

OISA	JAVA部会	適用	Java2	作成者	中村	作成日	2000/12/14	頁
iMode JavaとMIDP								1/5

・MIDP(Java2 Micro Edition MIDP)についての技術的詳細説明

MIDPについては、別紙(Java2について)を参照してください。 J-PHONEでは基本的にMIDPをそのまま使用するようです。

トピックス	詳細	補足
MIDPのAPIマニュアル	<ul style="list-style-type: none"> MIDP(ver1.0)ではAPIマニュアルは提供されていませんが、以下の方法で、APIマニュアルを入手できます。 <ol style="list-style-type: none"> MIDPで提供されているソースファイルから Javadocにより、自分で作成する。 Java2 Micro Edition Wireless Toolkit を入手する。 doc¥api にAPI マニュアルがあります。 	MIDPでは、API マニュアルに記載されているAPIしか使用できません。 J2SEのパッケージの内、 <ol style="list-style-type: none"> java.io java.lang java.util が使用できますが、使用できるクラスやメソッドには制限があります。
MIDPの特殊パッケージ	<ul style="list-style-type: none"> MIDP(ver1.0)では、組込機器用に特殊パッケージが追加されています。 	MIDPの特殊パッケージ <ol style="list-style-type: none"> javax.microedition.lcdui javax.microedition.rms javax.microedition.midlet javax.microedition.io
MIDPの言語制限	<ul style="list-style-type: none"> 通常のJava2にはない制限事項があります。 <ol style="list-style-type: none"> 浮動小数点が使えない SecurityManager クラスがない 	(1)浮動小数点(float,double)を使用するメソッドもありません。 (2)セキュリティ制限がない...と思われる??
アップロード方法	(1)jar ファイルを作成 <ol style="list-style-type: none"> MIDlet 関連ファイル(画像ファイル等)・jad ファイルを一つの jar ファイルに圧縮 (2)jar ファイルと jad ファイルをネットワーク上にアップロード	(圧縮方法) jar cvmf [jadファイル名] [jarファイル名] *.class *.png (注意) (1)jarファイルのアップロード先はjadファイルで記述した位置でなければならない。 (2)jarファイル中のjadファイルとアップロードしたjadファイルは同一でなければならない。
	注意 ダウンロード可能な MIDlet を作るには jad ファイルをが必要です。 (jadファイル) MIDlet の内容を記述したテキスト形式のファイル	
実行方法(ネットワーク経由)	(1)ネットワークからダウンロードして実行する。(アドレスを指定して実行する) 例) midp -transient http://sample.co.jp/OISA/Java/sample1.jad (2)ネットワークからダウンロード・インストールして実行する。 例) midp -install http://sample.co.jp/OISA/Java/sample2.jad	(インストール後) (1)ネットワークを切った後の実行 例) midp -run sample2 (2)アンインストール 例) midp -remove Tunnel (3)インストール内容の確認 例) midp -list
	内部処理 <ol style="list-style-type: none"> jad ファイルをダウンロード ダウンロード対象の MIDlet 情報をチェック (実行可能かどうか(MIDPのバージョン・ファイルサイズ等)) jar ファイルのダウンロード 	

OISA	JAVA部会	適用	Java 2	作成者	中村	作成日	2000/12/14	頁
iMode JavaとMIDP								2/5

プログラミング	・ MIDlet の基本的なプログラミング方法を説明します。	
	基本	MIDP のプログラムは MIDlet を継承して作成します。
	注意	(1)public void startApp(); (2)public void pauseApp(); (3)ublic void destroyApp(boolean unicoditional); 上記3メソッドはオーバーライド必須です。
	インターフェイス	Javax.microedition.lcdui パッケージを使用します。
	補足	Awt はありません。
	画面の表現	Display オブジェクトを使います。
	PGM	Display display = Display.getDisplay(midlet);
	表現方法	・ Dsplayable オブジェクトで表現します。 Displayable オブジェクトが表示されている時に使用可能なメニューボタンを Command オブジェクトと、CommandListener オブジェクトで作成できます。
	PGM	Displayable txt = new TextBox("Hello OISA", "OISA Java Working Group", 256, 0); Command exit=new Command("Exit",Command.SCREEN,2); txt.addCommand(exit); txt.setCommandListener(listener);
	補足	Dsplayable クラスには(1)Canvas, (2)Screen という2つのサブクラスが存在します。
Canvas クラス	Awt の Canvas クラスに似ていますが、ゲーム用デバイスの使用も想定しているようです。	(画像) Image オブジェクト (文字) String オブジェクト (フォーム) Item オブジェクト
Screen クラス	(1)Form というサブクラスが存在します。	
Form クラス	携帯で Web を表示している画面に相当し、Form オブジェクトに 画像・文字・フォームといったオブジェクトを追加する事で 並べて表示出来ます。	

OISA	JAVA部会	適用	Java 2	作成者	中村	作成日	2000/12/14	頁
i Mode JavaとMIDP								3/5

i-mode で動く Java についての技術的詳細説明 (2000/12/15 段階)

