عادل مبارك

مجال تطوير تطبيقات الهاتف المحمول يعتبر من الفوع الهامة في مجال تقنية المعلومات و علوم الحاسوب، وذلك يعود الي التوسع الكبير في مستخدمين الهاتف المحمول، نحن الآن في عصر تقنية المعلومات، لقد انتقلت معظم بلدان العالم، من ثورة الاتصالات التي تهتم بقل البيانات من شخص الي آخر، الي ثروة تقنية المعلومات، في هذا الكتاب سوف تتعرف علي اساسيات تطوير تطبيقات اندرويد، والتي تهتم ببرمجة وتشبيك الاجهزة النقالة التي تعمل علي نظام تشغيل اندرويد، والتي تضم عدد كبير من الاجهزة منها الهاتف المحمول، الهواتف الذكيه، وبقية الحاسبات النقالة التي تستخدم نظام تشغيل اندرويد. هذا الكتاب موجه الي شريحة المبتدئن في هذا المجال حيث يعطيهم المفاهيم الاساسية التي تمكنهم من الدخول في هذا المجال.

# تطوير تطبيقات أندرويد للمبتدئين

## تعلم تطوير تطبيقات اندرويد

## في ستة أبواب

صفحة الكتاب علي فيسبوك

**Adil Mobarek Ahm**

عادل مبارك

مايو 2016م

© جميع حقوق المؤلف محفوظة – إيداع رقم 2015/977م - الخرطوم

## جدول المحتويات

عن المؤلف: . . . . . 5
الباب الاول: مقدمة عن اندرويد . . . . . 6
الباب الاول: مقدمة عن اندرويد . . . . . 7
1. 1تقديم اندرويد . . . . . 7
1.2 البنية الهيكلية لنظام اندرويد . . . . . 10
1.3 اصدرات نظام اندرويد . . . . . 19
1.4 سوق تطبيقات اندرويد . . . . . 20
الباب الثاني: تجهيز بيئة عمل اندرويد . . . . . 22
تجهيز بيئة عمل اندرويد . . . . . 23
2.1 الادوات المطلوبة لبناء تطبيق اندرويد . . . . . 23
2. 1. 1ادات تطوير الجافا (Java Devel opment Ki t ( JDK : . . . . . 23
2. 1. 2بيئة تطوير تدعم تطبيقات اندرويد (I DE) . . . . . 23
2. 1. 3ادوات تطوير تطبيقات اندرويد ADT : . . . . . 23
2.2 استعراض ادوات تطوير اندرويد Andr oi d SDK . . . . . 25
2. 2. 1اتفاقية الترخيص l i cense Agr eement: . . . . . 25
2.2.2 الحقوق الممنوحة . . . . . 25
2. 2. 3استخدام ال SDK . . . . . 26
2. 2. 4تغير SDK والتوافقية . . . . . 26

2.2.5Android documentation . . . . . 26

2.2.6Application Framework . . . . . 26

الباب الثالث: كتابة اول تطبيق اندرويد . . . . . 27

كتابة اول تطبيق اندرويد . . . . . 28

3.1 خطوات انشاء مشروع اندرويد جديد . . . . . 28

3.2 انشاء Virtual device . . . . . 34

3.3 تنفيذ البرنامج علي Virtual device . . . . . 35

3.4 تشغيل البرنامج علي جهاز اندريد حقيقي . . . . . 36

3.5 شرح مكونات مشروع اندرويد : . . . . . 37

الباب الرابع: تصميم واجهات المستخدم . . . . . 41

تصميم واجهات المستخدم . . . . . 42

4. 1 مفهوم ال layout وانواعه . . . . . 42

4. 2 تصميم واجهات المستخدم البسيطة . . . . . 49

4.3 التنقل بين الشاشات عن طريق Intent . . . . . 53

4.4 واجهات المستخدم المتقدمة: . . . . . 53

4.5 استخدام ال Fonts و ال Styles: . . . . . 58

الباب الخامس: فهم اساسيات تطوير تطبيقات اندرويد . . . . . 63

الباب الخامس: فهم اساسيات تطوير تطبيقات اندرويد . . . . . 64

5.1 المكونات الاساسية لتطبيق اندرويد: . . . . . 64

5.2 Service: . . . . . 72

5.3 Intent: . . . . . 75

Explicit Intents . . . . . 76

نقل البيانات بين ال Intents (Data transfer between activities) . . . . . 77

Defining intent filters: . . . . . 78

5.4 Context: . . . . . 88

5.5 التعرف علي ملف Manifest File . . . . . 88

الباب السادس: استخدام المكتبات البرمجية الشائعة . . . . . 91

اباب السادس: استخدام المكتبات البرمجية الشائعة . . . . . 92

6. 1استخدام مكتبات اندرويد الخاصة بالاتصال بالشبكة Android Networking . . . . . 92

6.2 مكتبات اندرويد لانشاء وادارة قاعة البيانات المحليةLocal Database . . . . . 95

6.3 مكتبات اندرويد للتعامل مع الوسائط المتعددةAndroid Multimedia . . . . . 100.

مراجعة للجافا . . . . . 102

المراجع . . . . . 126

عادل مبارك احمد

تابع التدريب العملي للكتاب علي قناتي علي يوتيوب

http://www.youtube.com/c/AdilMobarek

عن المؤلف:

- بكلاريوس الشرف بجامعة السودان للعلوم والتكنلوجيا – كلية علوم الحاسوب وتقانة المعلومات في عام 2010م.
- عمل كمساعد تدريس جامعة السودان للعلوم والتكنلوجيا – كلية علوم الحاسوب وتقانة المعلومات في عام 2011
- ماجستير علوم الحاسوب – جامعة الخرطوم - يناير 2015.
- مدرب لمادة تطوير برمجات الهاتف - من جامعة السودان للعلوم والتكنولوجيا-2012م و اكاديمية المنهل للعلوم، 2014م.
- مدرب معتمد من مركز نيوترين للتدريب بالخرطوم ومعاهد العالم العربي بالقاهرة- 2016.
- نشر ورقة علمية في موضوع Mobile Database في مجلة (IJCSA)

الرابط : http://t.co/MvaoGC7nDx

- مطور لتطبيقات الموبايل منذ عام 2010م.
- Email : adil.mobarek@gmail.com

عادل مبارك احمد

# الباب الاول: مقدمة عن اندرويد

تابع التدريب العملي علي قناتي علي يوتيوب

http://www.youtube.com/c/AdilMobarek

# 1 الباب الاول: مقدمة عن اندرويد

## 1. 1تقديم اندرويد

الاندرويد هو عبارة عن بيئة عمل متكاملة شاملة و مفتوحة للاجهزة النقالة، فمن ناحية مصنعوا الاجهزة يمكن لاي شركة ان تقوم بتصنيع أجهزة موافقة مع منصة عمل الأندرويد اما من ناحية مطوروا البرامج فأندرويد يوفر لهم بيئة تطوير برمجيات مجانية ومفتوحة المصدر، وهي تحتوي علي نظام تشغيل لينكس " ker nal Li nux 2. 6 "وواجهة مستخدم و عدد من التطبيقات وقامت شركة قوقل بتطوير نظام أندرويد بالتعاون مع تحالف الهواتف المفتوح OHA .

**تحالف الهواتف المفتوح**

هي عبارة عن اتفاقية عمل، تشارك في هذه الاتفاقية 47 شركة، وذلك من أجل تطوير معايير مفتوحة لاجهذة الهاتف المحمول، الشركات التي تشارك في هذه الإتفاقية لها مجالات مختلفة لاكنها من جهة ما لديها مصالح من تطوير تقنيات الهاتف المحمول، وهذه المجالات هي:

- مصنعو أجهزة الهاتف المحمول مثل شركة سامسونق
- شركات الإتصالات
- شركات تطوير البرمجيات مثل قوقل
- شركات أشباه الموصلات
- الشركات التجارية

لقد احدثت أندرويد ثورة في عالم الهواتف المحمولة لما يتمتع به من مميزات فهي بيئة عمل متكاملة، إضافة الي أن بيئة العمل أندرويد لا تعتمد علي جهاز نقال معين فهي تفصل بين الجهاز النقال و التطبيقات فقد تم تصميمه ليعمل علي اي جهاز نقال متوافق مع نظام اندرويد "فهو يحتوي علي وثيقة تعريف التوافق لوصف القدرات المطلوبة للجهاز لدعم حزمة البرمجيات sof t war e st ack"، هذا مما ادي الي انتشار النظام بسرعة خلافا لانظمة الهواتف الاخري مثل ايفون ففي الايفون نجد ان بيئة العمل و الاجهزة مملوكة من قبل شركة ابل.

**خلفية بسيطة عن اندرويد:** اندي روبن يعتبر من مؤسسي منصة العمل اندرويد و هو مؤسس شركة اندرويد، في عام 2005 شركة قوقل قامت بشراء شركة اندرويد، وبدات شركة قوقل مع أعضا تحالف الهاتف المفتوحOHA (Open Handset Al l i ance) في تطوير منصة عمل غير مسجلة الملكية مبنية علي التقنية المطورة في شركة اندرويد وكانت النتيجة هي مشروع اندرويد، معظم منصة عمل اندرويد طورت بالكامل عن طريق فريق اندي روبن في شركة قوقل، وهو لا يذال يدير خارطة طريق منصة عمل اندرويد، في عام 2007 اعلن تحالف الهواتف المفتوح OHA ان اندرويد مفتوح رسميا.

وقد قامت شركة قوقل بنشر مشروع اندرويد عبر الانترنت كما وفرت توثيق كامل لمنصة العمل، ادوات التطوير SDK للمطورين، كما شجعت شركة قوقل المطورين لكتابة تطبيقات جديدة ومنحت جوائز لافضل المطورين تصل الي 10 مليون دولار!.

في عام 2008 تم اصدار او نسخة من اندرويد Andr oi d SDK 1. 0

في عام 2009 ظهرت اصدارات جديدة من نظام التشغيل وهي Cupcake (1. 5) Donut (1. 6), Ecl ai r (2. 0). وهناك اكثر من 20 نوع جهاز يعمل علي نظام اندرويد.

الاندرويد احدث موجة جديدة في انظمة تشغيل الهواتف المحمولة الحديثة - فمنصة العمل مثل مايكروسوفت وندوس و ابل واي فون تدعم بيئة تطوير سهلة وغنية بالتطبيقات لتطبيقات الهاتف المحمول علي الرقم من ذلك فهي ليس مثل الاندرويد فالتطبيقات مبنية علي نظام مملوك من قبل كل من شركة مايروسفت و ابل لذلك في بعض الاحيان يتم اعطاء اولوية للتطبيقات المطورة في داخل الشركة Nat i ve Appl i cat i ons عن التطبيقات الاخريThi r d- par t y Appl i cat i ons و تقييد الاتصال بين التطبيقات المطورة في داخل الشركة و بيانت الهاتف المهمة مثلا لا يستطيع المطورين الاخرين من كتابة برنامج لعرض حالة الزاكرة او بيانات الهاتف او

ارقام الهواتف المخذنة في زاكرة الهاتف. و كذلك تتحكم وتقيد توزيع التطبيقات الاخري المنافسة لتطبيقات الشركة في بيئة العمل الخاصة بهم.

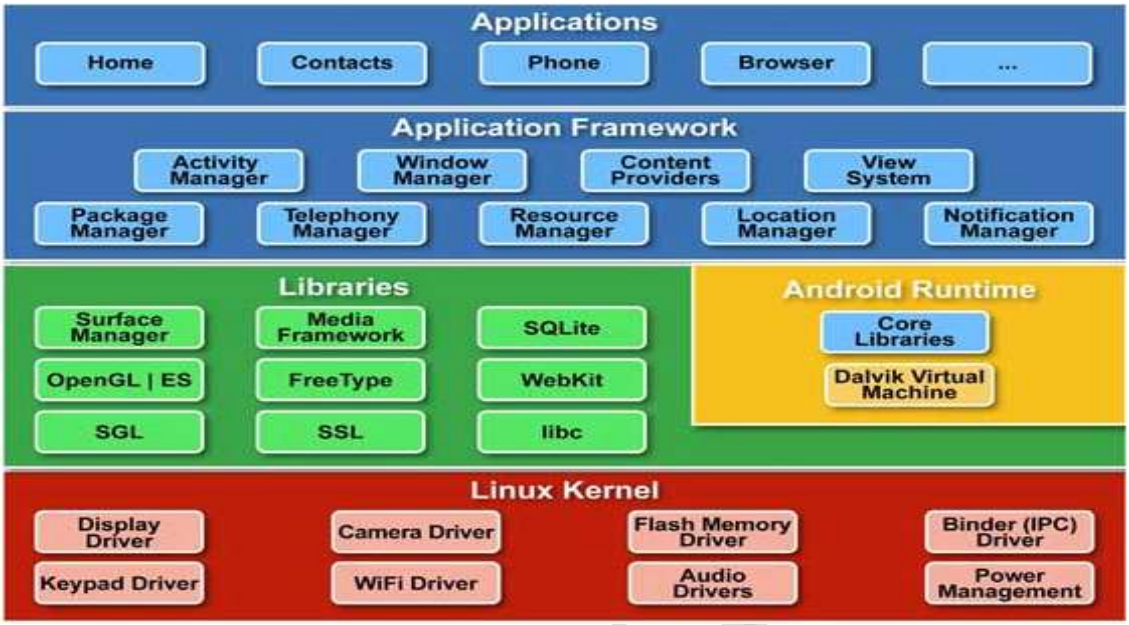
الاندرويد وفر امكانية جديدة لتطبيقات الهاتف بمنح بيئة التطوير مفتوحة المصدر المبنية علي نواة نظام التشغيل Li nux المفتوحة المصدر، كذلك عملية التعامل مع العتاد Har dwar e متاح لكل التطبيقات عن طريق المكتبات API s Li b والتفاعل بين التطبيقات المختلفة ايضا متاح فمثلا تطبيق C ont act يقم بعرض قائمة ارقام الهواتف والاسماء وبيانات اخري، يمكنك من داخل تطبيق آخر من عرض هذه القائمة.

