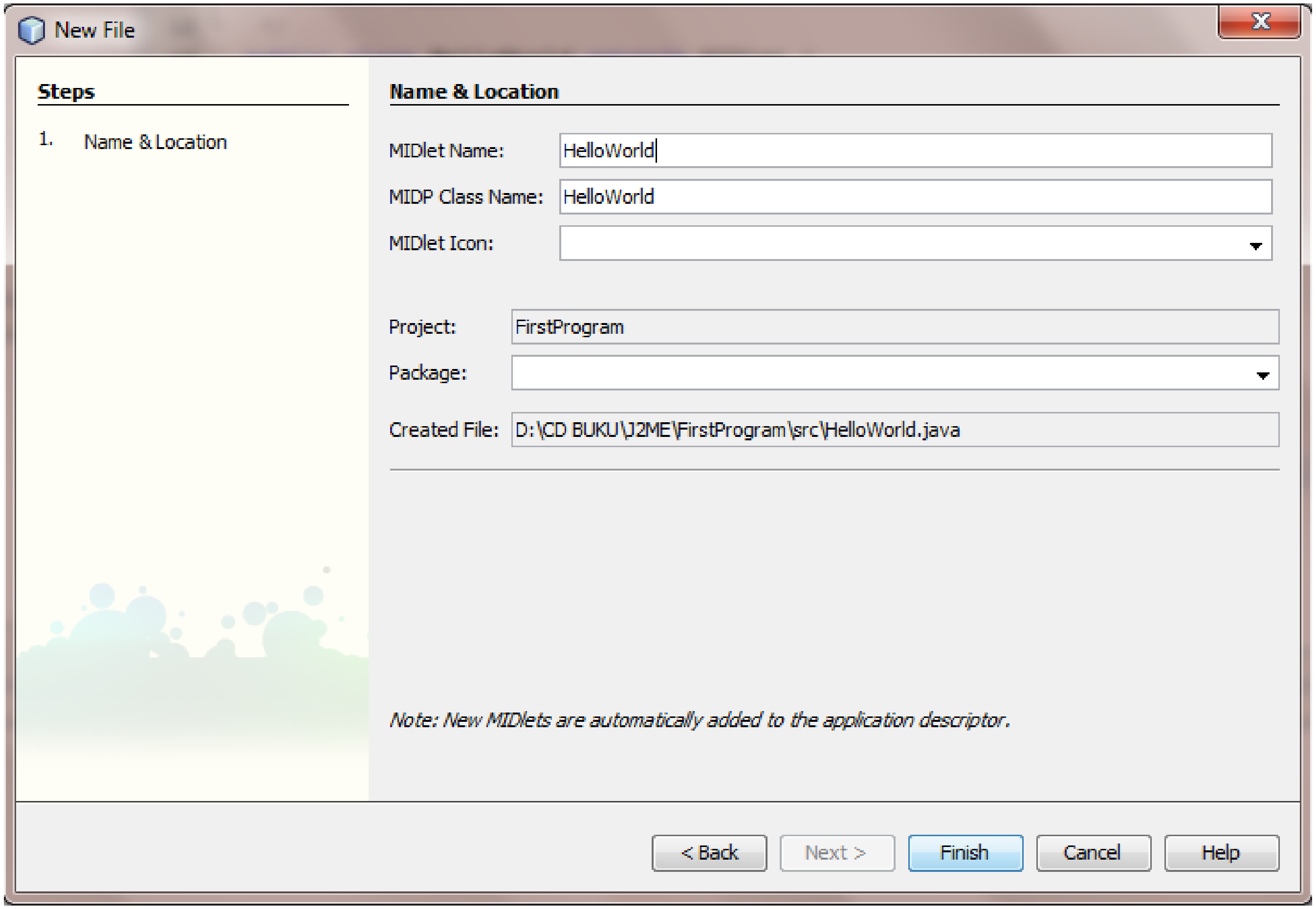
File Types: MIDlet, MIDP Canvas, Visual MIDlet, Visual Design, Visual Game Design, Empty JMUnit Test, Localization Support Class, Mobile Client to Web Application, Java ME Web Service Client

Description:

Create a basic executable class for wireless devices.

Building the MIDlet using Netbeans 6

- Class Name : **HelloWorld**
- Click : **Finish**



New File

Steps

1. Name & Location

Name & Location

MIDlet Name: HelloWorld

MIDP Class Name: HelloWorld

MIDlet Icon:

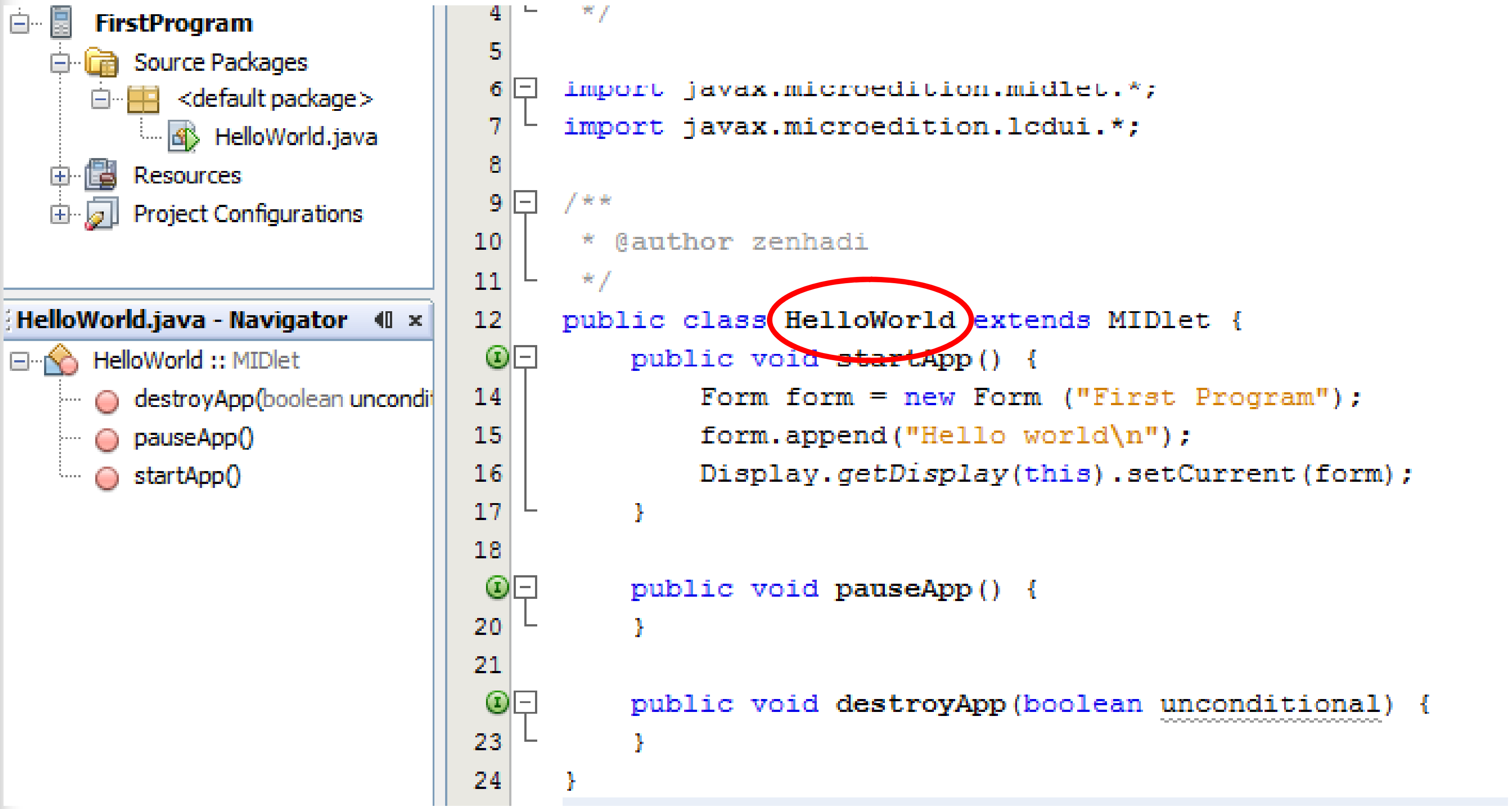
Project: FirstProgram

Package:

Created File: D:\CD BUKU\J2ME\FirstProgram\src\HelloWorld.java

Note: New MIDlets are automatically added to the application descriptor.