- (1) i モードで動く Java の正式名称は、Java2 Micro Edition CLDC i モード対応 Java プロファイル (以下、i-Mode Java) といいます。
(2) i-mode Java で作成されたアプリケーションは、i モード Java アプリケーション といいます。

i - mode Java が動作する機種	
機種	補足
503 i シリーズ	(1) ダウンロードして iJava メニューに登録する形で実行される。 (2) 保存した java プログラムは iJava メニュー (i MENU に類似) から選ぶ。
504 i シリーズ	(1) web ページに組み込まれた形 (アプレット形式) でも実行できる。
次世代	W-CDMA の次世代携帯でもサポートされる。

i - mode Java の特徴		
特徴	補足	備考
ファイルサイズに限界がある	jar 圧縮で最大 10K 迄	
スクラッチパッドが存在する	一時的にデータ (画像も) を保存する領域が存在	(1) 最低 5K を保証 (2) 最低 3 領域を保証
プログラムが保存可能	jar ファイルを保存できる	最低 3 つを保証
ファイルタイプが増える	GIF ファイルと i-メロディファイルが使用できる	
アクセス制限	携帯内部のアドレス帳等の個人情報にはアクセスできない	
接続制限	ダウンロード元サイト以外とは通信できない	

開発環境		
i - Mode Java の開発に最低限必要な環境	(1) Java SDK Standard Edition ver1.3 (2) Java2ME CLDC (3) i モード Java 拡張 API クラスライブラリ (4) i モード Java 文字コンバータ	(3) と (4) は DoCoMo から別途公開されるようです。
まだ、開発環境は (i モードコンテンツプロバイダ以外には) 公開されていないのですが、2000/12/15 現在、入手している情報では上記のようになっているようです。		

OISA	JAVA部会	適用	Java 2	作成者	中村	作成日	2000/12/14	頁
iModeJavaとMIDP								4/5

i-mode Java の構成要素		
Java2ME CLDC の API	必要最小限の API が使用可能	
i-mode 用 API	i モード対応 Java プロファイルで定義されている API	Com.nttdocomo.* パッケージ 音声・バリエーションといった機能は、拡張機能用 API に含まれる為、使えない可能性あり
拡張 API	機種毎に異なる拡張機能用 API	

トピックス	詳細	補足
iModeJava の API マニュアル		
iModeJava の 特殊パッケージ	・ i モード対応 Java プロファイルで定義されています。	I ModeJava の特殊パッケージ (1) com.nttdocomo.*
iModeJava の言語制限	・ MIDP と同様、通常の Java2 にはない制限事項があります。 (1) 浮動小数点が使えない (2) スレッドグループ・ファイナライズがサポートされない	(1)浮動小数点(float,double)を使用するメソッドもありません。
アップロード方法	(1)jar ファイルを作成 (a) i アプリケーションと関連ファイル(画像ファイル等)を jar ファイルに圧縮 (2) jar ファイルに関する情報を記述した ADF (jam ファイル) を作る。 (3)jar ファイルと jam ファイルを web サーバー上にアップロード。 (4)HTML にダウンロード用のタグを記述。	(HTML のダウンロード用タグ記述例) <OBJECT DECLARE ID="SampleOISA" DATA="http://OISA/Java/sample/test.jam" TYPE="application/x-jam"> サンプルプログラム </OBJECT>
	注意 ダウンロード可能な i アプリケーションを作るには jam ファイルが必要です。 (jam ファイル) jar ファイルの置位置 (URL) やサイズを記述したテキスト形式のファイル	 ・ OBJECT タグで jam ファイルを指定 ・ A タグの IJAM 属性でそれを指す。
実行方法 (ネットワーク経由)	携帯で上記リンクをクリックするとダウンロードが出来る。	(i-mode Java 未対応の携帯) A タグの HREF 属性 (必須) で示されたページに飛びます。

OISA	JAVA部会	適用	Java 2	作成者	中村	作成日	2000/12/14	頁
i Mode JavaとMIDP								5/5

プログラミング	・ i モード Java アプリケーションの基本的なプログラミング方法を説明します。		
基本	i モード Java アプリケーションは com.nttdocomo.ui.IApplication クラスを継承して作成します。		
注意	(1) public void start(); メソッドはオーバーライド必須。 (2) i アプリケーションの終了には IApplication オブジェクトの terminate() メソッドを使用。 (3) i アプリケーションから別の i アプリケーションの起動は不可。		
インターフェイス	com.nttdocomo.ui パッケージを使用します。		
補足	Awt はありません。		
画面の表現	Display クラスで表現します。		
補足	Display クラスのメソッドは全て static 宣言されています。		
表現方法	Frame オブジェクトで表現します。 Frame のサブクラスから表示するオブジェクトを 1 つ選択し、それを Display クラスに設定して使用します		
PGM	Frame panel=new Panel(); Display.setCurrent(panel);		
補足	Frame クラスには (1)Canvas, (2)Panel, (3)Dialog という 3 つのサブクラスが存在します。		
Canvas クラス	awt の Canvas クラスと似ており、継承して使用します。 イベントは processEvent(int type, int param)をオーバーライドして使用します。		
Panel クラス	各種コンポーネント (ボタンやラベル等) が追加できる 高レベル API です。		
Dialog クラス	ダイアログです。		
<p>実機で確認していないので誤った記述があるかもしれません。 上記内容は、2000/12/15 段階でわかる限りの内容です。</p>			

MIDP , i モード Java アプリケーション共通	
画像ファイル	現在、MDIP では「PNG」形式の画像フォーマットのみ使用可能ですが、i モード Java では「GIF」形式の画像フォーマットも使用可能です。 「PNG」は 1996 年に W3C (World Wide Web Consortium: WWW の普及および標準化を推進する団体) で、Web ページで使える画像として正式採用されたフォーマットです。