في بيئة تطوير أندرويد كل التطبيقات متساوية في الصلاحيات، التطبيقات الاساسية مثل Gmai l , Cl ock, Cont act والتطبيقات المكتوبة من قبل المطورون مكتوبة بنفس A P I s وتنفذ علي نفس R unt i me خلافا لبيئات التطوير الاخري مثل Si mpi an c++ والتي تقيد من الوصول الي بعض API s الا بعد اعطاء التصريح بواسطة Li cense Key .

## 1.2 البنية الهيكلية لنظام اندرويد

الهدف من هذا الفصل هو فهم هيكلية نظام اندرويد وهو يساعدك كمطور تطبيقات من معرفة ما يمكن أن تسطتيع فعله و ما لايمكن فعله عن طريق البرمجة وكيف يعمل نظام اندرويد من اعلي مستوي الي ادني مستوي.

نظام تشغيل اندرويد هو عبارة عن طبقات من البرمجيات وهي مقسمة الي خمسة اجزاء واربعة طبقات، اي طبقة لها مميزاتها ومهامها كما موضح في المخطط التالي:




**شرح المخطط:**

## طبقة التطبيقات Application Layer

عند شراءك هاتف جديد يعمل علي نظام تشغيل اندرويد، سوف تجد مجموع من التطبيقات المثبتة علي هاتفك، مثل تطبيق البريد الالكتروني، هذه التطبيقات تاتي مع حذمة نظام التشغيل اندرويد، وهذه الطبقة من اسمها انها تحتوي علي التطبيقات المختلفة التي يستخدمها المستخدم مثل برنامج الحاسبة ومتصفح الانترنت، الخ.

## Application Framework

مهمة هذه الطبقة هي تسهيل اعادة استخدام عدد من المكنونات البرمجية API التي احتاجها لبنا تطبيق خاص بك، فمثال الي ذلك Tel ephony Manager، هي عبارة عن Appl i cat i on Fr amewor k او APIمجهز وموجود ضمن طبقات نظام تشغيل اندرويد، يمكن ان يستخدمه مطور برمجيات اندرويد لادارة عملية الاتصال بشبكة الهاتف المحمول، مثل الاتصال برقم هاتف معين من داخل تطبيق اندرويد، او قطع الاتصال، معرفة حالة الاتصال، الخ. والملاحظ هنا ان التطبيقات الاساسية مثل SMS ،emai l الموجودة في جهاز اندرويد تستخدم نفس API المتاحة لمطور تطبيقات اندرويد. الشكل ادناه يستعرض ال Feat ur es التي توفرها هذه الطبقة.

الجدول اداة يوضح دور كل جزء من طبقة اطار التطبيقات Appl i cat i on Fr amewor k

| Feat ur | المهمة |
| --- | --- |
| | |

| | |
|---|---|
| e | |
| Vi ew Syst em | يستخدم لعرض واجهات التطبيق، مثلا القائمة List، عرض النصوص Text، الازرار، الخ. |
| Cont en t Pr ovi d er | يمكن التطبيقات من الوصول الي البيانات من تطبيقات اخري، اي لجعل التطبيق يشارك بياناته للتطبيقات الاخري |
| Resour ce Manage | يدعم الوصول الي موارد التطبيق مثل النصوص St r i ng، الصور I mages، وملفات الواجهة l ayout f i l es |

عادل مبارك احمد

| | |
|---|---|
| r | |
| Not i f i c at i on Manager | يمكن تطبيق اندرويد من عرض رسالة تنبيه للمستخدم في شريط الحالة st at us bar |
| Act i vi t y Manage r | ادارة دورة حياة ال Act i vi t y في تطبيق اندرويد، |

عادل مبارك احمد

## Li br ar y and Andr oi d Runt i me

- **Libraries**

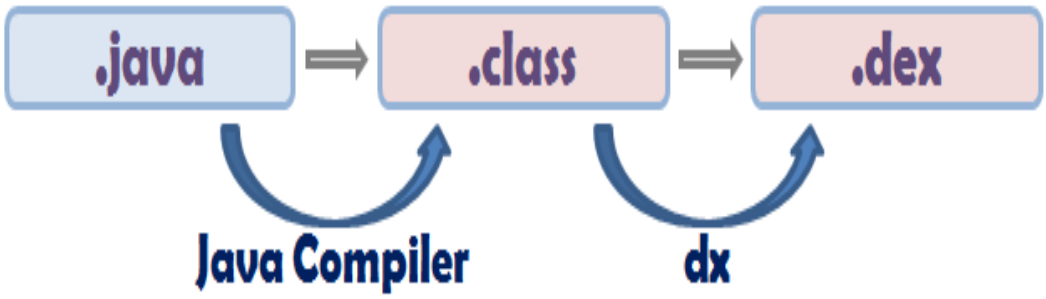
المكتبات تحتوي علي مجموعة من المكتبات المكتوبة بلغة c/ c++ تستخدم بواسطة مكونات نظام التشغيل، حيث ان تطبيق اندرويد يصل لهذة المكتبات عن طريق Application Framework، كما موضح في الشكل السابق.

- **Android Runtime**

كما موضح في الشكل اعلاه فانها تحتوي علي المكتبات الاساسية بالاضافة الي Dalvik Virtual Machine، ال Core Libraries تحتوي علي مجموعة من ال API، امثلة لها:

- Data Structures
- Utilities
- File Access
- Network Access
- Graphics

اما المكون الثاني وهو Dalvik Virtual Machine، وهي عبارة عن البيئة التي ينفذ فيها اي تطبيق اندرويد، حيث يتم تنفيذ عدد من VM في وقت واحد وب كفاءة عالية، حيث ان DVM مصممة علي ذلك، بمعني ان الDVM مجهذة لكي يتم تنفيذ اي تطبيق اندرويد فيVM خاصة به




## طبقة نواة لينكس Linux Kernel

نظام اندرويد هو مبني علي نواة Li nux2. 6 مع بعض التغيرات الهيكلية تمت بواسطة قوقل وهو نظام جيد ومفتوح المصدر و هناك عدة أسباب لإختيار Li nux وهي:

:Por t abi l i t y

نظام Li nux هو بيئة عمل قابلة للنقلpor t abl e pl at f or m يعني أن من السهل عمل لها Compi l e علي عدد من الاجهذة المختلفة، وبالتالي لا يوجد اي قلق من ناحية الاجهزة التي تنفذ علية.

:Security

عند كتابة اي تطبيق يتعامل مع الHW مطور تطبيقات اندرويد يطلب خدمة الوصول الي الHW عن طريق نظام Linux وليس مباشرتا، و نظام Linux يقوم بالتعامل مع الHW، هذا يمنع من كتابة تطبيقات قد تضر بالهاتف، فاصبح نظام Linux اشبه ببوابة العبور او جسر بين تعليمات البرامج و الHW .

يوجد نظام Linux في ادني طبقة من نظام اندرويد و هو مسؤول عن التعامل مع Hardware، ويحتوي علي مشغلات الاجهذة الاساسية Hardware drivers ، مشغلات الاجهذة Hardware drivers هي عبارة عن برامج Software تتحكم وتتعامل مع الاجهذة وهي الطبقة الوسيطة بين HW وطبقات Software الاخري.

اضافة الي ذلك نظام اندرويد يستخدم Linux في كل المهام الاساسية لنظام التشغيل مثل ادارة الزاكرة وادارة ال process، واعدادات الشبكة، الخ.

## 1.3 اصدرات نظام اندرويد

جهاز الاندرويد ينفذ اصدارات مختلفة من بيئة تطبيقات اندرويد Andr oi d pl at f or m عادتا الاصدارات القديمة اقل امكانتيات حتي الاجهذة التي تنفذ عليها سعرها اقل، بينما الاصدارات الجديدة اكثر تنفذ علي اجهذة زات امكانيات عالية، الجدول التالي يوضح اصدارات اندرويد و مستوي ال API.

| Andr oi d SDK Ver si on | ( Val ue as API Level I nt eger ) |
|---|---|
| Andr oi d 1. 0 SDK | 1 |
| Andr oi d 1. 1 SDK | 2 |
| Andr oi d 1. 5 SDK ( Cupcak | 3 |
| Andr oi d 1. 6 SDK ( Donut | 4 |
| Andr oi d 2. 0 SDK ( ‘cl ai r ) | 5 |
| Andr oi d 2. 0. 1 SDK ( ‘cl ai r ) | 6 |
| Andr oi d 2. 1 SDK ( ‘cl ai r ) | 7 |

Andr oi d Wi r el ess Appl i cat i on Devel opment – second Edi t i on *

I SBN 978- 0- 321- 74301- 5 - 2010

## 1.4 سوق تطبيقات اندرويد

شركة قوقل انشأت متجر مفتوح لنشر وبيع تطبيقات اندرويد، المتجر اسمة. Googl e pl ay

يمكنك ريارة الموقع من الرابط:

ht t ps: / / pl ay. googl e. com

كمتجر مفتوح شركة قوقل اعطت المطورين الحرية في كيفية بيع تطبيقاتهم كما يحبون يمكن ان توزع التطبيقات لكل المتاجر ولكل الاجهزة المختلفة، و كذلك يمكن ان تركز علي نوع معين من الاجهزة او عدد من الاجهزة، كذلك يمكنك تحديد سعر التطبيق او توزيعة مجانا، كما لك الحريه في تعديل سعر التطبيق متي شئت.

سوق التطبيقات Googl e Pl ay ليس متاح في كل الدول فبعض الدول

متاح تحميل التطبيقات المجانية ولكن غير متاح شراء التطبيقات و رفع التطبيقات لبيعها لوجود بعض القيود منها السياسية ومنها التقنية المتعلقة بطريقة الدفع.

ht t p: / / devel oper . andr oi d. com

عادل مبارك احمد

# الباب الثاني: تجهيز بيئة عمل اندرويد

تابع التدريب العملي علي قناتي علي يوتيوب

http://www.youtube.com/c/AdilMobarek

عادل مبارك احمد

# 2 تجهيز بيئة عمل اندرويد

في هذا الباب سوف نتعلم كيفية تجهيز بيئة العمل علي نظام نشغيل ويندوز.

## 2.1 الادوات المطلوبة لبناء تطبيق اندرويد

لكي تقوم بعملية كتابة تطبيق اندرويد يجب ان تكون البرامج الاتية مثبتة علي حاسبك:

### 2. 1. 1 ادات تطوير الجافا (JDK) Java Devel opment Ki t :

يمكنك تحميل الاصدارة الحديثة لجافا من موقع اوركل

ht t p: / / www. or acl e. com/ t echnet wor k/ j ava/ j avase/ downl oad s/ i ndex. ht ml

او اذا قمت بتثبيت جافا مسبقا علي حاسبك تاكد انها الاصدارة الخامسة او ما يليها يمكنك تحميلها من الرابط اعلاه.

### 2. 1. 2 بيئة تطوير تدعم تطبيقات اندرويد (I DE)

في هذا الكتاب سوف نقوم باستخدام بيئة العمل Ecl i ps يمكنك تحميل الملفات من الرابط

ht t p: / / www. ecl i pse. or g/ downl oads

### 2. 1. 3 ادوات تطوير تطبيقات اندرويد ADT :

وهي حذمة برمجية تحتوي علي المكتبات التي تستخدمها في تطوير تطبيقات اندرويد، قم بتحميل الحذمة من الرابط ht t p: / / devel oper . andr oi d. com/ sdk/ i ndex. ht ml .

لمعرفة تعليمات تثبيت هذه البرامج علي هاتفك اذهب الي الرابط

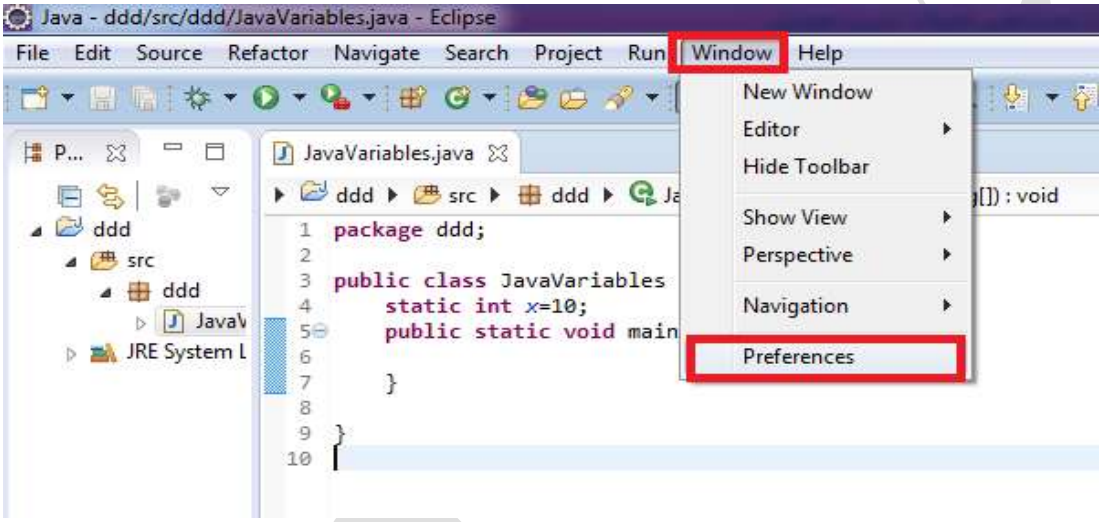
. http://developer.android.com/sdk/installing.html

بعد قراءة تعليمات التثبيت و تثبيت كافة البرامج علي حاسبك تكون جهزت بيئة العمل لتطوير تطبيقات اندرويد.

