

Javaルキーでも大丈夫!

勝手に作ろう!

iアプリ

第 ソフトキーでiアプリを動かそう

3 回

木寺祥友 + 株式会社エル・カミノ・リアル
www.ecreal.co.jp

iアプリを操作する基本 ソフトキーを使おう

前回まで文字を表示させたり、絵を表示させたりすることを中心に解説してきたが、今回はボタン操作、とくにソフトキーを使ってiアプリを制御する方法について解説する。

本題に入る前にまずソフトキーについて説明しよう。ソフトキーとは503iシリーズから採用された、画面の下あたりに左右に配置してある2つのボタンだ。iアプリの動作時には、アプリケーションの機能が割り振られた、いわばパソコンのファンクションキーのようなものになる。右の写真を見て欲しい。画面の下部に「終了」「更新」という文字が見えるが、これがそれぞれ左右のソフトキーに割り振られた機能だ。この場合、左のソフトキーを押せばアプリが終了し、右のを押せばデータが更新されるというわけだ。今回は、このようにそれぞれのキーに機能を割り振る方法と、写真のようにその機能を画面上で表示する方

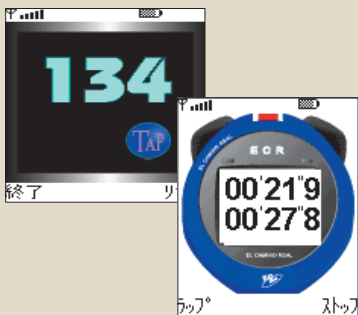
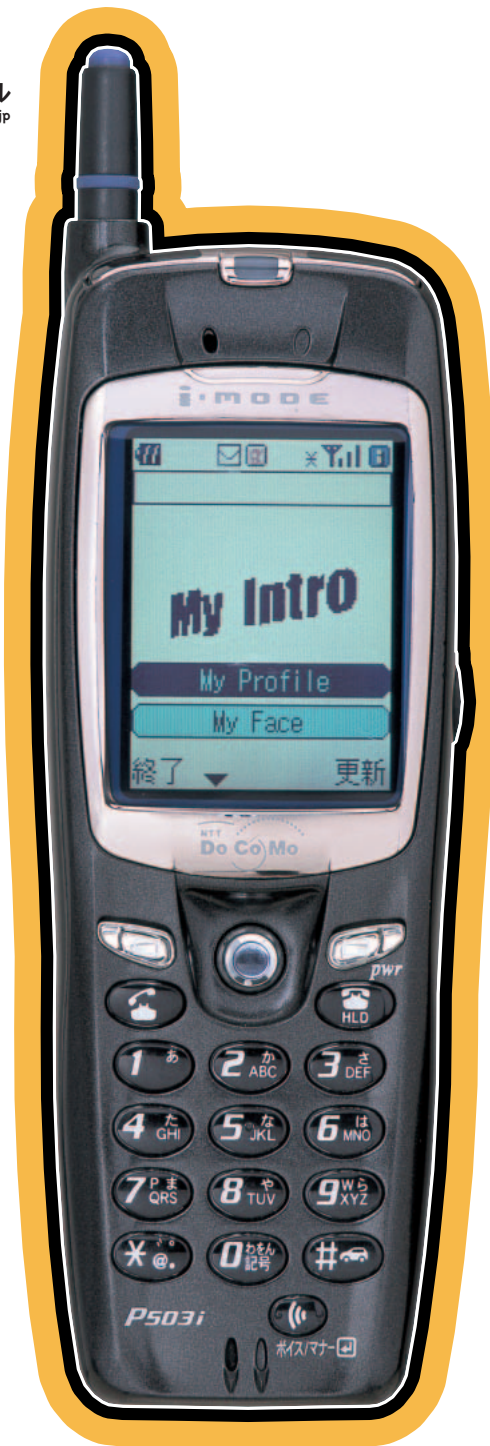
法を解説する。