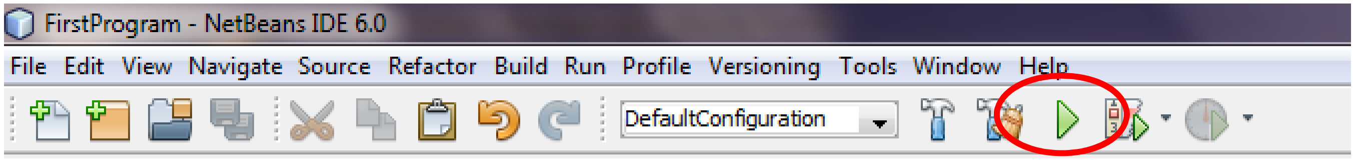
Building the MIDlet using Netbeans 6



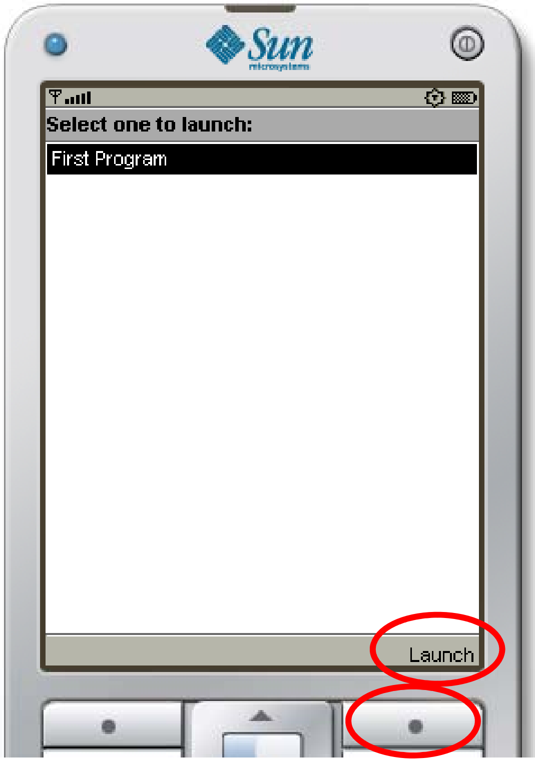
```
 */

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

/**
 * @author zenhadi
 */
public class HelloWorld extends MIDlet {
    public void startApp() {
        Form form = new Form ("First Program");
        form.append("Hello world\n");
        Display.getDisplay(this).setCurrent(form);
    }

    public void pauseApp() {
    }

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

Building and Run the MIDlet

- Click the **Run** buttons.


FirstProgram - NetBeans IDE 6.0
File Edit View Navigate Source Refactor Build Run Profile Versioning Tools Window Help

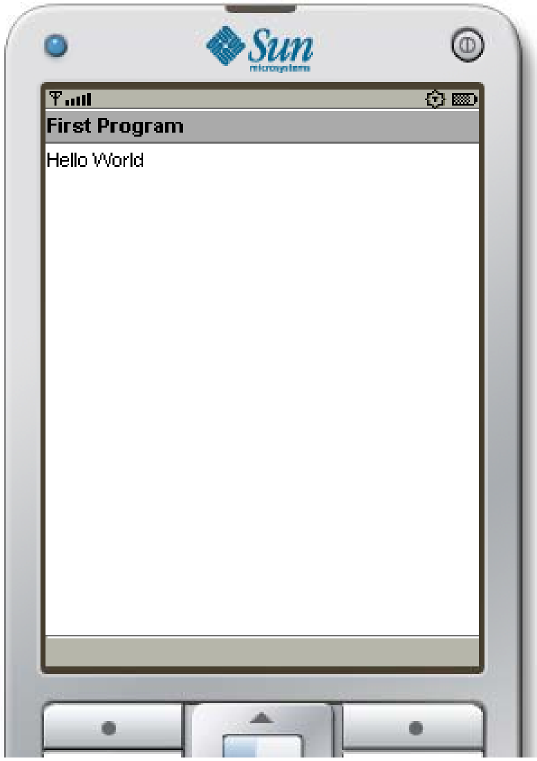
DefaultConfiguration



Select one to launch:

First Program

Launch



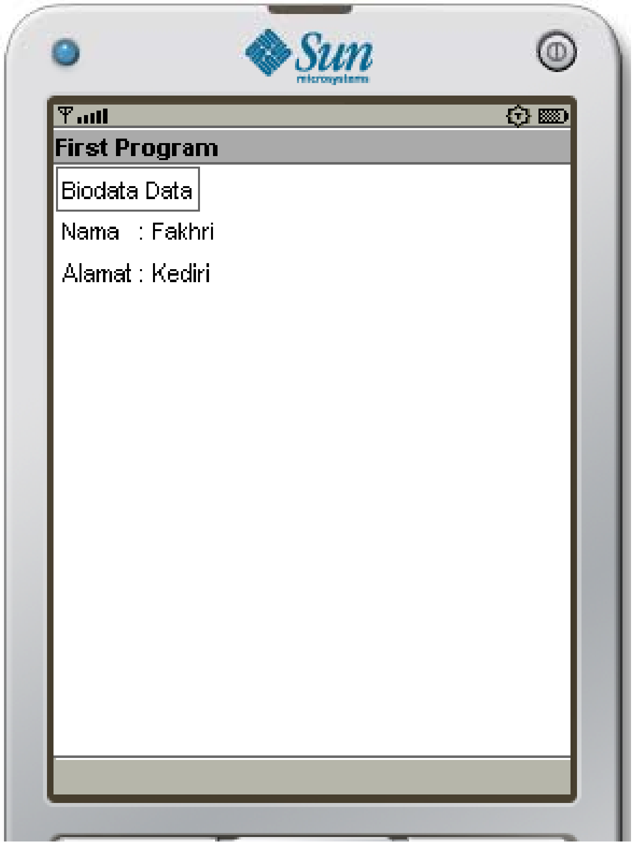
First Program

Hello World

Add another line

- To make newline use ‘\n’

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

public class biodata extends MIDlet {

    public biodata() {
    }

    public void startApp() {
        Form form = new Form( "First Program" );
        form.append("Biodata Data\n");
        form.append("Nama    : Fakhri\n");
        form.append("Alamat : Kediri\n");
        Display.getDisplay(this).setCurrent( form );
    }

    public void pauseApp() {
    }

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



First Program

Biodata Data

Nama   : Fakhri

Alamat : Kediri

Show Image

- It supports jpg, png image and save in ‘res’ folder.

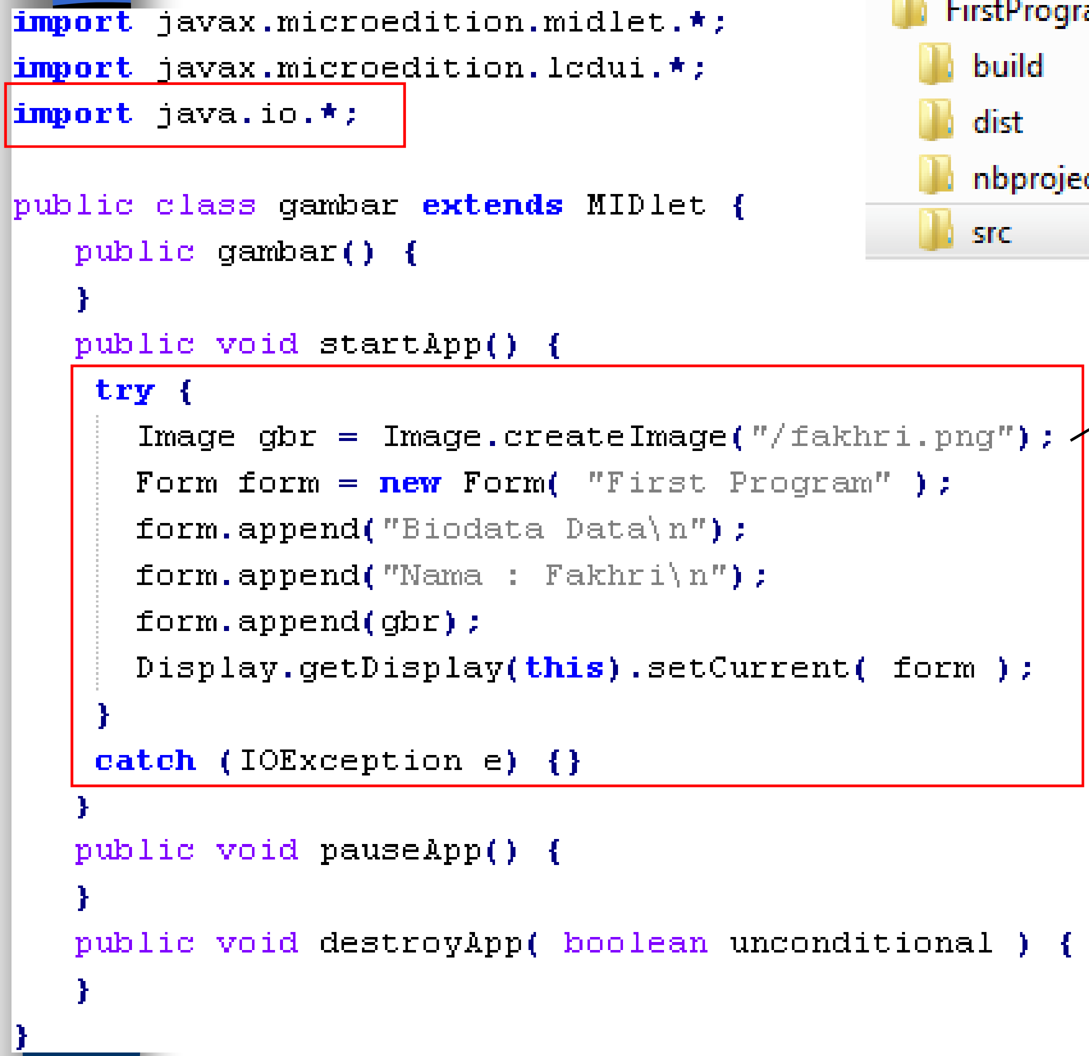
  Attention about image size, it must be enough to be displayed in LCD HP.
- Adding image, it needs 2 part i.e. :

  a. Add import : import java.io.*;

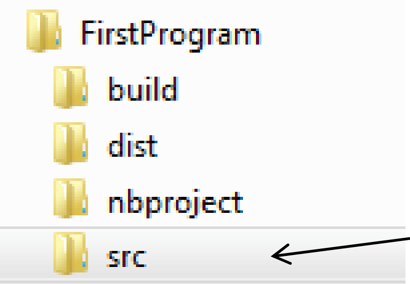
  b. Exception :