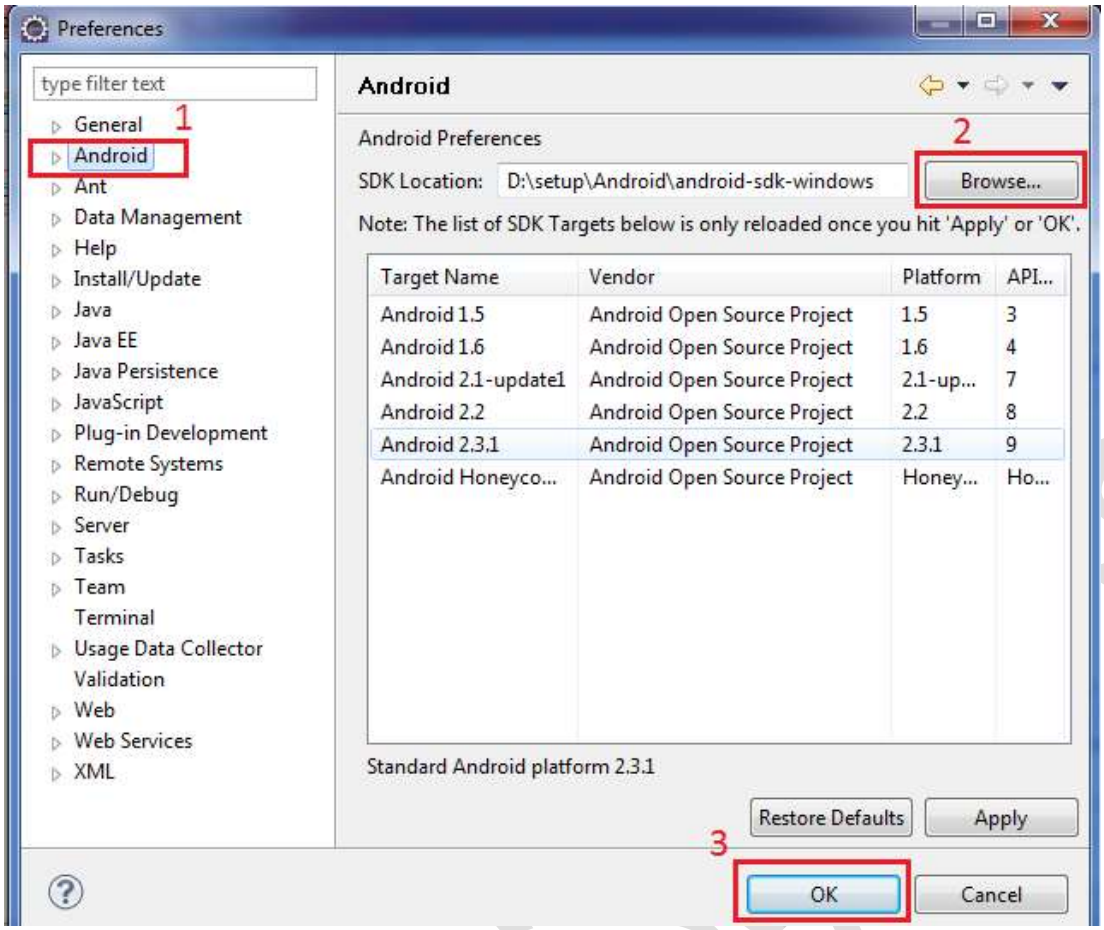
بعد تثبيت هذه البرامج علي حاسبك الشخصي، سوف تقوم بفتح الeclipse ، وربط الeclipse

بادات تطوير اندرويد ADT، لتقوم بهذة المهمة بنجاح، اتبع الخطوات البسيطة التالية:

1. من قائمة Windows في eclipse، اختر preference :




2. سوف تظهر لك الشاشة التالية، قم بتحديد مسار ادات تطوير اندرويد، ثم اضغط علي زر apply.




## 2.2 استعراض ادوات تطوير اندرويد Andr oi d SDK

ادات تطوير تطبيقات اندرويد بها خمس مكونات اساسية وهي :

### اتفاقية الترخيص l i cense Agr eement:

قبل تحميل ادوات تطوير برامج اندرويد في حاسبك، يجب ان تراجع وتوافق علي اتفاقية الترخيص الخاصة بادوات تطوير اندرويد، وهي عبارة عن عقد بينك انت كمطور تطبيقات وبين شركة قوقل كشركة مالكة لحقوق الطبع، عادتا في الشركات يقوم شخص ما بالموافقة علي هذه الاتفاقية، لكن انت كمطور يجب ان تعي الامور التالية:

### الحقوق الممنوحة

**شركة قوقل منحث المطورين رخصة محدودة لاستخدام اداة التطوير sdk، استخدام ال sdk لا يمنحك صلاحية استحدام اي علامة تجارية لقوقل، او اي شعار او اسم تجاري لقوقل.**

### استخدام ال SDK

**انت كمطور تطبقات اندرويد فقط تقوم بتطوير تطبيقات الاندرويد، لا يمكنك ان تقوم بعمل مشتق من الsdk، او توزيع ال SDK مع تطبيق اخر.**

### تغير SDK والتوافقية

**شركة قوقل لها الحق في تغيير ال SDK في اي وقت بدون اي اشعار بغض النظر عن التوافقية للاصدارات السابقة.**

## Andr oi d document at i on

يمكنك الاطلاع علي توثيق ادوات تطوير اندرويد من الموقع الالكتروني

ht t p: / / devel oper . andr oi d. com

## Appl i cat i on Fr amewor k

ادوات تطوير اندرويد قدمت كحذمة واحة في الملف andr oi d. j ar، وبداخلة عدد من الحذم المهمة كما في الجدول التالي:

استعراض الحذم المهمة في اداة تطوير اندرويد:

# الباب الثالث: كتابة اول تطبيق اندرويد

تابع التدريب العملي علي قناتي علي يوتيوب

http://www.youtube.com/c/AdilMobarek

# 3 كتابة اول تطبيق اندرويد

في لغات البرمجة المشهورة مثل ال c++ والجافا يستخدم مصطلح class للدلالة علي جزء او مكون من البرنامج ويتكون ال class من عدد من الدوال functions والمتغيرات variable.

بيئة التطوير اندرويد لا تختلف كثيرا عن هذة المكونات برنامج اندرويد يحتوي علي عدد من classes وبداخله عدد من الدوال والمتغيرات السؤال هو ما هو الجديد في الاندرويد؟.

عند تصميم بيئة عمل للهواتف المحمولة اندرويد هناك عدة اشياء تم مراعاتها حيث ان الهاتف المحمول له ذاكرة صغيرة نسبيا Ram ، بالاضافة الي صغر حجم الشاشة، لتقليل المساحة التي يحجزها البرنامج في الزاكرة تم اعداد عدد من ملفات xml لتخذين كل الموارد resource التي يحتاجها برنامج اندرويد ومن امثلة الموارد:

String variables, files, images, graphic interfaces

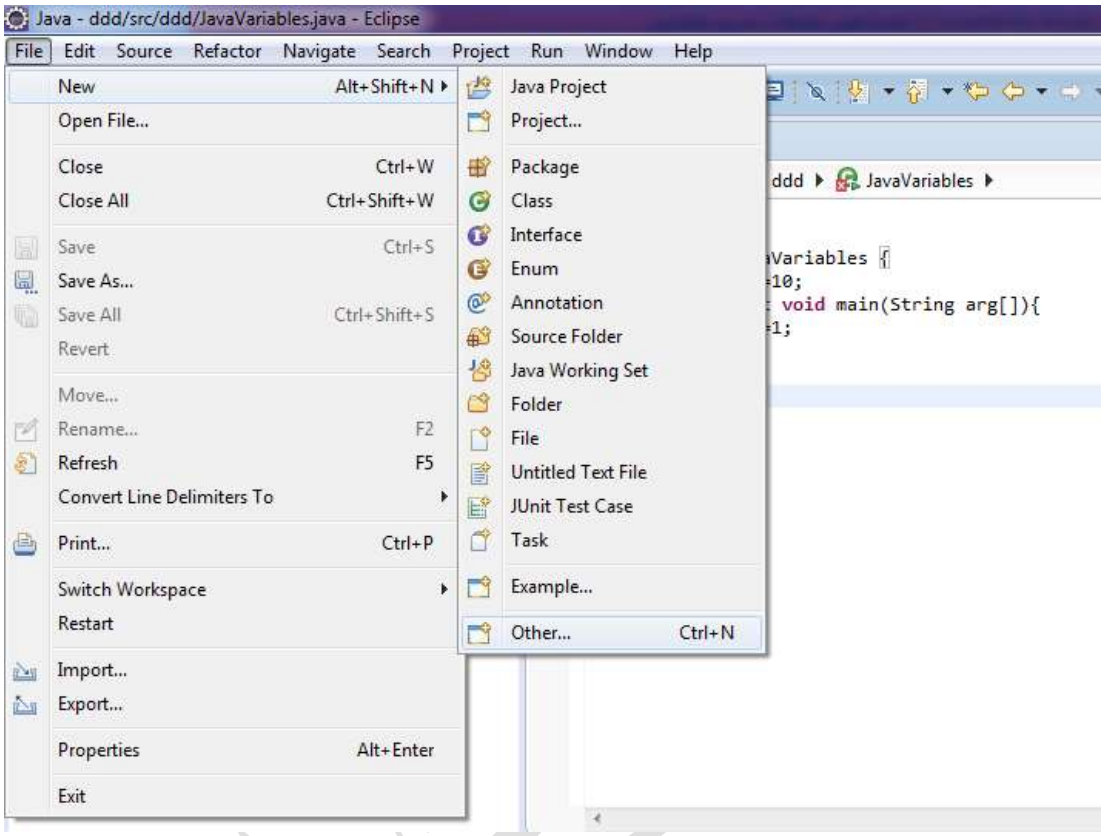
حيث يتم تخذين هذه الموارد مرة واحدة في ملف xml ونداءها كل مرة عند الحاجة، مما يؤدي الي تقليل المساحة المحجوزة في الزاكرة.

## 3.1 خطوات انشاء مشروع اندرويد جديد

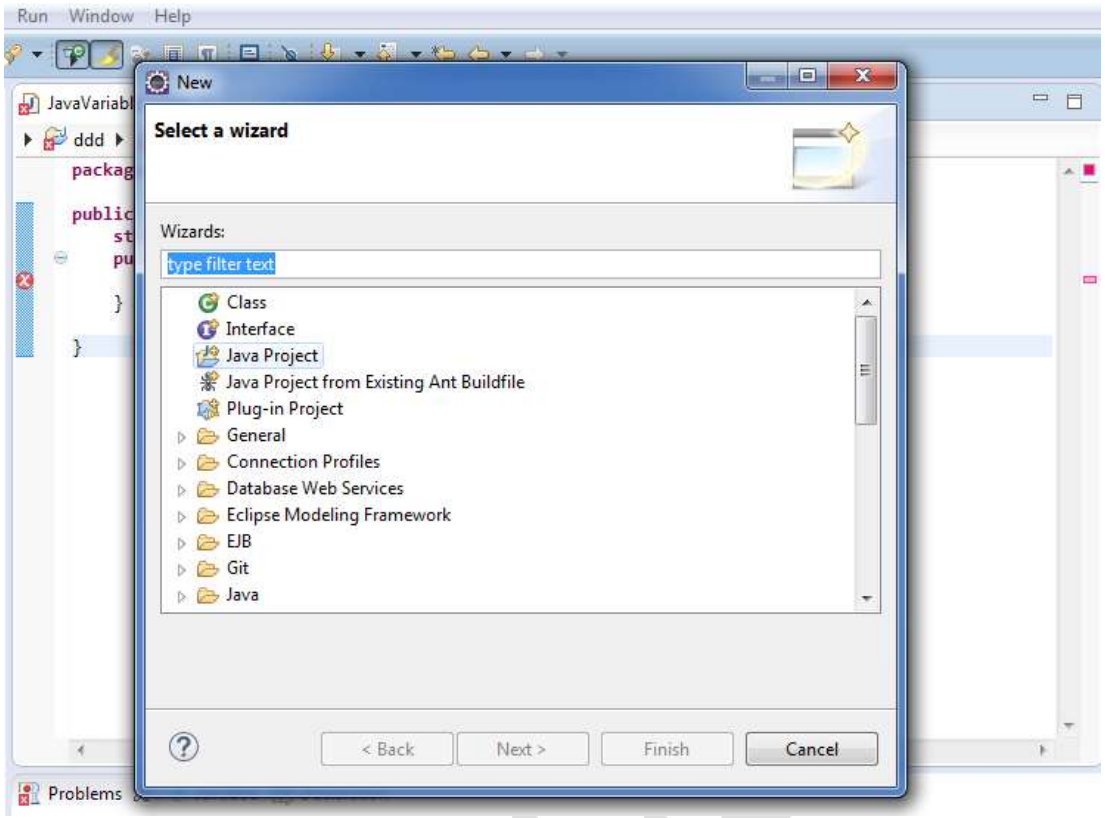
الان سوف تقوم بانشاء تطبيق اندرويد لكن قبل البدء يجب عليك اختبار بيئة التطوير من جاهزيتها من تنفيذ البرنامج، لكي تقوم بعملية الاختبار قم بتنفيذ احد البرامج الجاهزة الموجودة في بيئة التطوير.