もちろんソフトキー以外のボタン、たとえば*キーや#キーをうまく使うことによってソフトキーのような使い方もできるのだが、ソフトキー以外のボタンは、パネルとキャンバスで操作できるボタンが違ったり、機種依存が激しく、方向キーを取らない機種があったりして難しくなってしまうため、今回はあくわしくは解説しない。もしほかのボタンを使いたい場合は、専門の書籍を買うなどしてチャレンジしてもらいたい。

それでは、パネルとキャンバスのそれぞれのクラスにおいてどのようにソフトキーの設定するかを解説しよう。

ここで取り上げているiアプリ「MyIntro」はインターネットマガジンのウェブサイト*Jump*にある。参考にしてもらいたい。

internet.impress.co.jp/iappli/

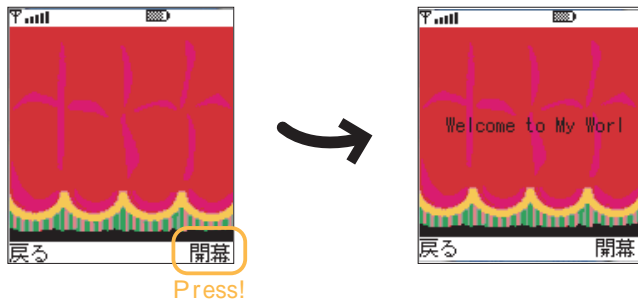


ソフトキーと前回解説したタイマーイベントを組み合わせると、時間をうまくつかったアプリケーションを作れる。ここではその一例を紹介しよう。

DJタップ：曲のBPMを計る計測用のiアプリだ。曲に合わせてキーを押すことによってリアルタイムにBPMを表示してくれる。DJをするときなどに大変便利で、曲のBPMをこれで計れば次のつなぎもスムーズにできる。

ストップウォッチ：専用デバイスではないが、ラップタイムにも対応している本格的なストップウォッチだ。駐車場のパーキングの目安などにも使える。

キャンバスでの ソフトキーの入力処理



processEventメソッド【137-178行目】

processEventメソッドは、キャンバスを使う場合に必ず使用する非常に大切なメソッドだ。なぜなら、端末キーを押したときに限らず、すべてのイベントはこのメソッドで処理するからだ。

```
public void processEvent ( int type, int param )
```

このメソッドは、type と param の2つの引数がある。それぞれ、

type : イベントのタイプ

param : タイプに応じたパラメーター

が入っている。processEventメソッドでは、この2つの引数から条件文を使ってイベントの内容を判断し、処理を記述するのだ。

ソフトキー 1の入力を処理する【154-159行目】

実際にソフトキー1が押されたときの処理の仕方を見てみよう。155行目を見ると、if文の条件が、

```
type == Display.KEY_PRESSED_EVENT ... 「キーが押された」  
param == Display.KEY_SOFT1 ... 「ソフトキー1が」
```

の両方を満たす場合となっている。これで「ソフトキー1が押された場合」を表しているため、この中にソフトキー1を押したときに行いたい処理を記述すればよい。ソフトキー1に限らず、キーの入力によるイベントでは引数 type に

キーが押された時 : Display.KEY_PRESSED_EVENT

キーが離された時 : Display.KEY_RELEASED_EVENT

が入っているため、それぞれ使い分けよう。さらに、type が上の2つの場合には引数 param に使用されたキーの種類 (Displayクラスの定数) が入っている。これを使い分ければ、キーごとにイベントを分けられる。

ソフトキーにラベルを付ける【54-56行目】

ソフトキーには、setSoftLabelメソッドを使うことで画面に表示させるラベルをつけられる。ただし、このメソッドで使うソフトキーの定数はFrame.SOFT_KEY_1 などFrameクラスの定数なので注意してほしい。

注意すべき点

このキーイベントは、残念ながらすべての端末キーが標準で使えるわけではない。標準で使えるのはソフトキー、数字キー、方向キーと選択キー、それに*と#キーだ。また、キー入力には「押されたとき」と「離されたとき」の2つのイベントがあるが、特殊な使い方をしない限りどちらか片方を使えばよい。