  ```
      try { }
      catch (IOException e) {}
  ```

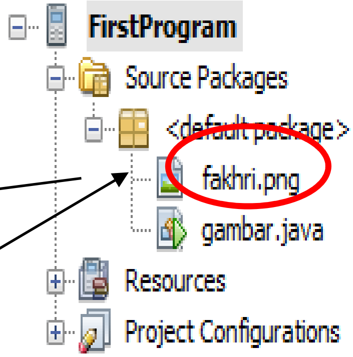
Example



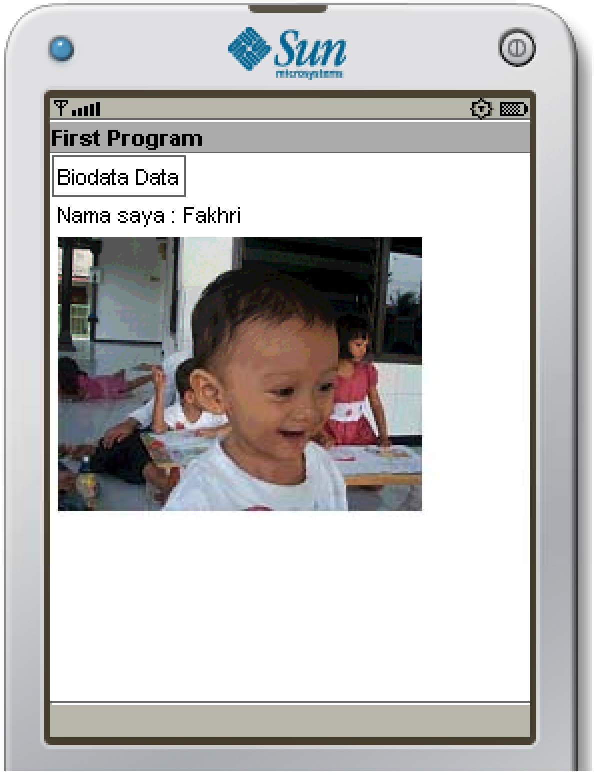
```
import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;
import java.io.*;

public class gambar extends MIDlet {
   public gambar() {
   }
   public void startApp() {
    try {
      Image gbr = Image.createImage("/fakhri.png");
      Form form = new Form( "First Program" );
      form.append("Biodata Data\n");
      form.append("Nama : Fakhri\n");
      form.append(gbr);
      Display.getDisplay(this).setCurrent( form );
    }
    catch (IOException e) {}
   }
   public void pauseApp() {
   }
   public void destroyApp( boolean unconditional ) {
   }
}
```