لكي تقوم باشاء تطبيق اندرويد اتبع الخطوات الاتية:

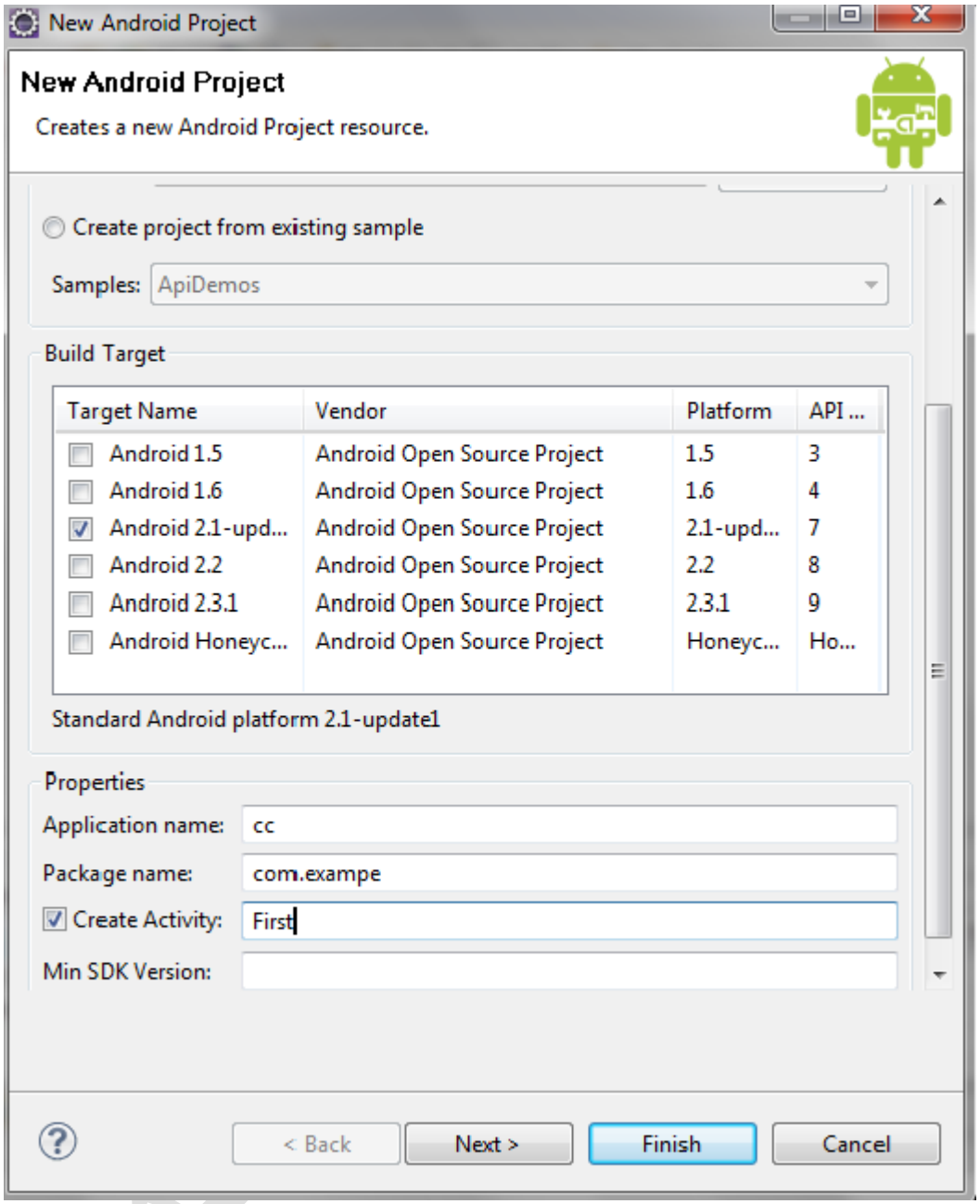
"Fi l e - > New - > Ot her .... "




اضغط علي . . Ot her سوف تظهر لك الشاشة التالية:




قم باختيار Andr oi d Pr oj ect سوف تظهر لك الشاشة التالية:




قم بادخال بيانات تطبيق اندرويد Andr oi d Appl i cat i on:

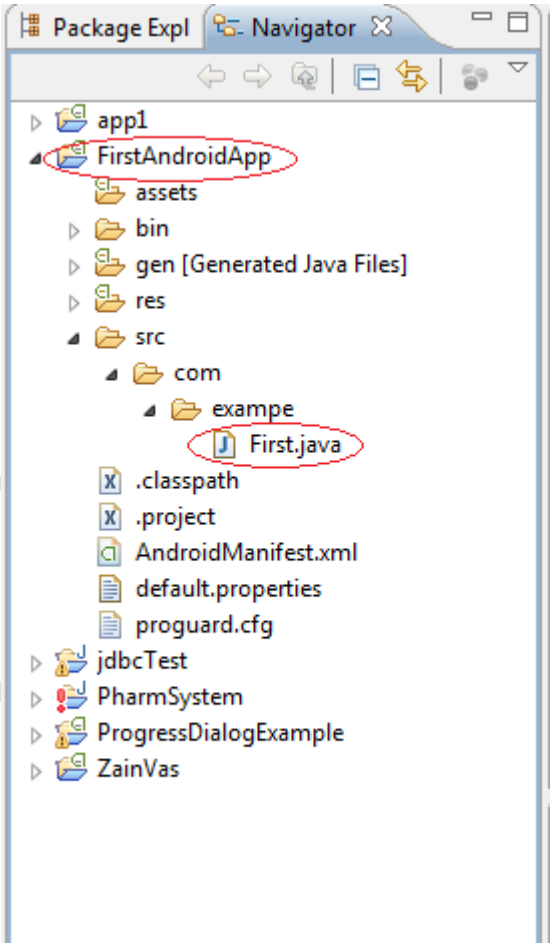
في حقل Pr oj ect Name قم بادخال اسم التطبيق مثلا:

Fi r st Andr oi dApp→ Pr oj ect Name

في قائمة خيارات ال cont ent قم باختيار cr eat e NewPr oj ect i n Wor kspace.

في قائمة Bui l t Tar get قم باختيار اصدارة الاندرويد التي التي تناسب الاجهزة المستهدفة.

قم بالضغط علي زر انهاء Fi ni sh سوف تظهر لك القائمة الاتية:




قم بفتح الملف Fi r st . j ava.

First.java

```
package com.exampe;

import android.app.Activity;

public class First extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

في الدالة onCreate() قم باضافة الكود الاتي:

First.java

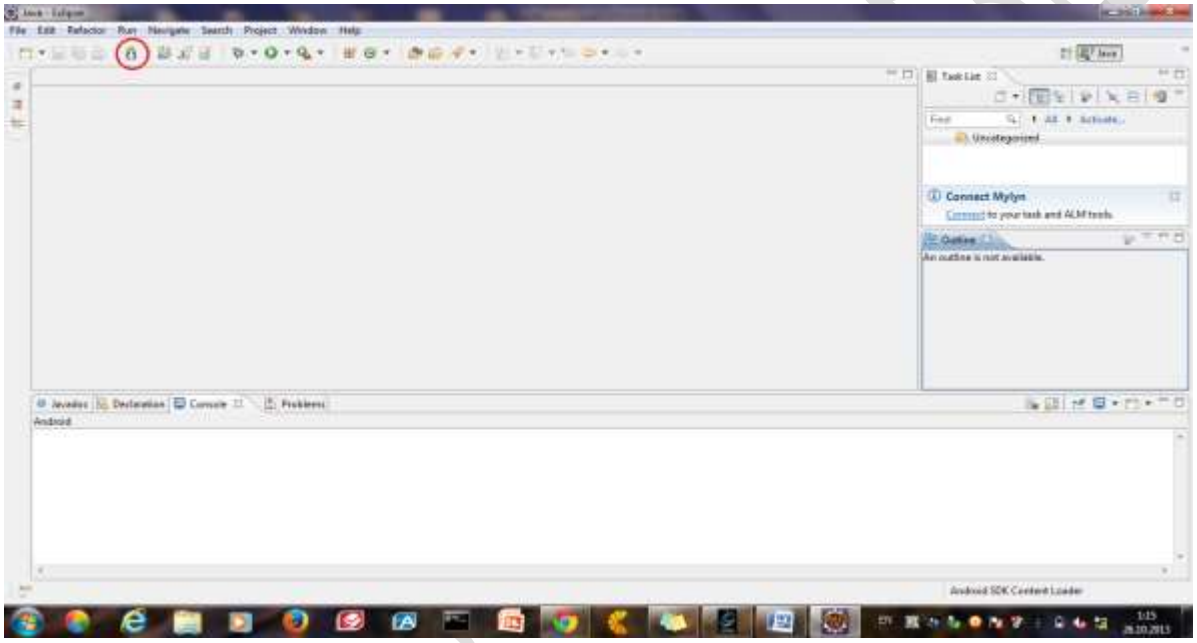
```
package com.exampe;

import android.app.Activity;

public class First extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        TextView text = new TextView(this);
        text.setText("Hello World, Android");
        setContentView(text);
    }
}
```

عند تنفيذ البرنامج يتم طباعة الرسالة "Hello World, Android" في اعلي الشاشة.

## 3.2 انشاء Virtual device

قبل ان تبداء بتنفيذ البرنامج لا بدء من انشاء Virtual device وهو عبارة عن برنامج محاكي لاجهذة الاندرويد الحقيقية حيث يقوم بتنفيذ البرنامج عليه، لانشاء Virtual devise اتبع الخطوات الاتية:

اضغط علي لفتح Android Virtual devise Manage كما في الصورة التالية:



بعد الضغط علي يظهر لك مربع الحوار كما في الصورة:

قم بالضغط علي زرNEW يظهر لك مربع لاعدادات ال AVD قم بكتابة اسم AVD وتحديد API Level ثم اضغط علي زر***Creat e AVD***.

## 3.3 تنفيذ البرنامج علي Vi r t ual devi ce

بعد ان قمت باعداد AVD، قم بالضغط علي زر التنفيذ من قائمة المهام لبيئة عمل ecl i ps.

اذا تمت عملية التنفيذ بنجاح سوف يظهر لك ناتج التنفيذ كما في الصورة:

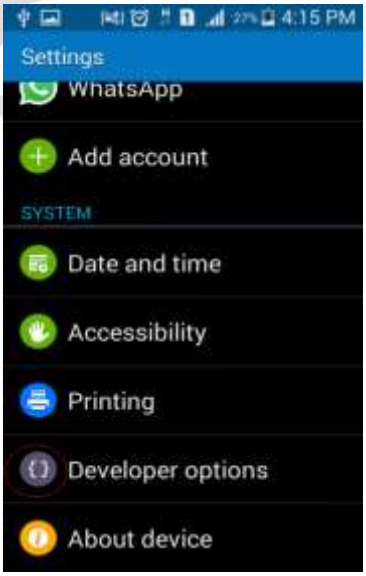
## 3.4 تشغيل البرنامج علي جهاز اندريد حقيقي

الان قد قمت بتنفيذ البرنامج ولاكن علي محاكي جهاذ اندرويد (AVD) Andr oi d vi r t ual devi ce في هذه الجزئية سنتعلم كيف تقم بتنفئذ اي برنامج اندرويد علي الجهاز الحقيقي r eal devi ce، اليك الخطوات الاتية:

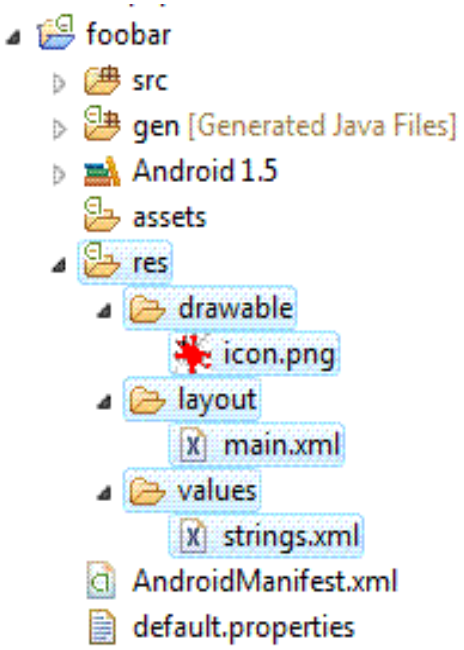
**بعض الاعداد علي جهاذ الهاتف:**

لكي تمكن هاتفك من تنفيذ البرنامج مباشرتا بواسطة بيئة تطوير اندرويد يجب عليك ضبط الاعدادات الاتية:

Set t i ng - > devel oper opt i on- > al l ows usb debuggi ng



## 3.5 شرح مكونات مشروع اندرويد :



مشروع تطبيق اندرويد يحتوي علي مجوعة من المجلدات كما في الشكل ادناة، كل مجلد لديه اسم معين يدل علي نوع الملفات التي يحتفظ بها، هذة المفات كالاتي:

**اولا: ملفات الشفرة البرمجية (src)**

وهو مجلد ملف يحتوي علي مصدر شفرة البرنامج Source files، يحتوي هذا المجلد علي ملفات الجافا، اي ملف عبارة عن صنف معين class، فمثلا اذا كان لديك ثلات شاشات في البرنامج three Activities فان هذا المجلد يحتوي علي ثلاث ملفات وهي عبارة عن java classes، كل class يمثل Activity معينة، فاي شفرة برمجية يقوم كتابتها المبرمج توضع في هذا المجلد.

**ثانيا:**

gen: هذا المجلد يحتوي علي مجموعة من الملفات التي تم توليدها من عن طليق بيئة التطوير IDE، فاي معلومة داخل هذا المجلد لا تحتاج الي تعديلها، فتترك كما هي.

**ثالثا**

res: وهي مختصرة من كلمة Resource، وهو المجلد المسؤول عن حفظ موارد المشروع، يقصد بموارد المشروع كل انواع الملفات التي يحتاجها البرنامج لكي يودي وظيفتة عداء ملفات المصدر والتي امتدادها java.، قمثلا اذا كان البرنامج يعرض صور معينة فان هذة الصور تعتبرضمن موارد المشروع، او يقوم بتشغير ملف صوتي فان الملف الصوتي يعتر ضمن موارد المشروع، او يقوم بعرض عناصر واجهة المستخم عن طريف ملف Layout xml ، فان ال Layout.xml يعتبر ضمن موارد المشروع، هكذا، كل هذه الملفات توضع في المجلد res.

Assets:هي طريقة لادارة موارد التطبيق، يمكنك في هذا المجلد تخذين اي نوع من الملفات التي تحتاجها لتشغيل التطبيق، يستخدم الصنف **Asset Manager** للوصول الي محتويات هذا الملف،

```
InputStream stream= null;

    try {

      stream= assetManager.open("generated-keys.bin");

    } catch (IOException e) {

      // handle

    }
```

كما في جميع تطبيقات الجافا، يمكنك في النهاية اغلاق

InputStraem

```
    finally {

        if (stream!= null) {

            try {

                stream close();