キャンバスを使ったiアプリでは、意外と簡単にソフトキー入力を処理できる。キャンバスが表示されている状態でキーが押されると、そのキャンバスクラスのprocessEventメソッドが呼び出されるため、処理のすべてをprocessEventメソッドに記述すればキー入力に対応できるのだ。

OpenFrm.java 54-56行目

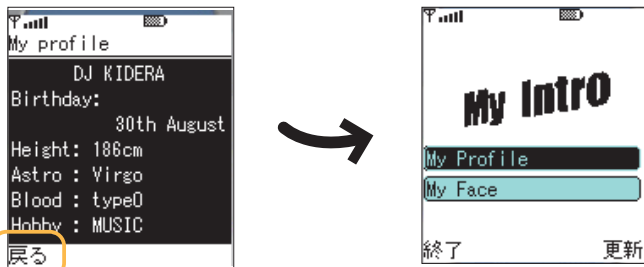
```
54 // ソフトキーにラベルをつける  
55 setSoftLabel( Frame.SOFT_KEY_1, "戻る" );  
56 setSoftLabel( Frame.SOFT_KEY_2, "開幕" );
```

OpenFrm.java 137-178行目

```
137 /**  
138 * イベント処理  
139 */  
140 public void processEvent( int type, int param )  
141 {  
142 // タイマーイベント  
143 if ( type == Display.TIMER_EXPIRED_EVENT )  
144 {  
145 // カウントを増やす  
146 tmCnt++;  
147  
148 // 開幕処理  
149 opening();  
150  
151 return;  
152 }  
153  
154 // ソフトキー1&キープレスイベント  
155 else if ( param == Display.KEY_SOFT1 && type  
== Display.KEY_PRESSED_EVENT )  
156 {  
157 // 戻る  
158 Display.setCurrent( MyIntro.topF );  
159 }  
160 // ソフトキー2&キープレスイベント  
161 else if ( param == Display.KEY_SOFT2 && type  
== Display.KEY_PRESSED_EVENT )  
162 {  
163 // 開幕  
164 // 既に開幕の場合は初期化  
165 try  
166 {  
167 st.stop();  
168 st.dispose();  
169 }  
170 catch ( Exception e ) {}  
171 // 開幕  
172 tmCnt = 0;  
173 repaint();  
174 st = ShortTimer.getShortTimer( this, 0, ITV,  
true );  
175 st.start();  
176 }  
177 }  
178 }
```

: 変数名 : コメント : 外部ファイル名やカスタマイズ可能なところ

パネルでのソフトキーの入力処理



Press!

パネルでソフトキーを使う流れ

パネルでソフトキーを使うには、ソフトキーが押されたというイベントを知らせるものと、そのことを知らされてから処理を行うメソッド(リスナー)とを用意する必要があります。また、それにはSoftKeyListenerの実装と言って、パネルクラスの宣言であらかじめSoftKeyListenerを使うことを記述しておく必要がある。それではその過程を見ていこう。

SoftKeyListenerの実装する【16行目】

SoftKeyListenerを実装するには、パネルクラスの宣言のあとで“implements SoftKeyListener”と記述すればよい。

パネルにリスナーを登録する【32行目】

ここでは、どのインスタンスにあるメソッドをリスナーにするかを設定している。ここでの引数thisとは自分自身を指す言葉であり、ProfiFrmインスタンス自身を指している。つまり、「このインスタンス内でのキーイベントに対応するリスナーは、このインスタンス内に書いておく」という意味である。なお、29行目でソフトキーラベルをセットしているが、これはキャンパスの場合と同じだ。

softKeyPressedメソッドの記述【97-108行目】

このメソッドは、「ソフトキーが押されたとき」に呼び出されるリスナーだ。つまり、このなかにソフトキーが押されたときに行う処理を記述すればよい。引数はsoftKeyだけで、ここにはFrameクラスのソフトキーを表す定数が入っている。先ほどラベルのセットに使ったものと同じだ。

softKeyReleasedメソッドの記述【110-113行目】

このメソッドは「ソフトキーが離されたとき」に呼び出されるリスナーだ。記述方法はsoftKeyPressedメソッドと同じだ。今回の例では、このメソッドには中身がなにも記述されていないが、これは「SoftKeyListenerを実装した場合は、たとえ空でもsoftKeyPressedメソッドとsoftKeyReleasedメソッドの両方をそろえないといけない」というルールがあるためだ。