```
FirstProgram
  build
  dist
  nbproject
  src
```



```
FirstProgram
  Source Packages
    <default package>
      fakhri.png
      gambar.java
  Resources
  Project Configurations
```



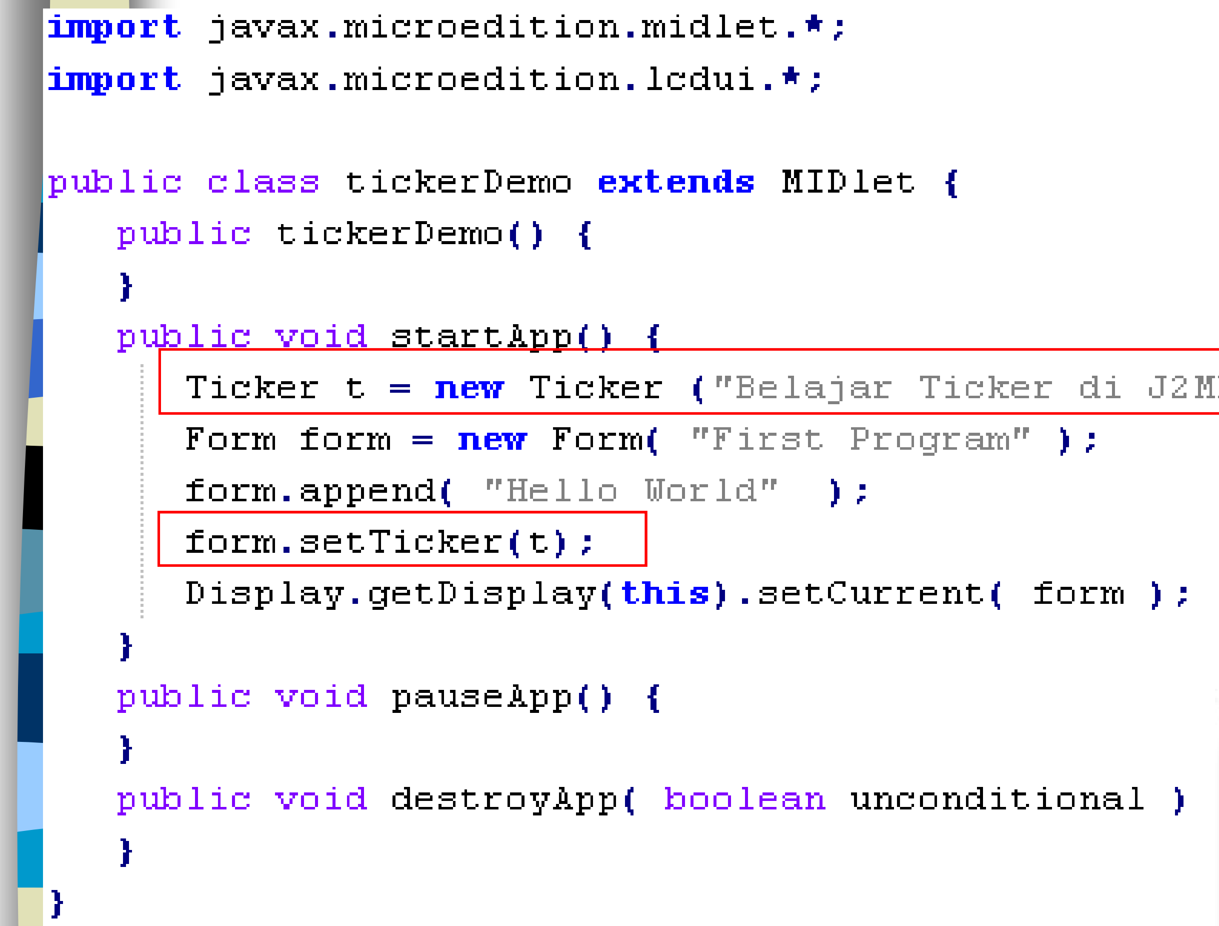
First Program

Biodata Data

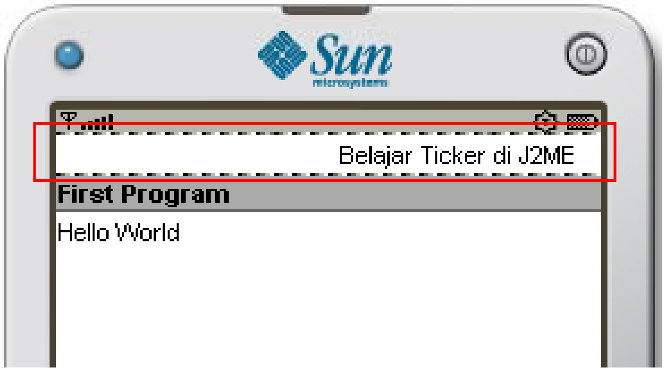
Nama saya : Fakhri

Aplikasi Ticker

- To add running text application in the HP.



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

public class tickerDemo extends MIDlet {
   public tickerDemo() {
   }
   public void startApp() {
      Ticker t = new Ticker ("Belajar Ticker di J2ME");
      Form form = new Form( "First Program" );
      form.append( "Hello World"  );
      form.setTicker(t);
      Display.getDisplay(this).setCurrent( form );
   }
   public void pauseApp() {
   }
   public void destroyApp( boolean unconditional )
   }
}
```