            } catch (IOException e) {}

        }

    }
```

ليس مثل ملف الموارد Res f ol der حيث ان ال ASSET يمكن تخذين اي نوع من الملفات مدعومة بواسطة نظام التشغيل.

Andr oi dMani f est: هو عبارة عن xml f i l e، يحتوي علي المعلومات الاساسية لاي مشروع اندرويد، فمثلا قد يحتوي مشروع اندرويد علي ثلاث شاشات، تحديد الشاشة التي سوف تظهر للمستخدم اولا تسمي St ar t Act i vi t y تسجل في هذا الملف، لمذيد من المعلومات عن مهمة هذا الملف انظر الي الباب الخامس.

bi n: يحتوي علي الملف التنفيذي لتطبق اندرويد، عادتا لا يظهر ضمن الملفات التي تعرض في بيئة تطوير اندرويد، لانة ليس من الملفات التي يحتاجها المبرمج وقت التطوير، فهو يتم انشاءة بواسطة بيئة التطوير مثلecl i ps ، وذلك بعد عملية تنفيذ المشروع، يتم انشاءة بعد تنفيذ المشروع بسلامة بدون اي اخطاء في اللغة. فاسم هذا الملف يكون بنفس اسم التطبيق الذي فمت بادخالة في شاشة ***انشاء مشروع جديد*** وامتدادة يكون apk .، اختصارا لكلمة Appl i cat i on، وهو الملف الذي تم تنفيذة علي جهاذ اندرويد الحقيقي.

# الباب الرابع: تصميم واجهات المستخدم

تابع التدريب العملي علي قناتي علي يوتيوب

http://www.youtube.com/c/AdilMobarek

# 4 تصميم واجهات المستخدم

نجاح التطبيق الذي تقوم بتصميمة يعتمد كثيرا علي تصميم واجهة مستخدم اكثر سهولة وامان، . توفر بيئة تطوير برمجيات اندرويد طريقة سهلة لتصميم الواجهة، عن طريق السحب واللصق او ما يعرف pl ug- and- pl ay، ففي هذا الباب سوف نشرح مفهوم Layout وانواعه المختلفة، ثانيا سوف نستعرض كيفية تصميم واجهات المستخدم البسيطة، واخيرا سوف نستعرض كيفية تصميم واجهات المستخدم المتقدمه.

## 4. 1 مفهوم ال l ayout وانواعه

واجهة المستخدم في الاندرويد تنظم في شكل Layout و Vi ews، اداة تطوير البرمجيات اندرويد توفر الصنف Layout وهو المسؤول عن تنظيم ال Vi ews او W dget s التي تظهر في الشاشة كواجهة للمستخدم، وهو لا يرسم اي شي علي الشاشة فقط يقوم بتنظيم ال W dget او ال cont r ol s الاخري، الصنف Layout موجود في المكتبة البرمجية او الحذمة . andr oi d. wi dgt

اذا كان لديك خبرة مسبقة في تطوير واجهات المستخدم بواسطة لغة البرمجة جافا فان الLayout يشبة الصنف Container و ال views تشبة component.

هناك طريقتين لتنظيم الViews في الشاشة اما عن طريق كتابة اسطر من الكود او تعريف الواجهة عن طريف ملف xml الذي يتم وضعة علي المجلد folder الذي يسمي layout. الطريقة الشائعة هي تنظيم الواجهة عن طريق xml file، عند استخدام هذه الطريق كل ملفات Layout او بعبارة اخرة شاشات البرنامج تخذن في المسار Res/ Layout.

يوجد العديد من انواع ال Layout او بمعني اخر يوجد عدة طرق لتنظيم عناصر الشاشة، فمتاح لك كمبرمج ان تختار اي من الانواع الاتية حسب الطريقة التي تحب:

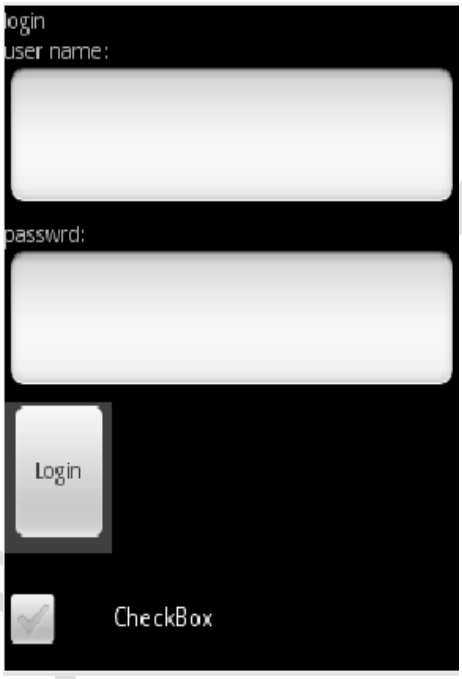
## التنظيم الخطي Liner Layout

يتم تنظيم ال Views في شكل خطي اما صف افقي horizontal اوعمودي vertical فيما يلي سوف نقوم بشرح مثال بسيط لل Liner Layout.

مثال يوضح تنظيم ال Views علي الشاشة باستخدام Liner Layout، في هذا المثال كان نمط التنظيم Vertical

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
   android:orientation="vertical"
   android:layout_width="fill_parent"
   android:layout_height="fill_parent"
   >
<TextView
   android:layout_width="fill_parent"
   android:layout_height="wrap_content"
   android:text="Simple Views"
   />
<TextView android:text="user name:" android:id="@+id/chickBox1"
android:layout_width="wrap_content" android:layout_height="wrap_content"></TextView>
<EditText android:id="@+id/editText1" android:text="" android:layout_width="fill_parent"
android:layout_height="wrap_content"></EditText>
<TextView android:text="passwrd:" android:id="@+id/textView1"
android:layout_width="wrap_content" android:layout_height="wrap_content"></TextView>
<EditText android:id="@+id/editText2" android:text="" android:layout_width="fill_parent"
android:layout_height="wrap_content"></EditText>
<Button android:layout_width="wrap_content" android:layout_height="wrap_content"
android:id="@+id/button1" android:text="Login"></Button>
<CheckBox android:layout_width="wrap_content" android:layout_height="wrap_content"
android:id="@+id/checkBox1" android:text="CheckBox"></CheckBox>
</LinearLayout>
```

الصورة ادناة توضح شكل واجهة المستخدم بعد تنفيذ البرنامج:




**:Rel at ed Layout**

المثال التالي يوضح كيفية تنظيم Vi ews علي واجهة المستخدم عن طريق استخدام الصنف Rel at edLayout.

```
<RelativeLayout
xmlns:android=”http://schemas.android.com/apk/res/android”
xmlns:tools=”http://schemas.android.com/tools”
android:layout_width=”match_parent”
android:layout_height=”match_parent” >
<TextView
android:layout_width=”wrap_content”
android:layout_height=”wrap_content”
android:layout_centerHorizontal=”true”
android:layout_centerVertical=”true”
android:text=”hello_world, this is RelativeLayout example ”
/>
</RelativeLayout>
```

## Frame Layout

المثال التالي يوضح كيف تعريف layout من نوع FrameLayout

```
<FrameLayout xmlns:android=

“http://schemas.android.com/apk/res/android”

android:id=”@+id/FrameLayout01”

android:layout_width=”wrap_content”

android:layout_height=”wrap_content”

android:layout_gravity=”center”>



</FrameLayout
```

هناك طريقتين في الاندرويد لتعريف Layout الطريقة الاولي هي المذكورة في الامثلة السابقة عن طريق xml layout file ، اما الطريق الثانية فهي عن طريق كتابة اسطر الكود، وهي غير شائعة الاستخدام، لانها اكثر صعوبة من ناحية الاستخدام، لاكنك في بعض الحالات سوف تضطر لاستخدامها، مثلا في بعض البرامج اختيار نوع Layout يعتمد علي طريقة تفاعل المستخدم، بمعني اخر انت كمبرمج ليس لديك قرار مسبق لاختيار Layout المحدد في وقت تطوير البرنامج.

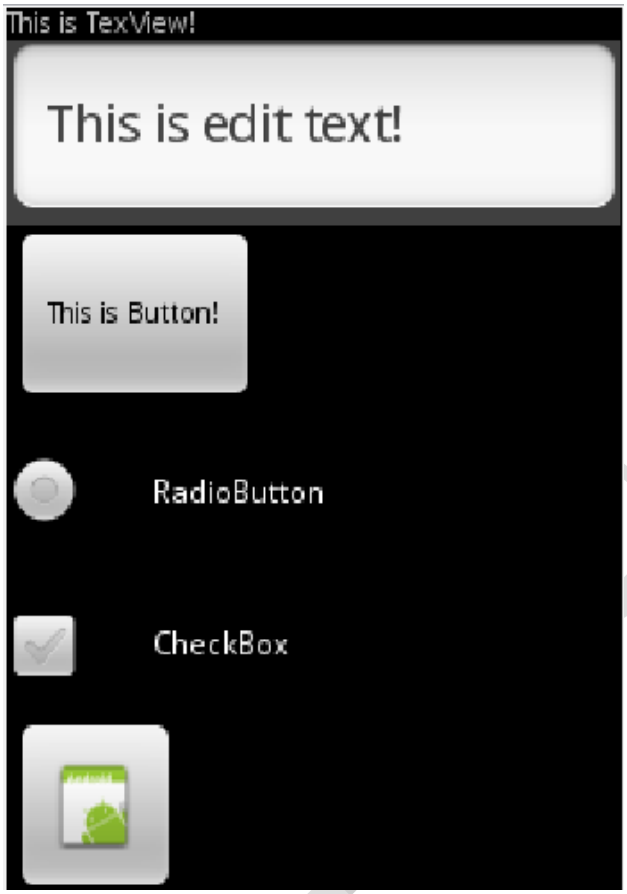
**مثال:**

التعليمات المستخدمة لتعريف Liner Lyout عن طريق كتابة اسطر داخل الصنف Activity هي:

```
public void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
TextView tv1 = new TextView(this);
tv1.setText("Hello!, I'm TV1");
TextView tv2 = new TextView(this);
tv2.setText("Hello!, I'm TV2.");
LinearLayout llayout = new LinearLayout(this);
llayout.setOrientation(LinearLayout.VERTICAL);
llayout.addView(tv1);
llayout.addView(tv2);
setContentView(llayout);
```

## 4. 2 تصميم واجهات المستخدم البسيطة

في هذه الجزئية سوف نقوم بدراسة كيفية تصميم الواجهات البسيطة والتي تشمل انواع ال vi ew الموضحة في الصورة التالية:




تحتوي الصورة علي النواع الاتية من الViews:

TextView: وهو المسؤول عن عرض النصوص علي واجهة المستخدم.

EditText: ويستخدم لعرض حقل ادخال البيانات النصية مثلا طلب ادخال اسم المستخدم.

Button: عند الضغط عليه يتم تنفيذ مهمة معينة.

RadioButton: عبار عن زر يستخدم لاختيار شي محدد او عدم اختيارة، حيث يتم اظهال لون اخضر في حالة الضغط علي الزر.

CheckBox: عبارة عن زر مربع لعرض قائمة من الاشياء للمستخدم، حيث يمكن للمستخدم الضغط علي اكثر من زر من قائمة الازرار.

ImgeButton: وهو عبارة عن Button لكن يظهر للمستخد في شكل صورة، عاتدتا الصورة توضح المهمة التي يقو م بها الز ر Button.

في الشكل التالي قمنا بعض ملف xml layout المسؤول عن عرض واجهة المستخدم في الصورة السابقة

مثال لل View باستخدام XML Layout

**projectName/res/layout/main.xml**

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"

android:orientation="vertical"

android:layout_width="fill_parent"

android:layout_height="fill_parent"

>

<TextView android:text="This is TexView!" android:id="@+id/chickBox1"
android:layout_width="wrap_content"
android:layout_height="wrap_content"></TextView>

<EditText android:id="@+id/editText2" android:text="This is edit text!"
android:layout_width="fill_parent"
android:layout_height="wrap_content"></EditText>

<Button android:layout_width="wrap_content"
android:layout_height="wrap_content" android:id="@+id/button1"
android:text="This is Button!"></Button>

<RadioButton android:text="RadioButton"
android:id="@+id/radioButton1" android:layout_width="wrap_content"
android:layout_height="wrap_content"></RadioButton>

<CheckBox android:layout_width="wrap_content"
android:layout_height="wrap_content" android:id="@+id/checkBox1"
android:text="CheckBox"></CheckBox>

<ImageButton android:layout_height="wrap_content"
android:id="@+id/imageButton1" android:src="@drawable/icon"
android:layout_width="wrap_content"></ImageButton>

</LinearLayout>
```

- كل عنصر من عناصر واجهة المستخدم اعلاه يسمي vi ew.

**مثال: استخدام CheckBox**

في السطر رقم 1 قمنا بتحديد اسم package التي يستخدما المشروع الحالي com chapt er 3، القاعدة هنا في تسمية ال packuge هي ان تتكون من اسمين علي الاقل ويفصل بين الاسم والاخر بعلام "."