ソフトキー以外のキー入力

ソフトキー以外のキーは、KeyListenerを使う。準備の流れはソフトキーの場合とほぼ同じになっている。それぞれ、setKeyListener、keyPressed、keyReleasedとメソッド名を変更すればよい。

ただし、N503iでパネルを使った場合、標準では方向キーのイベントが取れないので注意してほしい。

参考にキャンパスクラスでのイベントの種類を右の表に挙げておく。

パネルを使ったiアプリでは、キー入力の処理の仕方がキャンパスの場合とまったく異なる。パネルではイベントリスナーという考え方でキー入力を処理することになる。ここでは、実践的にソフトキーの例を使ってイベントリスナーの使い方だけに注目してみよう。

ProfiFrm.java 13-17行目

```
13 /**
14  * プロフィールパネル
15  */
16 class ProfiFrm extends Panel implements
   SoftKeyListener
17 {
```

ProfiFrm.java 23-32行目

```
23 /**
24  * 初期化を行うコンストラクタ
25  */
26 public ProfiFrm()
27 {
28     // ソフトキーにラベルをつける
29     setSoftLabel( Frame.SOFT_KEY_1, "戻る" );
30
31     // リスナーを登録する
32     setSoftKeyListener( this );
```

ProfiFrm.java 97-113行目

```
97 /**
98  * ソフトキーが押されたら時に呼ばれる
99  */
100 public void softKeyPressed( int softKey )
101 {
102     // ソフトキー-1
103     if ( Frame.SOFT_KEY_1 == softKey )
104     {
105         // 戻る
106         Display.setCurrent( MyIntro.topF );
107     }
108 }
109
110 /**
111  * ソフトキーが離された時に呼ばれる
112  */
113 public void softKeyReleased( int softKey ) {}
```

: 変数名 : コメント : 外部ファイル名やカスタマイズ可能なところ

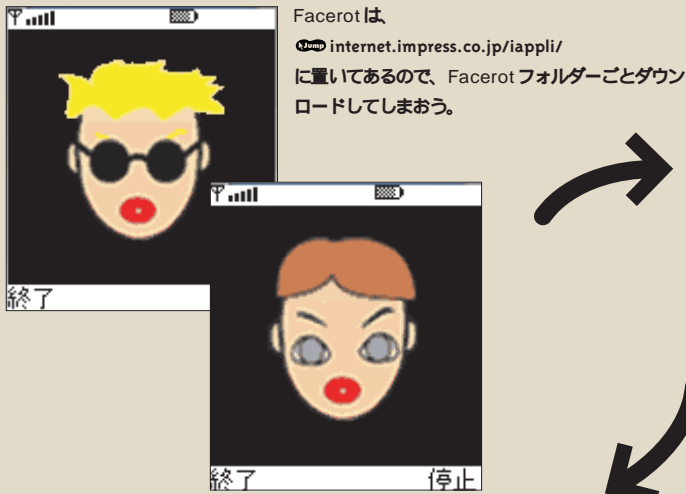
KEY_PRESSED_EVENT	キーが押されたイベント
KEY_RELEASED_EVENT	キーが離されたイベント
TIMER_EXPIRED_EVENT	タイマーイベント

3分で完成するオリジナルiアプリ

「自分の絵でスロットを作ろう」!

