من اعداد مصعب غيلان - جمهورية اليمن العربية- بالاعتماد على الكتاب التالي

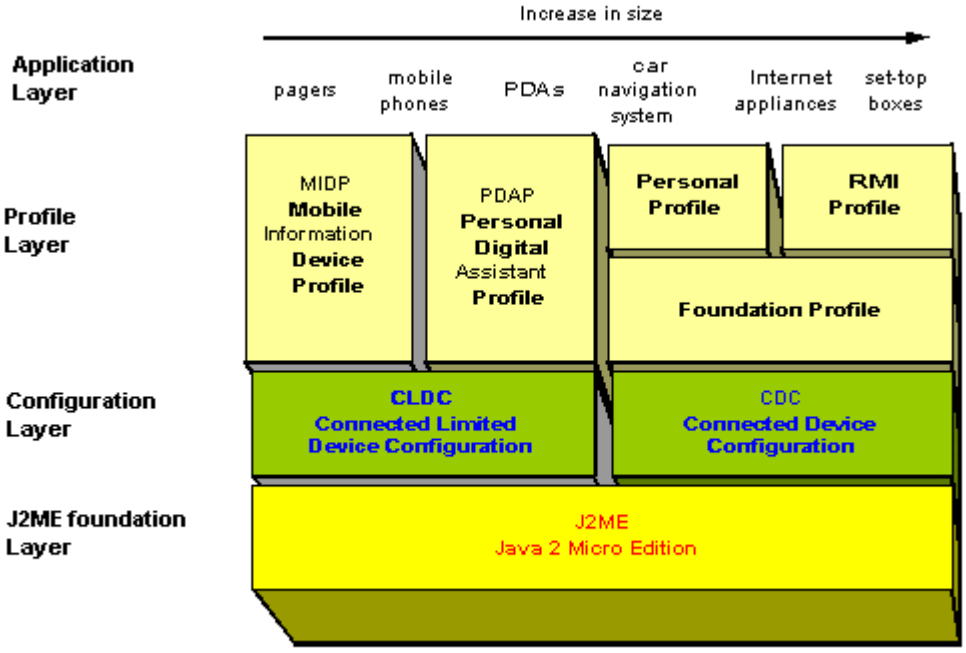
Learning Wireless for Qusay Moh

J2ME-1 معلومات عامه حول

small device هي احدى تقنيات الجافا المستخدمة لعمل تطبيقات للاجهزة الصغيرة J2ME يمكننا القول ان
اضيفت هذة التقنية الى الجافا مؤخرا تقريبا في عام 1999
J2ME وشهدت هذة التقنية تطورا كبيرا في الاونة الاخيرة مع تطور الاجهزة التي تدعم

وبهذا يمكننا تقسيم الجافا الى ثلاثة اقسام
تستخدم لبناء التطبيقات الكبيرة J2EE
تستخدم لبناء التطبيقات التجارية المتوسطة ومنها التطبيقات التي تعمل على اجهزة الحاسوب المكتبية J2SE
تستخدم لبناء التطبيقات على الاجهزة الصغيرة مثل الموبايلات واجهزة البوكت وغيرها J2ME

الى طبقات J2ME ويتم تقسيم



J2ME البرمجيات المطلوبة للبدء ببرمجة-2

http://www.arabteam2000-forum.com/index.php?showforum=120

J2ME, التطبيق الاول باستخدام -3

خطوات بناء التطبيق

J2EE و J2SE هي القسم الثالث من اقسام الجافا بعد J2ME عرفنا فيما سبق ان
Small device يعمل على الاجهزة الصغيرة J2ME وهذا القسم

والان سنتعرف على المكتبات الخاصة بهذة اللغة وكذلك انواع المعطيات المستخدمة و هيكلية بناء البرنامج ثم ننتقل الى بناء
J2ME التطبيق الاول باستخدام

: هي J2ME المكتبات الاساسية الخاصة بالـ *
تزودنا هذة المكتبة بالعديد من واجهات المستخدم مثل النوافذ والازرار وصناديق النصوص javax.microedition.lcdui
تزودنا هذة المكتبة بالطرق المستخدمة لخزن البيانات على وحدة الخزن الرئيسية للموبايل javax.microedition.rms
وكذلك بالبيئة التي سيعمل فيها التطبيق MIDP تزودنا هذة المكتبة بال javax.microedition.midlet
GenericConnection framework تزودنا هذة المكتبة بالمناهج الخاصة بعمليات الاتصال javax.microedition.io
GCF
تزودنا هذة المكتبة بالمناهج الخاصة بعمليات الادخال والاخراج java.io
وانواع المعطيات J2SE تحتوي على المناهج الخاصة بال :java.lang
تحتوي على المناهج الملحقة java.util

كما ان هناك العديد من المكتبات الاخرى التي تعمل على اجهزة معينة فقط
J2ME بينما هذة المكتبات هي المكتبات الاساسية الخاصة بال