```
package com chapt er 3;
i mpor t andr oi d. app. Act i vi t y;
i mpor t andr oi d. os. Bundl e;
i mpor t andr oi d. vi ew. Vi ew;
i mpor t andr oi d. wi dget . CheckBox;
i mpor t andr oi d. wi dget . Text Vi ew;

publ i c cl ass Fi r st Act ext ends Act i vi t y {
/ ** Cal l ed when t he act i vi t y i s f i r st cr eat ed. */
@Over r i de
publ i c voi d onCr eat e( Bundl e savedI nst anceSt at e) {
 super . onCr eat e( savedI nst anceSt at e) ;
  set Cont ent Vi ew( R. l ayout . mai n) ;
f i nal CheckBox check_but t on = ( CheckBox)
f i ndVi ewByI d( R. i d. checkBox1) ;
check_but t on. set OnCl i ckLi st ener ( new
Vi ew. OnCl i ckLi st ener ( ) {
```

```
publ i c voi d onCl i ck ( Vi ewv) {
Text Vi ewt v = ( Text Vi ew) f i ndVi ewByI d( R. i d. chi ckBox1) ;
t v. set Text ( check_but t on. i sChecked( ) ?
"Thi s opt i on i s checked" :
 "Thi s opt i on i s not checked") ;
}
}) ;
}
}
```

الاسطر من 2-6 تم استدعاء المكتبات التي سوف نستخدمها في هذا البرنامج، لا تقلق من هذه المهمة فبيئة التطوير تقوم بهذة المهمة بطريقة تلقائية، السطر رقم 8 قمنا بتعريف اسم الصنف cl ass اسمة Fi r st Act ، اما الكلمة المفتاحية keywor d توضع ان هذا ال cl ass عبارة عن امتداد ل cl assاخر اسمة Act i vi t y، اي صنف من نوع Act i vi t y يتم من خلاله عرض واجهة المستخدم.

في السطر رقم 11 قمنا باعادة كتابة الدالة onCr eat e( ) وهي من الدوال التي تم وراثتها من Act i vt y، هذا الدالة يتم تنفيذها لحظة تنفيذ البرنامج – هذة الدالة تشبه الي قدر ما الدالة mai n( ) في لغة البرمجة جافا، الان ما يهمنا اكثر في هذا المثال السطر رقم 13، تم استدعاء الدالة set Cont ent Vi ew( ) وهي من ضمن دوال Act i vi t y، وهذا الامر يعني اضع محتويات الواجه في لهذ ه Act i vi t y - محتويات الواجهة موضوعة في ملف في ضيغة xml، يتم تحديده في مدخل الدالة في هذا المثال R. l ayout . mai n

## 4.3 التنقل بين الشاشات عن طريق Intent

افترض ان لديك برنامج فيه ثلاث شاشات او واجهات، كيف تتم عملية التنقل بين الشاشات المختلفة؟، الاندرويد يوفر صنف يسمي Intent وهو المسؤول عن عملية التنقل و نقل البيانات من activityالي اخري، فمثلا اذا كان ال Activity الاولي اسمها FirstAct.java و ال Activityالثانية اسمها SecondAct.java فان عملية التنق من FirstAct.java الي SecondAct.java يتم عن طريق الاسطر الاتية:

```
Intent i=newIntent(this, SecondAct.java)
i.startActivity(i);
```

ففي السطر الاول قمنا بتعريف Object من النوع Intent و قمنا بتحديد ال Activity الحالية , واشرنا اليها

ب this، ثم حددنا اسم ال Activityالتي تريد الانتقال اليها في المدخل الثاني من ال Constructor.

**عند اضافة activity الي برنامج اندرويد يجب ان تعلن عنها في ملف AndroidManafisoFile .**

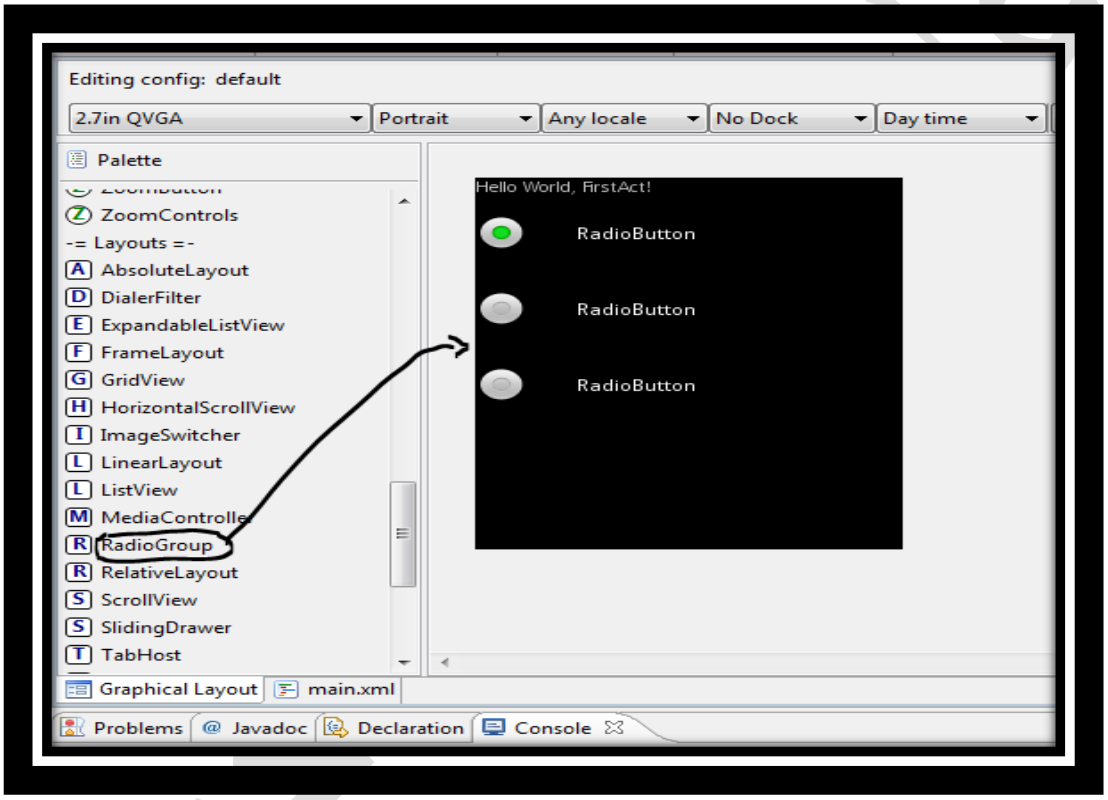
## 4.4 واجهات المستخدم المتقدمة:

### RadioGroup

مهمته هي عرض مجموعة من الازرار الدائرية في شكل مجوعة واحدة بحيث يمكن اختيار زر واحد فقط من ضمن هذة الخيارات، مثال لذلك في شاشة تسجيل بيانات المستخدم، يمكن عرض الدول في شكل

مجموعة Radi oGr oup و المستخدم يختار خيار واحد فقط، حيث لابد للمستخد ان ينتمي الي دولة واحدة.

تعريف واظهار Radi oGr oup يتضمن خطوتين، الاولي اضافة Radi oGr oup في ملف xml l ayout والخطوة الثانية استدعاء Radi oGr oup من صنف ال Act i vi t y بواسطة الدالة f i ndVi ewByI d( ).




الصورة السابقة توضح عملية اضافة Radi o Gr oup الي ملف mai n. xml عن طريق السحب واللصق، لكي تظهر اسطر xml t ags الخاص

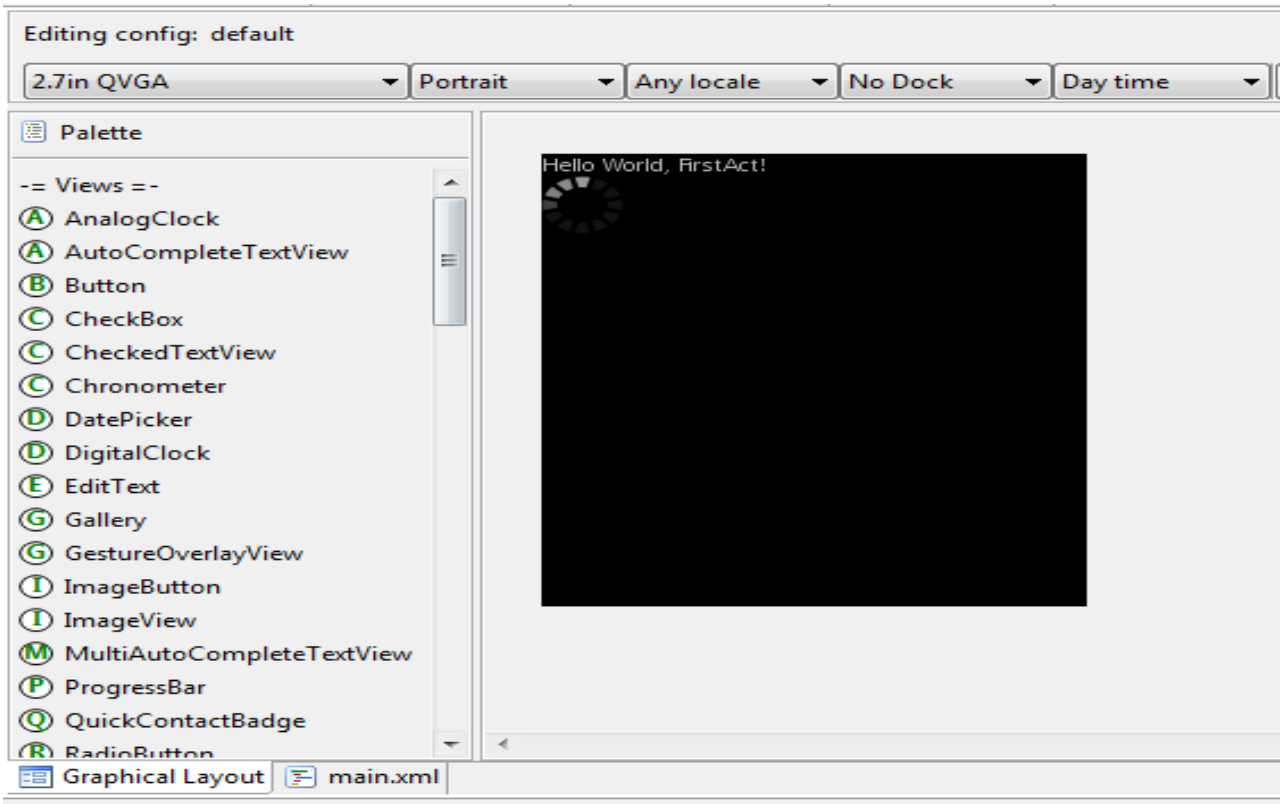
ة ب Radi o Gr oup قم باستعراض mai n. xml ، حيث يحتوي الملف علي xml t ags الاتية:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
   android:orientation="vertical"
   android:layout_width="fill_parent"
   android:layout_height="fill_parent"
   >
<TextView
   android:layout_width="fill_parent"
   android:layout_height="wrap_content"
   android:text="@string/hello"
   />
<RadioGroup android:layout_height="wrap_content"
android:layout_width="fill_parent" android:id="@+id/radioGroup1">
   <RadioButton android:layout_height="wrap_content"
android:layout_width="wrap_content" android:id="@+id/radio0"
android:text="RadioButton" android:checked="true"></RadioButton>
   <RadioButton android:layout_height="wrap_content"
android:layout_width="wrap_content" android:id="@+id/radio1"
android:text="RadioButton"></RadioButton>
   <RadioButton android:layout_height="wrap_content"
android:layout_width="wrap_content" android:id="@+id/radio2"
android:text="RadioButton"></RadioButton>
</RadioGroup>
</LinearLayout>
```

```
@Override

  public void onCreate(Bundle savedInstanceState) {

     super.onCreate(savedInstanceState);

     setContentView(R.layout.main);



     new AlertDialog.Builder(this).setMessage(

"Delete Pet Record?").setPositiveButton(

{                "Delete", new DialogInterface.OnClickListener()

@Override

 public void onClick(DialogInterface dialog,int which) {

"DONE!",   Toast.makeText(FirstActivity.this,
```

```
<?xml version="1.0" encoding="utf-8"?>

<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"

   android:orientation="vertical"

   android:layout_width="fill_parent"

   android:layout_height="fill_parent"

   >

<TextView

   android:layout_width="fill_parent"

   android:layout_height="wrap_content"

   android:text="@string/hello"

   />

<ProgressBar android:id="@+id/progressBar1"
android:layout_width="wrap_content"
android:layout_height="wrap_content"></ProgressBar>

</LinearLayout>
```

ناتج تنفيذ البرنامج موضح في الشكل ادناه:



### 4.5 استخدام ال Font s و ال St yl es:

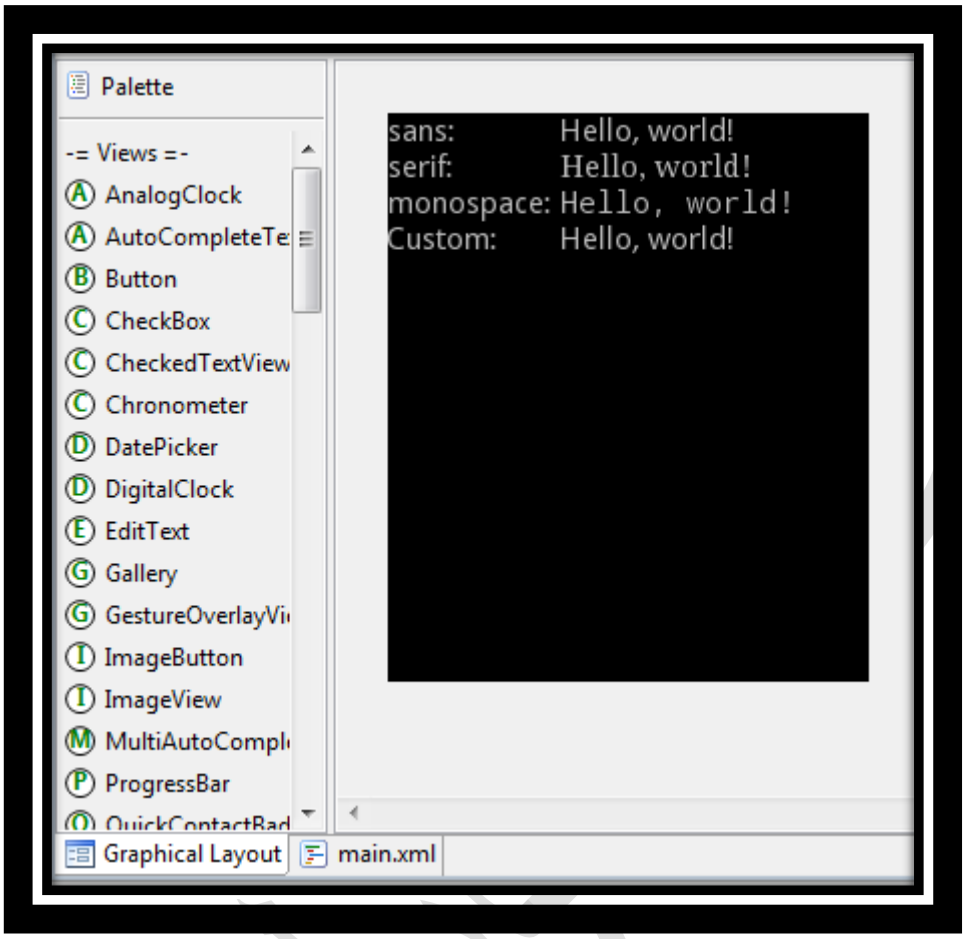
تصميم الخطوط من المواضيع المهمة في تصميم واجهات المستخدم، يجب ان يستخدم نوع موحد من الخطوط في جميع الشاشات، في يلي توضيح كيفية

المثال التالي يوضح بعض انواع الخطوط الأساسية التي يوفرها اندرويد:

```
<?xml version="1.0" encoding="utf-8"?>
<TableLayout
xmlns:android="http://schemas.android.com/apk/res/android"
android:layout_width="fill_parent"
android:layout_height="fill_parent"
android:stretchColumns="1">
<TableRow>
<TextView
android:text="sans:"
android:layout_marginRight="4px"
android:textSize="20sp"
/>
android:id="@+id/sans"
```

```
<TextView
android:text="Hello, world!"
android:typeface="sans"
android:textSize="20sp"
/>
</TableRow>
<TableRow>
<TextView
android:text="serif:"
android:layout_marginRight="4px"
android:textSize="20sp"
/>
<TextView
android:id="@+id/serif"
android:text="Hello, world!"
android:typeface="serif"
```

```
android:textSize="20sp"
/>
</TableRow>
<TableRow>
<TextView
android:text="monospace:"
android:layout_marginRight="4px"
android:textSize="20sp"
/>
<TextView
android:id="@+id/monospace"
android:text="Hello, world!"
android:typeface="monospace"
android:textSize="20sp"
/>
</TableRow>
<TableRow>
<TextView
android:text="Custom:"
android:layout_marginRight="4px"
android:textSize="20sp"
/>
<TextView
android:id="@+id/custom"
android:text="Hello, world!"
android:textSize="20sp"
/> </TableRow>
</TableLayout>
```


Palette
-= Views =-
AnalogClock
AutoCompleteTe
Button
CheckBox
CheckedTextView
Chronometer
DatePicker
DigitalClock
EditText
Gallery
GestureOverlayVi
ImageButton
ImageView
MultiAutoCompl
ProgressBar
Graphical Layout
main.xml
sans: Hello, world!
serif: Hello, world!
monospace: Hello, world!
Custom: Hello, world!