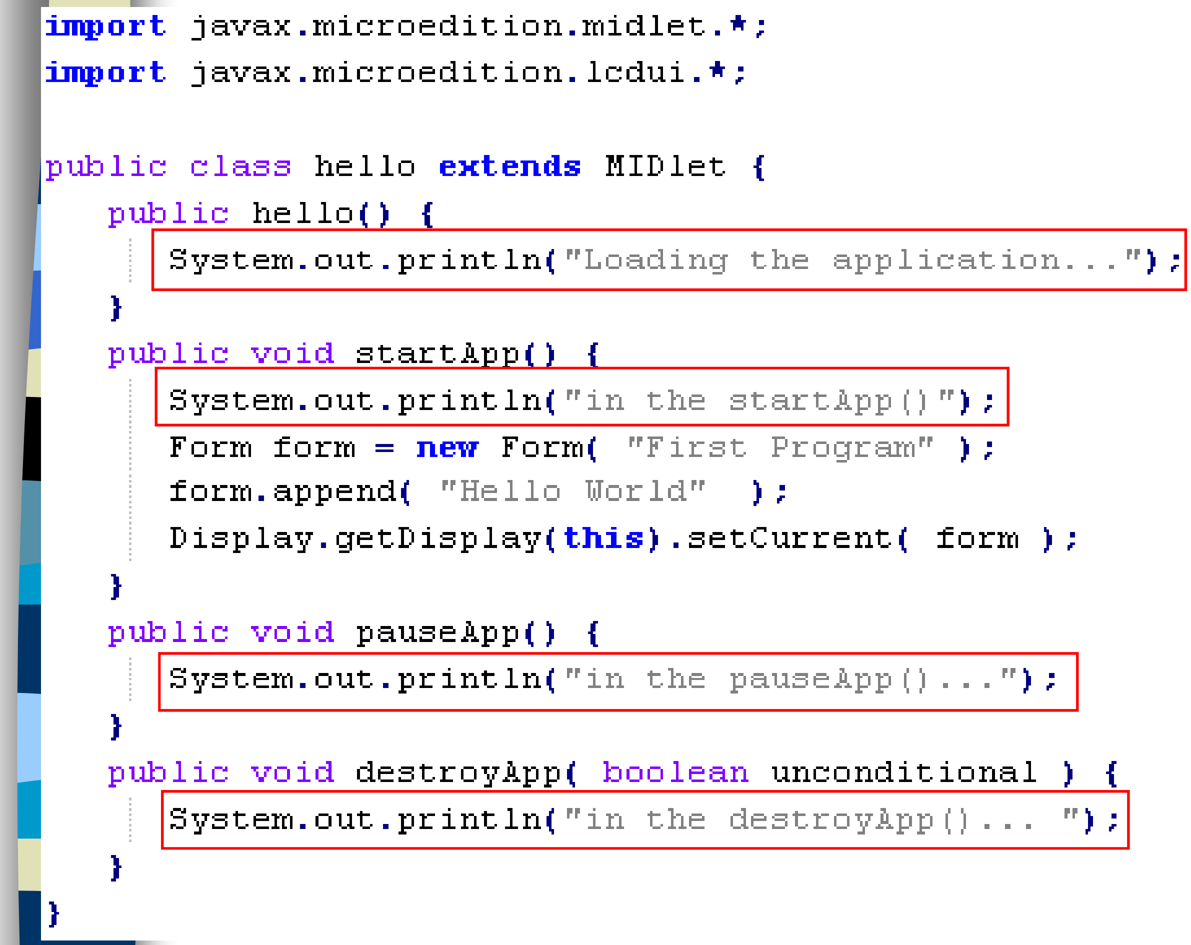


Belajar Ticker di J2ME

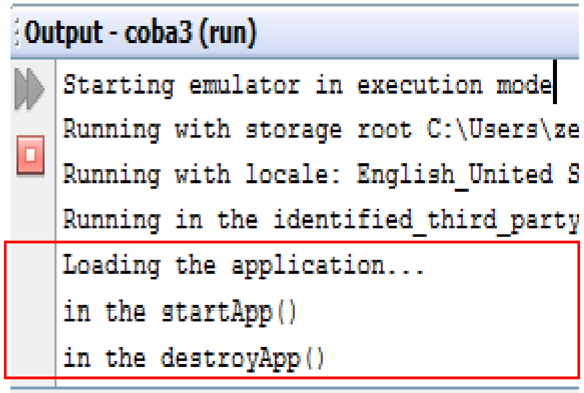
First Program

Hello World

Simple Debugging

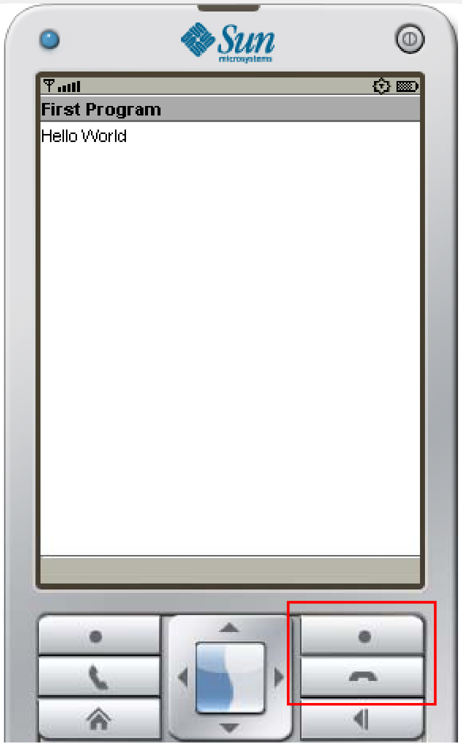


```
import javax.microedition.midlet.*;
import javax.microedition.lcdui.*;

public class hello extends MIDlet {
    public hello() {
        System.out.println("Loading the application...");
    }
    public void startApp() {
        System.out.println("in the startApp()");
        Form form = new Form( "First Program" );
        form.append( "Hello World"  );
        Display.getDisplay(this).setCurrent( form );
    }
    public void pauseApp() {
        System.out.println("in the pauseApp()...");
    }
    public void destroyApp( boolean unconditional ) {
        System.out.println("in the destroyApp()... ");
    }
}
```

