الفصل التاسع

إضافة صورة Image إلى الفورم Form في J2ME، ضمن High Level

تعرفنا سابقاً على كيفية التعامل مع الفورم Form وكيفية إضافة كائن من النوع TextField إلى الفورم.

سنتعرف الآن على الصنف Image وكيفية إضافتها إلى الفورم.

قبل هذا يجب علينا معرفة ثلاث أشياء:

١ - يجب علينا أولاً معرفة كيفية إنشاء صورة:

لإنشاء صورة وربطها مع التطبيق يجب علينا استخدام الطريقة cteateImage التابعة للصنف Image ، سيكون الكود المستخدم كالآتي:

CODE

```
  try{
اسم ملف          Image img = Image.createImage( /*
*/ );  الصورة مع الدليل
      } catch (java.io.IOException e)
*/ }  إظهار رسالة خطأ          { /*
```

وسنلاحظ هنا أننا استخدمنا الاستثناءات Exception ومعالجة الأخطاء try و catch.

والسبب في ذلك أن التطبيق يقوم بالبحث عن الصورة في وقت التنفيذ فإذا لم يحصل عليها فإن التطبيق يفشل إذا لم نقم بمعالجة الخطأ.

٢ - يجب علينا معرفة الأنماط التي يمكن عرضها في تطبيق J2ME:

إلى الآن يوجد نوع واحد فقط من الصور التي يمكن إضافتها إلى التطبيق هذا النوع هو:

obeikandi.com

CODE

```
PNG Image FormatPNG (Portable Network Graphics)
```

٣ - كيفية إضافة الصورة إلى الحزمة JAR باستخدام Wireless ToolKits .

يمكنك استخدام هذا الأمر:

CODE

```
*/\bin\jar.exe umf MANIFEST.MF دليل الجافا/*
MyApp.jar Image.png
```

يجب الانتباه هنا أننا استخدمنا العبارة umf.

أيضاً يستحسن أن تكون الصورة موجودة في نفس الدليل. أما إذا وضعتها في دليل آخر فإنه يجب كتابة اسم الدليل كاملاً مع ملاحظة أنه سيتم إنشاء نفس الدليل في الكلاسات المحزومة JAR.

لا يوجد أي مشيد Constructor لصنف Image مباشرة بعد تعريف كائن من نوع Image يتم استخدام createImage والسبب أن عملية إنشاء كائن Image يحدث تلقائياً عند استخدام الطريقة createImage هناك عدة طرق لاستخدام الطريقة createImage .

CODE

```
public static Image createImage(byte[]
imageData, int imageOffset, int imageLength)
```

imageData: مصفوفة تحتوي على بيانات النقاط للصورة على شكل بايتات Bytes.

مثال على ذلك عند تحويل كائن من نوع صورة إلى String باستخدام الدالة toString ومن ثم تحويلها إلى Bytes باستخدام الدالة getBytes

obeikandi.com

CODE

```
public static Image createImage(Image source)
```

source: كائن صورة موجود من قبل نريد نسخه إلى كائن آخر باستخدام هذه الطريقة.

أي أن هذه الطريقة تستخدم فقط لنسخ كائن من نوع Image إلى كائن آخر من نفس النوع.

CODE

```
public static Image createImage(int width, int height)
```

إنشاء مساحة فارغة من نمط رسومي لأجل إضافة أي رسومات أو صور لهذه المساحة:

height: ارتفاع هذه المساحة.

width: عرض هذه المساحة.

CODE

```
public static Image createImage(String name)
```

إنشاء كائن من صورة موجودة على شكل ملف محزوم مع الكلاسات المحزومة في الملف jar.

ويجب علينا كتابة اسم الملف مع الدليل الموجود فيه ملف الصورة وهذه الطريقة هي المستخدمة بكثرة.

obeikandl.com

والآن سنضع مثالاً يوضح كل هذا:

CODE

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

public class MyForm extends MIDlet  implements
CommandListener
 {
  Display display;
  Form frm;
  Command exitCommand;

 public MyForm()
  {
    frm = new Form("Form Title");
    display = Display.getDisplay(this);
    exitCommand = new Command("Exit",
Command.EXIT, 1);
  }
 public void startApp()
  {
    frm.append("This text in Form");
    addImageToForm();
    frm.append("\nwww.arabteam2000.com");
    frm.addCommand(exitCommand);
    frm.setCommandListener(this);
    display.setCurrent(frm);
  }

  public void  pauseApp() { }
```

obeikandl.com

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

 public void addImageToForm()
  {
   try{
       Image img = Image.createImage(
                                        "/Test.png");  //
اسم ملف الصورة مع الدليل
        frm.append(img);
      } catch (java.io.IOException e)
        {
          Alert alert = new Alert ( "Error: Load
Image" );
          alert.setString( e.getMessage() );
          display.setCurrent(alert);
         }
  }

   public void commandAction(Command c,
Displayable s)
      {
if (c == exitCommand)
          {
  destroyApp(false);
  notifyDestroyed();
}
       }
}
```

obeikandl.com

obeikandl.com