تدريب: قم بالبحث عن طريق عرض كل من Views الاتية:

seekBar

Rating bar

listView

gallery

Notification

اكتب برنامج صعير يحتوي علي عدد من ال Views منظمة بواسطة Table Layout

مبارك احمد

# الباب الخامس: فهم اساسيات تطوير تطبيقات اندرويد

تابع التدريب العملي علي قناتي علي يوتيوب

http://www.youtube.com/c/AdilMobarek

عادل مبارك احمد

# 5 الباب الخامس: فهم اساسيات تطوير تطبيقات اندرويد

في هذا الفصل سوف نتعرف علي اساسيات تطوير تطبيقات اندرويد، كما سنشرح المكونات الاساسية لتطبيق اندرويد وهي Service Activity، Intent , Context.

## 5.1 المكونات الاساسية لتطبيق اندرويد:

### Activity

هي المكون الاساسي لاي برنامج اندرويد،وهي اطار بناء واجهة المستخدم يوضع بداخلة واجهات المستخدم User Interfaces، يمكن ان نعتبر ال Activity هي عبارة عن مربع حوار او نافذة في نظام Windows ومن خلالها يتم التفاعل مع المستخدم، انشاء اي Activity في البرنامج لايختلف عن طريقة انشاء صنف في لغة الجافة، حيث ان Activity ما هي الا صنف يحتوي علي مجوعة من الدوال functions والمتغيرات Variables، حيث ان هذا الصنف يرث من الصنف الاساسي و اسمة Activity. غير ان الاندرويد لديه مفاهيم خاصة في طريقة تنفيذ بعض الدوال المعرف في Activity، والتي تسمي call back functions، هذه الدوال تنفذ بطريقة تلقائيية في كل دورة من دورات حياة ال Activity، ال Activity يشبه صفحة الويب، القرض من صفحة الويب هو اظهار محتويات الصفحة للمستخدم، كذلك ال Activity في

تطبيق اندريد مسؤلة من اظهار واجهة المستخدم، كماء ان تطبيق الويب يمكن ان يحتوي علي صفحة ويب واحدة او اكثر كذلك تطبيق اندرويد يمكن ان يحتوي علي Activityواحدة او اكثر، الاندريد ليه مكون اسمة Activity Manager وهو مسؤول من ادارة Activity ، فعند تنفيذ Activity لاول مرة يتم انشاءها بواسطة Activity Manager، عند الانتقال من هذة Activity الي Activity اخري، يقوم ال Activity Mnager، بتخذين بيانات Activity الاولي حتي يتم الرجوع اليها بشرعة وليس اقلاغها الا في حالة الاحياج الي موارد الهاتف بواسطة Activithyاخري او تطبيق اخر، و عند الاضطرار لاغلاق Activity يتم اغلاق الاكثر اقدمية.

**دورة حياة Activity**

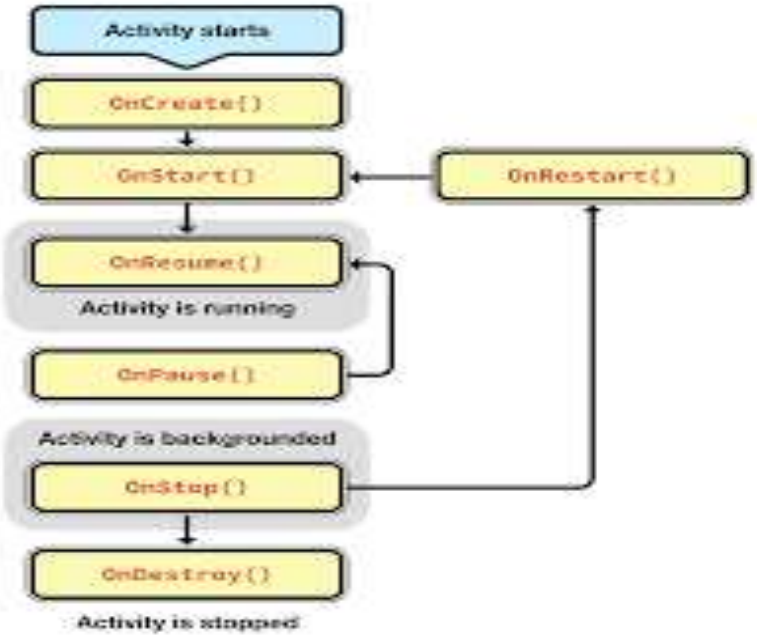
تنفيذ Activity يمر بعدة مراحل تسمي Activity Life Cycle في كل مرحلة هناك دالة تنفذ لتمثيل تلك المرحلة، هذة الدوال موجودة في الصنف Activity ، نقوم بتطبيق الدوال Implement methods حسب حاجة التطبيق.

المثال التالي يوضح الدوال الاساسية لاي Activity:

```
import android.app.Activity;
import android.os.Bundle;
public class FirstActivity extends Activity {
   /** Called when the activity is first created. */
   @Override
   public void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.main);
}
@Override
protected void onDestroy() {
// TODO Auto-generated method stub
```

```
// TODO Auto-generated method stub
super.onPostResume();
}
@Override
protected void onRestart() {
// TODO Auto-generated method stub
super.onRestart();
}

@Override
protected void onResume() {
// TODO Auto-generated method stub
super.onResume();
}

@Override
protected void onStart() {
// TODO Auto-generated method stub
super.onStart();
}
@Override
protected void onStop() {
// TODO Auto-generated method stub
super.onStop();
}
}//End
```

المخطط التالي يوضح مراحل تنفيذ Act i vi t y




**شرح دورة حياة ال Act i vi t y**

onCr eat e

وهي عبارة عن دالة موجودة صمن دوال الصنف Act i vi t y ويتم تنفيذها تلقائيا عند بدائة تشغيل التطبيق، حيث ان هذه الدالة تنفذ في مرة واحدة بعد بداية ال Act i vi t y الموجودة مفي تطبيق اندرويد، انظر الي الصورة اعلاه.

onSt ar t

ايضا تنفذ هذه الدالة بصورة تلقائية بعد الدالة onCr eat e مباشرتا، حتي هذه المرحلة من دورة حياة ال Act i vi t y، لا تعتبر ال Act i vi t yتم تنفيذها، حيث لم تتم رسم واجهة المستخدم

علي الشاشة بعد، ففي هذة الدالة يمكنك وضع الشفرة البرمجية الخاصة بالاتصال بالانترنت، او انشاء ال object التي تحتاجها في لتؤدي ال Activity دورها

onResume

في هذة الدالة تعتبر Activity جاهذة للتفاعل مع المستخدم، حيث تم رسم واجهة المستخدم علي الشاشة

**في المثال التالي سوف نصمم تطبيق اندرويد يحتوي علي Activity واحدة، من خلالها سوف نعرض مربع حوار Dialog:**

عادل مبارك احمد

```
package com.example;

import android.app.Activity;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.os.Bundle;
import android.widget.Toast;

public class FirstActivity extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        new AlertDialog.Builder(this).setMessage(
                "Delete Pet Record?").setPositiveButton(
                "Delete", new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog,int which) {
                    Toast.makeText(FirstActivity.this, "DONE!",
                    Toast.LENGTH_SHORT).show();
                }}).show();
    }
}
```

**مثال: استخدام الدالة OnCl i ck**

معظم التطبيقات تحتاج الي التفاعل مع المستخدم مثلا اذا كان لديك واجه لتسجيل الدخول فيها اسم المستخدم وكلمة السر و زر "دخول" عند الضغط علية يتم الانقال الي الشاشة الرئسية، طبعا عن الضغط علي الزر "دخول" هناك حدث معين يتم تنفيذة Act i on، هذا الحدث ستقوم بكتابة التعليمات البرمجية المناسبة في دالة تكون قد حددتها في ملف الواجه في ملف XML Layout، فيما يلي مثال لل Layout Fi l e

```
<?xml ver si on=" 1. 0"  encodi ng=" ut f - 8" ?>
<Tabl eLayout
 xml ns: andr oi d=" ht t p: / / schemas. andr oi d. com/ apk/ r es/ andr oi d"
 andr oi d: l ayout _hei ght =" f i l l _par ent "
andr oi d: l ayout _wi dt h=" f i l l _par ent "
andr oi d: backgr ound=" #FFFFFF"
 >
 ..........................................................................
                                                    ..........
                                  . .
                                  . .
                                  . .
<But t on andr oi d: l ayout _wi dt h=" 40di p"  andr oi d: onCl i ck=" l ogi n"
andr oi d: t ext =" دخول" andr oi d: l ayout _hei ght =" 50di p"
andr oi d: i d=" @+i d/ but t on1" ></ But t on>

 </ Tabl eLayout >
```

نفترض اننا نريد عن الضغط علي **But t on يتم عرض الرسال " تم الدخول بنجاح" المثال الثالي يوضح ذلك: (لاحظ سوف نقوم بكتابة كود الدالة l ogi n التي حددنا اسمها في ملف ال Layout xml)**

```
package com.example;
import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

public class MainActivity extends Activity {



  @Override
  public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.login);
  }
// سيتم تنفيذ هذة الدالة عند الضغط علي الزر "دخول"
  public void login(View view) {
      Toast.makeText(this, "تم الدخول بنجاح ", Toast.LENGTH_SHORT).show();
  }

  @Override
  protected void onActivityResult(int requestCode, int resultCode, Intent
data) {


     }
    }
```

```
  }
 }
}
```

## Service 5.2:

عادتا ما تحتاج استخدام ال service لتنفيذ المهام التي تتم بطريقة منتظمة و دورية، و المهام التي تاخذ فترة طويلة من الوقت مثل جزء من برنامج يختص بعمل تنبية اذا وصلت رسالة من ال email، دائما يتم استخدام ال service لتنفيذ المهام التي لا تحتاج الي التفاعل مع المستخدم خلافا لل Activity التي تستخدم لتنفيذ المهام التي يتفاعل معها المستخدم.

**مثال لاستخدام ال Service**

**المثال التالي يوضح استخدام ال SERVICE لتشغيل ملف صوتي**

```
// @ auther- Adil Mobarek 2015
package com.ex;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.CheckBox;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.SeekBar;
import android.widget.TextView;

public class FirstAc extends Activity {
/** Called when the activity is first created. */
@Override
public void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.main);


}
public void onclick(View v){

Intent service = new Intent(getApplicationContext(),MyService.class);
service.putExtra("update-rate", 5000);
getApplicationContext().startService(service);

}
}
```

بن ادي احمد

```
//@ auther- Adil Mobrek 2015
package com.example;
import android.app.Service;
import android.content.Intent;import
android.media.MediaPlayer;import android.os.IBinder;import android.util.Log;
import android.widget.Toast;

public class MyService extends Service {
private static final String TAG = "MyService";
MediaPlayer player;
@Override
public IBinder onBind(Intent intent) {
return null;
}

@Override
public void onCreate() {
Toast.makeText(this, "My Service Created", Toast.LENGTH_LONG).show();
Log.d(TAG, "onCreate");
player = MediaPlayer.create(this, R.raw.test);
player.setLooping(false); // Set looping
}

@Override
public void onDestroy() {
Toast.makeText(this, "My Service Stopped", Toast.LENGTH_LONG).show();
Log.d(TAG, "onDestroy");
player.stop();
}

@Override
public void onStart(Intent intent, int startid) {
Toast.makeText(this, "My Service Started", Toast.LENGTH_LONG).show();
Log.d(TAG, "onStart");
player.start();
}
}
```

## 5.3 Intent:

وهي طريقة التواصل بين Activity المختلفة في البرنامج او نداء Activity معينة من برنامج اخر، ساعطيك مثال:

مثلا اذا قمت بتصميم برنامج لعبة فيها ثلاث شاشات كل شاشة عبارة عن Activity وكانت الشاشة الاولي اسمها MainScreen و الشاشة الثانية اسمها Hight Score لعرض اعلي درجة، اما الشاشة الثالثة اسمها Start Game لبدء اللعبة – الطريقة التي سوف تستخدمها للتنقل بين الشاشات المختلفة اسمها Intent.