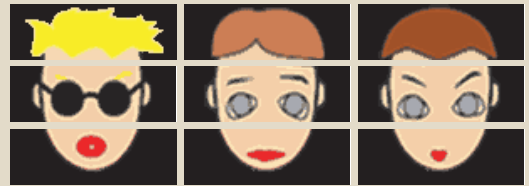
ここでは、ちょっとした変更だけで作れる自分だけのiアプリを紹介していく。
 今月は自分で用意した画像をそのままスロットにできる「Facerot」だ。

1 ダウンロードしよう



Facerotは、
internet.impress.co.jp/iappli/
 に置いてあるので、Facerot フォルダごとダウンロードしてしまおう。

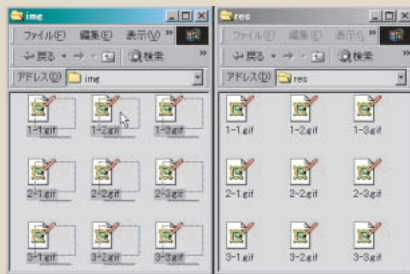
2



差し替える画像を作ろう

スロットの画像は髪と目と口、それぞれ3つずつ用意する。1つあたりの大きさは、99×33ピクセルで、全体のファイルサイズは6KB以内に収めておこう。

3 画像を差し換えよう!



それでは用意した9つの画像に差し換えてしまおう。

5 完成!



自分のホームページエリアにアップロードしよう。その際、iモードからダウンロードするときを使うhtmlファイルを忘れずに。

4 コンパイルしよう!



DoJaを使いコンパイルしよう。これで、新しい画像を使ったスロットに生まれ変わるぞ。なお、DoJaについて分からない人はDoJaのページ
www.nttdocomo.co.jp/i/java/tool.html
 を参照してほしい。

オリジナルiアプリ募集中!

今月のオマケは顔全体を統一した大きさにするのがポイントだ。でないと顔がずれてしまうぞ。うまくできたら、オリジナルiアプリをアップロードしたサイトのURLを編集部宛てにメールしよう。もしできなくても、質問でも感想でもなんでもOKだ。応募作品はWe Love Internet Peopleでも紹介するので、ドシドシ送ってきてもらいたい。宛て先は次のとおり。
im-iappli@impress.co.jp



もっと知りたいキミにはコレ!

『今すぐできるiアプリプログラミング』

小社発行 本体価格2,400円

来月なんて待ってられない、もっとくわしくいろいろ知りたい! と思っている君にはコレがオススメ。本連載の木寺氏が執筆しているため、ここで取り上げているiアプリについて、もっとくわしく解説している。一緒に読んでいけば、理解が深まるぞ。



[インターネットマガジン バックナンバーアーカイブ] ご利用上の注意

このPDFファイルは、株式会社インプレスR&D(株式会社インプレスから分割)が1994年～2006年まで発行した月刊誌『インターネットマガジン』の誌面をPDF化し、「インターネットマガジン バックナンバーアーカイブ」として以下のウェブサイト「All-in-One INTERNET magazine 2.0」で公開しているものです。

<http://i.impressRD.jp/bn>

このファイルをご利用いただくにあたり、下記の注意事項を必ずお読みください。

- 記載されている内容(技術解説、URL、団体・企業名、商品名、価格、プレゼント募集、アンケートなど)は発行当時のものです。
- 収録されている内容は著作権法上の保護を受けています。著作権はそれぞれの記事の著作者(執筆者、写真の撮影者、イラストの作成者、編集部など)が保持しています。
- 著作者から許諾が得られなかった著作物は収録されていない場合があります。
- このファイルやその内容を改変したり、商用を目的として再利用することはできません。あくまで個人や企業の非商用利用での閲覧、複製、送信に限られます。
- 収録されている内容を何らかの媒体に引用としてご利用する際は、出典として媒体名および月号、該当ページ番号、発行元(株式会社インプレス R&D)、コピーライトなどの情報をご明記ください。
- オリジナルの雑誌の発行時点では、株式会社インプレス R&D(当時は株式会社インプレス)と著作権者は内容が正確なものであるように最大限に努めましたが、すべての情報が完全に正確であることは保証できません。このファイルの内容に起因する直接のおよび間接的な損害に対して、一切の責任を負いません。お客様個人の責任においてご利用ください。

このファイルに関するお問い合わせ先

株式会社インプレスR&D

All-in-One INTERNET magazine 編集部

im-info@impress.co.jp