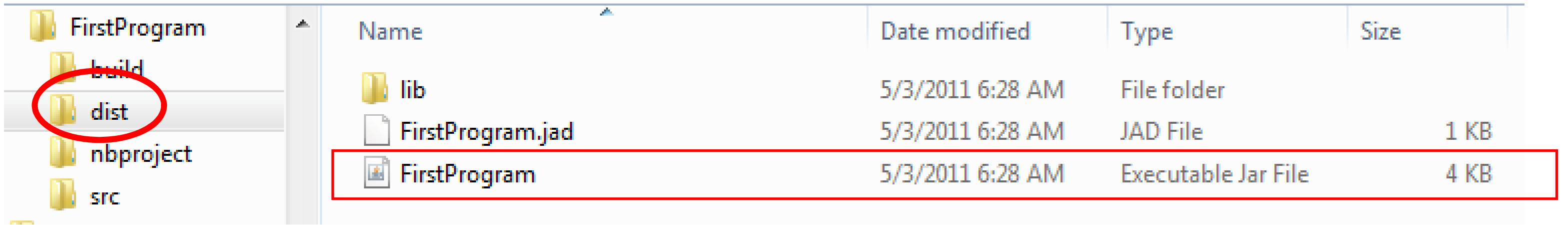


Output - coba3 (run)

```
Starting emulator in execution mode
Running with storage root C:\Users\ze
Running with locale: English_United S
Running in the identified_third_party
Loading the application...
in the startApp()
in the destroyApp()
```



How to download in to the HP

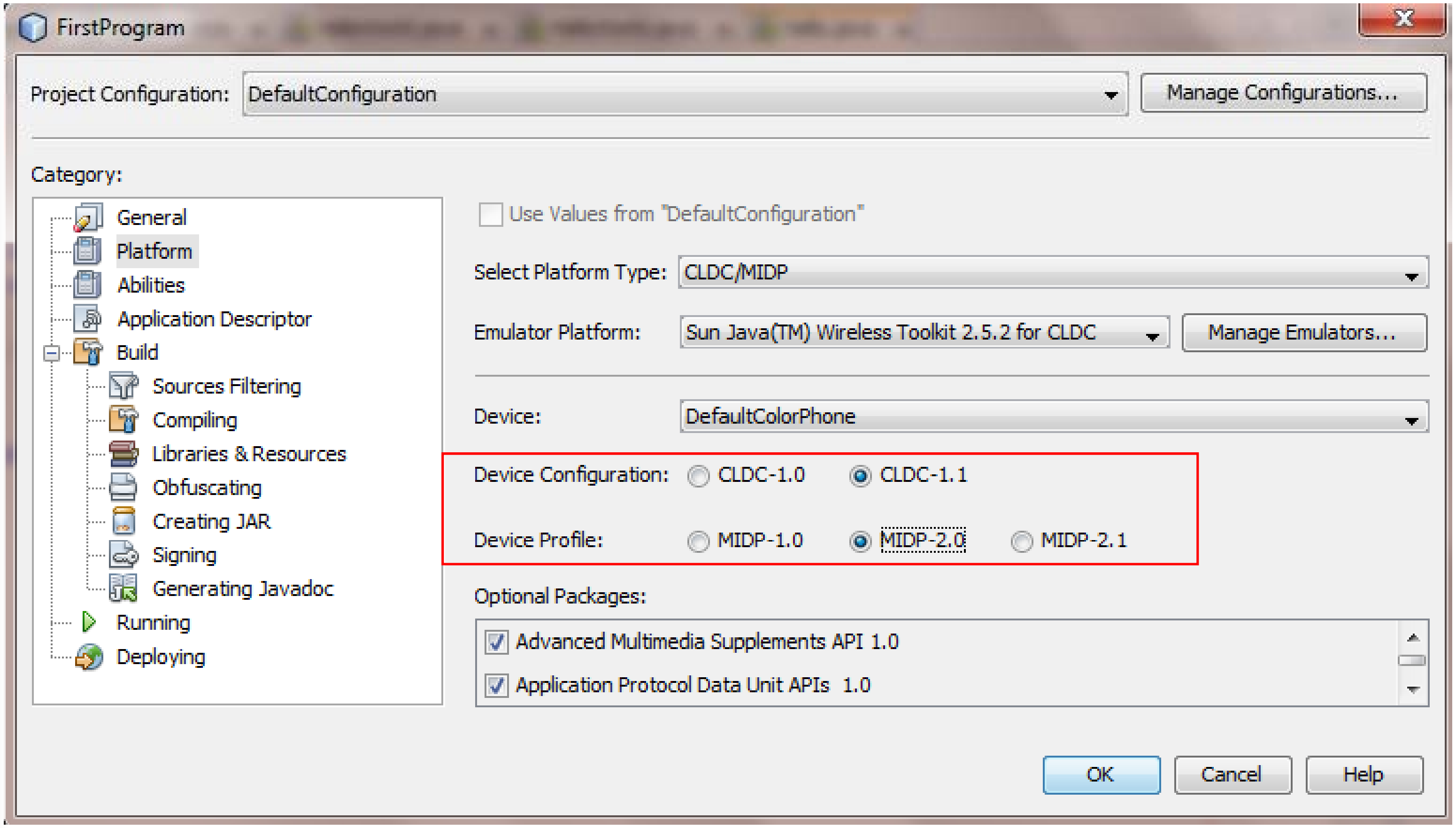


- FirstProgram
  - build
  - dist
  - nbproject
  - src

Name	Date modified	Type	Size
lib	5/3/2011 6:28 AM	File folder	
FirstProgram.jad	5/3/2011 6:28 AM	JAD File	1 KB
FirstProgram	5/3/2011 6:28 AM	Executable Jar File	4 KB