J2ME انواع المعطيات في *
java.lang.Boolean
java.lang.Byte
java.lang.Character
java.lang.Integer
java.lang.Long
java.lang.Short

float , double لا تدعم الاعداد ذو النقطة العائمة J2ME ونلاحظ هنا ان
وذلك لان النقطة العائمة تحتاج الى عتاد كبير وبرمجيات ذو تكلفة عالية
بناء البرنامج الاول *
عند قراءة هذا الموضوع يفترض ان يكون عندك معلومات لا بأس بها عن البرمجة الكائنية

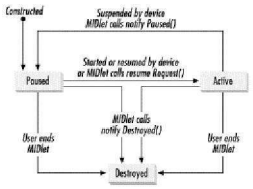
main في برنامج الجافا او السي مثلا نرى ان بداية تنفيذ الكود يبدأ من الدالة
وذلك لسبب بسيط وهو startApp من الدالة J2ME فإن الامر مختلف قليلا وسيبدأ تنفيذ البرنامج في J2ME اما بالنسبة لل

لنشير للمترجم ان Main انه عندما نقوم بتحديد برنامج ما هو البرنامج الرئيسي في الجافا او السي يكفي ان نقوم بكتابة الدالة يبدأ من هنا
javax.microedition.midlet الموجود في الحزمة MIDlet فاننا نقوم بانشاء منهج مشتق من المنهج J2ME اما في

destroyApp , pauseApp , startApp الثلاث التالية abstract Methods ويوجد في هذا المنهج الطرق المجردة
ومن اساسيات البرمجة الكائنية ان الطرق المجردة في صنف مشتق منه يجب ذكرها كاملة

وسيبدأ تنفيذ البرنامج من الدالة MIDlet ولهذا فإن البرنامج الرئيسي سيكون هو البرنامج الذي سيتم اشتقاقه من الصنف
startApp

destroyApp , pauseApp ايظا هناك الطريقتان
يتم تنفيذ هذة الدالة عند انتقال التطبيق من الحالة النشطة الى الحالة الغير نشطة pauseApp
يتم تنفيذ هذة الدالة عند اغلاق التطبيق destroyApp
MIDlet دورة حياة البرنامج الرئيسي *



وكما هو واضح من الرسم نجد ان التطبيق يمر بثلاثة مراحل او حالات
الحالة الابتدائية النشطة ثم الحالة غير النشطة (التوقف المؤقت) ثم حالة الاغلاق وانهاء التطبيق

pauseApp لا تهتم حاليا بالحالة
startApp , destroyApp ولكن سنركز على الحالتين

والذي يحمل نفس اسم Constructor يجب ملاحظة انه عند تنفيذ اي تطبيق فإنه سيقوم اولا بتنفيذ الكود الموجود في المشيد المنهج
startApp وبعد ذلك سينتقل التنفيذ الى الدالة
والان سنقوم بكتابة المثال التالي *
HelloMidlet بكتابة هذا الكود مع ملاحظة انه يجب حفظ هذا الملف بنفس اسم المنهج

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

import javax.microedition.lcdui.*;


public class HelloMidlet extends MIDlet
{

private Display display;


TextBox box = null;

public void HelloMidlet(){}

public void startApp(){


display =Display.getDisplay(this);

box =new TextBox("Simple Example ","Hello ",20,0);

display.setCurrent(box);
}

public void pauseApp(){ }

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

مرحلة فهم الكود
import javax.microedition.midlet.*;
javax.microedition.midlet هنا قمنا باخبار المترجم باننا سوف نستخدم جميع الكلاسات المضمنه في الدليل

وهذا هو معنى النجمة
معناها جميع الكلاسات الموجودة في دليل معين
import javax.microedition.lcdui.*;
هنا ايظا نفس الحكاية السابقة
من اجل تضمين الكلاسات
Display و TextBox

public class HelloMidlet extends MIDlet
هنا قمنا بانشاء البرنامج الرئيسي الذي هو عبارة عن كلاس
ولكن قمنا باشتقاقه من الكلاس
MIDlet
والذي تحدثنا عنه مسبقا

private Display display;
تعريف كائن
display
والذي هو عبارة عن تمثيل لشاشة الموبايل
ويتم التعامل مع الشاشة عبر هذا الكائن
public void HelloMidlet(){}
وهذا هو المشيد
constructor
ولن نحتاج في مثالنا هذا اضافة اي كود بداخله لذلك هو خال من الكود
public void startApp(){
من هنا سيبتدأ تنفيذ البرنامج
display =Display.getDisplay(this);
بعد ان قمنا بتعريف الكائن في السابق
الان سنقوم بربطه مع شاشة هذا التطبيق
طبعا هذا بشكل افتراضي

display.setCurrent(box);
الان نقوم بعرض صندوق النص على شاشة البرنامج
public void destroyApp(boolean unconditional){}
هذة الداله نقوم بإستدعائها عندما نريد اغلاق التطبيق
وقد قمنا باستخدامها في الدروس اللاحقة

يكفي ان نقوم بذكر اسم الدالة و كذلك نقوم بتشغيل الملاحظ الذي يتحسس عملية الاغلاق كما في هذا المثال
Exit عندما نقوم بالضغط على الزر

```
if ( c == cmdExit )
{
destroyApp(false);
notifyDestroyed();
}
```