- After click Run button, automatically it will create jar file in the folder dist.
- Only jar file will be downloaded in to the HP

How to download in to the HP



FirstProgram

Project Configuration: DefaultConfiguration

Category:

- General
- Platform
- Abilities
- Application Descriptor
- Build
  - Sources Filtering
  - Compiling
  - Libraries & Resources
  - Obfuscating
  - Creating JAR
  - Signing
  - Generating Javadoc
- Running
- Deploying

□ Use Values from "DefaultConfiguration"

Select Platform Type: CLDC/MIDP

Emulator Platform: Sun Java(TM) Wireless Toolkit 2.5.2 for CLDC

Device: DefaultColorPhone

Device Configuration: ○ CLDC-1.0 ● CLDC-1.1

Device Profile: ○ MIDP-1.0 ● MIDP-2.0 ○ MIDP-2.1

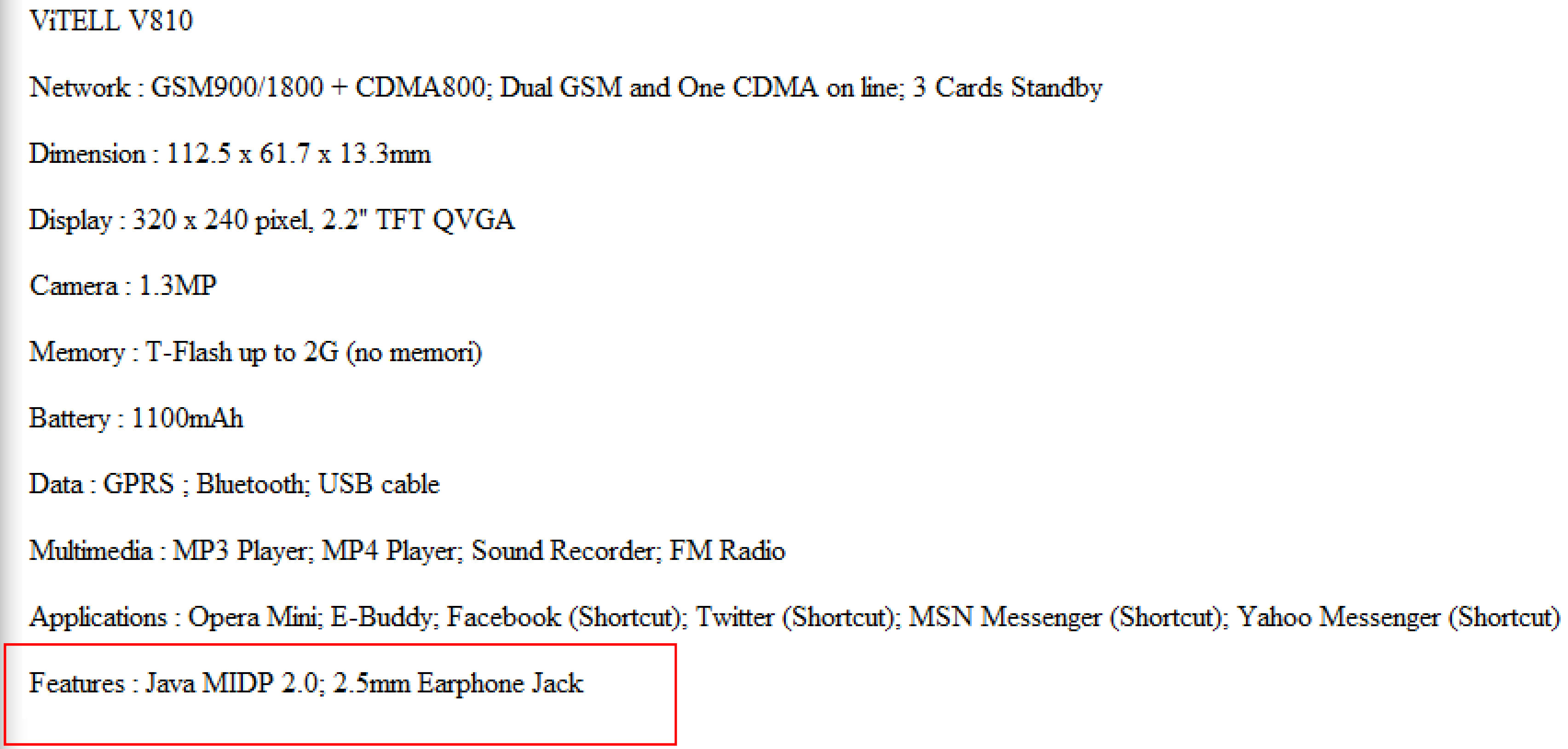
Optional Packages:

☑ Advanced Multimedia Supplements API 1.0

☑ Application Protocol Data Unit APIs 1.0

It must be same with the Java-supported in the HP.

HP Specification



ViTELL V810

Network : GSM900/1800 + CDMA800; Dual GSM and One CDMA on line; 3 Cards Standby

Dimension : 112.5 x 61.7 x 13.3mm

Display : 320 x 240 pixel, 2.2" TFT QVGA

Camera : 1.3MP

Memory : T-Flash up to 2G (no memori)

Battery : 1100mAh

Data : GPRS ; Bluetooth; USB cable

Multimedia : MP3 Player; MP4 Player; Sound Recorder; FM Radio

Applications : Opera Mini; E-Buddy; Facebook (Shortcut); Twitter (Shortcut); MSN Messenger (Shortcut); Yahoo Messenger (Shortcut)

Features : Java MIDP 2.0; 2.5mm Earphone Jack