**مثال لاستخدام ال Intent**

**كيف تبداء تنفيذ activity اخري باستخدام Intent**

```
Intent i = new Intent(this, ActivityTwo.class);
startActivity(i);
```

**انواع ال Intent**

**هناك نوعين من ال Intent**

**اولا : النداء الصريح او تسمي**

## Explicit Intents

**وهنا يتم مباشرتا تحديد اسم ال Activity**

**التي نريد تنفيذها بطريقة صريحة**

**مثال:**

```
Intent i = new Intent(this, ActivityTwo.class);
i.putExtra("Value1", "This value one for ActivityTwo");
i.putExtra("Value2", "This value two ActivityTwo");
```

**النوع الثاني: النداء الضممني او Implicit Intents**

**هنا فقط نقوم بتحديد الحدث الذي نود تنفيذة، بعد ذلك يقوم نظام التشغيل باختيار ال ACTIVITY**

**المناسبة لهذا الحدث**

**مثال لذلك يطلب التطبيق من نظام التشغيل بفتح صفحة ويب :**

```
Intent i = new Intent(Intent.ACTION_VIEW,
Uri.parse("http://www.vogella.com"));
startActivity(i);
```

## نقل البيانات بين ال Intents (Data transfer between activities)

**مثال:**

```
public void onClick(View view) {
 Intent i = new Intent(this, ActivityTwo.class);
 i.putExtra("Value1", "This value one for ActivityTwo");
 i.putExtra("Value2", "This value two ActivityTwo");
 // set the request code to any code you like,
 // you can identify the call back via this code
 startActivityForResult(i, REQUEST_CODE);
}
```

```
Bundle extras = getIntent().getExtras();
if (extras == null) {
 return;
}
// get data via the key
String value1 = extras.getString(Intent.EXTRA_TEXT);
if (value1 != null) {
 // do something with the data
}
```

## Defining intent filters:

```
String url = "http://www.vogella.com";
Intent i = new Intent(Intent.ACTION_VIEW);
i.setData(Uri.parse(url));
startActivity(i);
```

**مثال :**

**تسجيل Activity كمتصفح انترنت**

**الكود ادناة يقوم بتسجيل Activity لل Intent والتي تنفذ عند فتح صفحة انترنت:**

```
<activity android:name=".BrowserActivity"
      android:label="@string/app_name">
 <intent-filter>
   <action android:name="android.intent.action.VIEW" />
   <category android:name="android.intent.category.DEFAULT" />
   <data android:scheme="http" />
 </intent-filter>
</activity>
```

**مثال لاستخدام ال intent**

**في هذا المثال سنقوم بانشاء مشروع جديد، فيه شاشتين رئسيتين هما:**

**الشاشة الاول اسمها Activity1 والشاشة الثانية اسمها Activity2**

1. **انشي مشروع جديد:**

من file اختر new ومن القائمة الفرعية اختر Android Project سوف تظهر لك النافذة التالية قم بملي الحقول التالية:

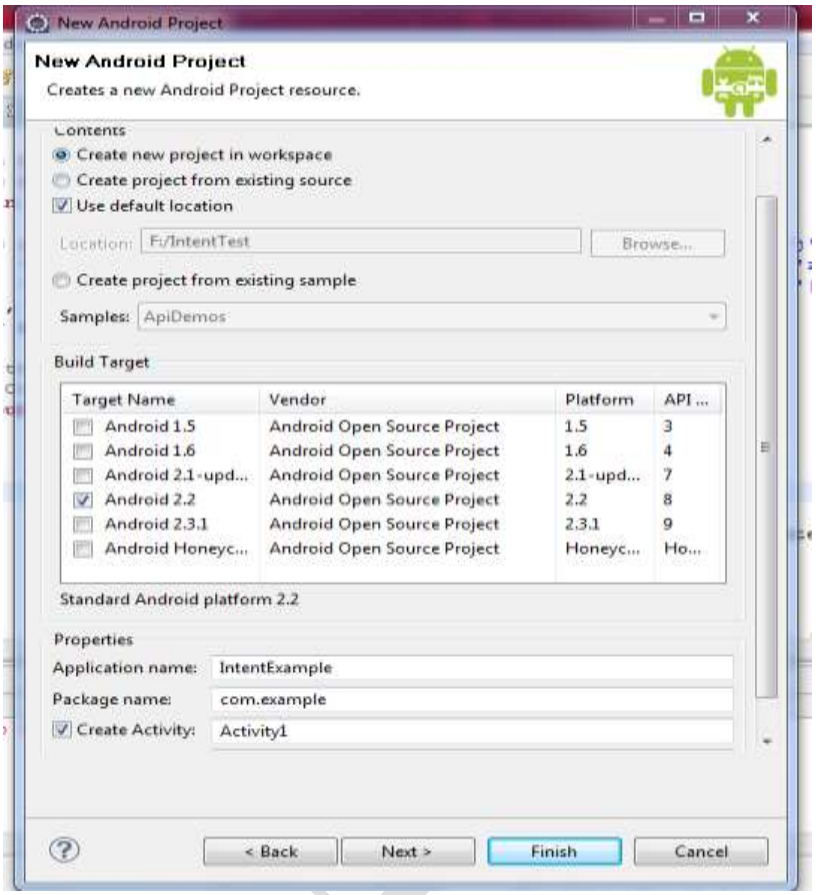
*Project name: Intent Test*

*Application Name: Intent Example*

*Package Name: com example*

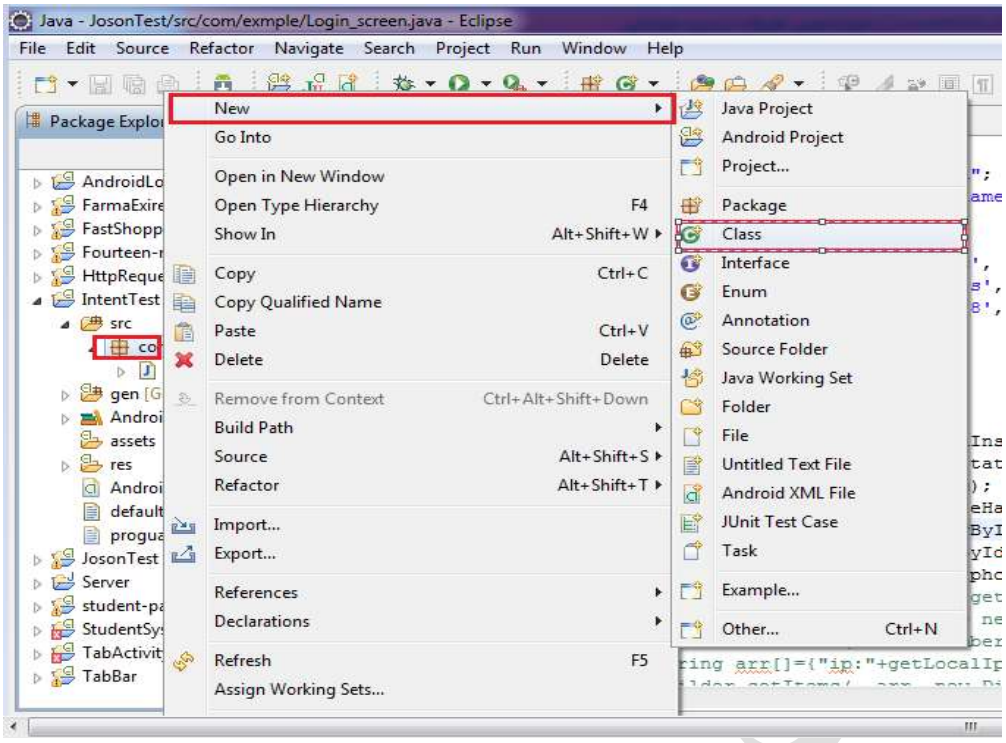
*Create Activity: Activity1*

عادل مبارك احمد

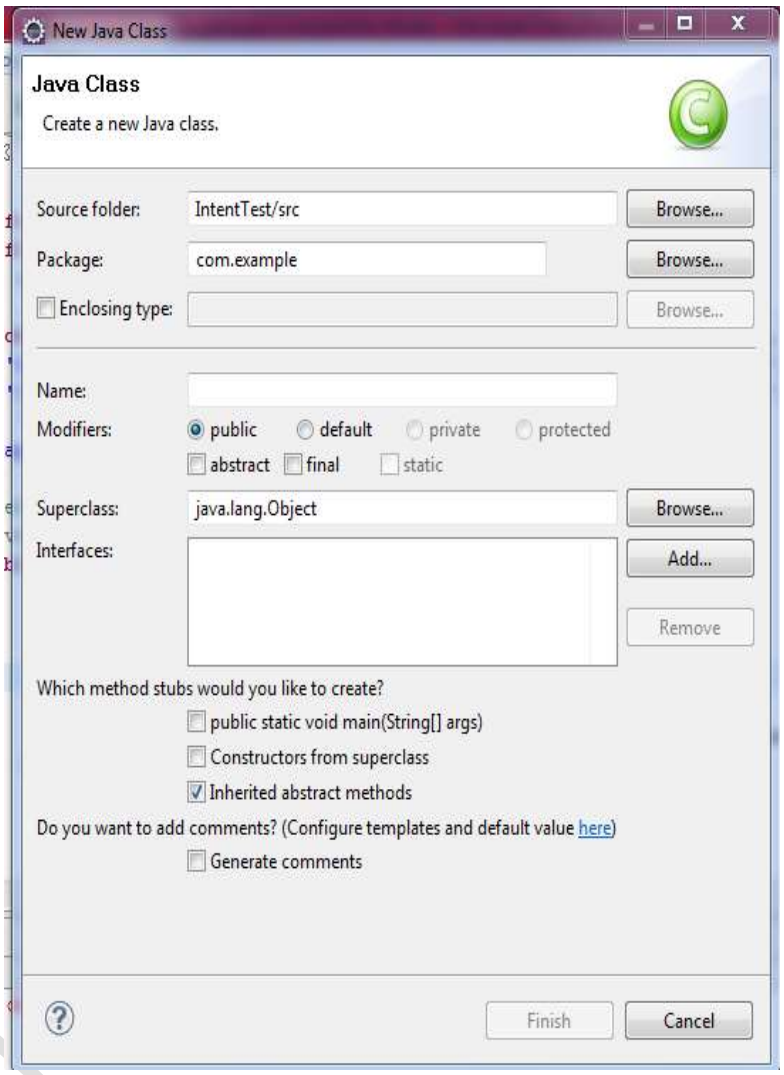



**2. إضافة Activity2 للمشروع**

**لعمل ذلك اقف علي اسم الحذمة com.example واضغط علي ذر الماوس الايمن واختر new class، قم بتسميته Activity2**


Java - JosonTest/src/com/exmple/Login_screen.java - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
New
Go Into
Open in New Window
Open Type Hierarchy F4
Show In Alt+Shift+W
Copy Ctrl+C
Copy Qualified Name
Paste Ctrl+V
Delete Delete
Remove from Context Ctrl+Alt+Shift+Down
Build Path
Source Alt+Shift+S
Refactor Alt+Shift+T
Import...
Export...
References
Declarations
Refresh F5
Assign Working Sets...
Java Project
Android Project
Project...
Package
Class
Interface
Enum
Annotation
Source Folder
Java Working Set
Folder
File
Untitled Text File
Android XML File
JUnit Test Case
Task
Example...
Other... Ctrl+N

اكتب اسم الcl ass واضغط علي زر f i ni sh



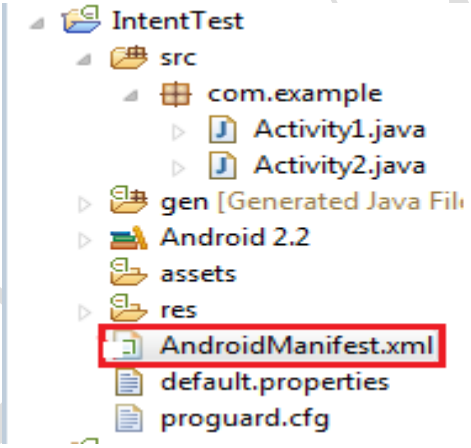

قم بفتح الملف Act i vi t y2. j ava واضف التالي: التعديلات موضحة باللون الاحمر

```
package com.example;

import android.app.Activity;
import android.os.Bundle;

public class Activity2 extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

الان اضفنا Activity2 المشروع، لابد من الاعلان عن هذة ال Activity الجديدة داخل ملف

AndroidManifest.xml ولعمل ذلك اتبع الخطوات التالية:

اولا: افتح ملف AndroidManifest.xml



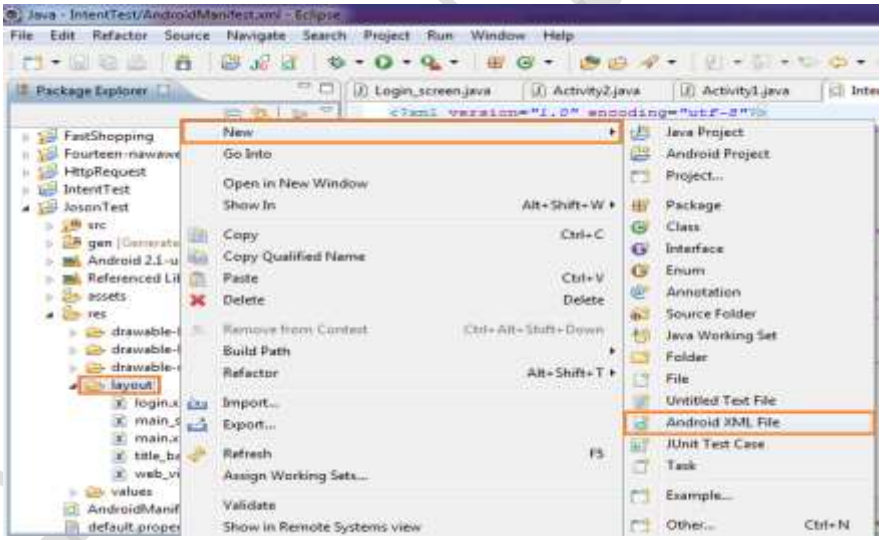

**AndroidManifest.xml**

هم باضافة التالي:

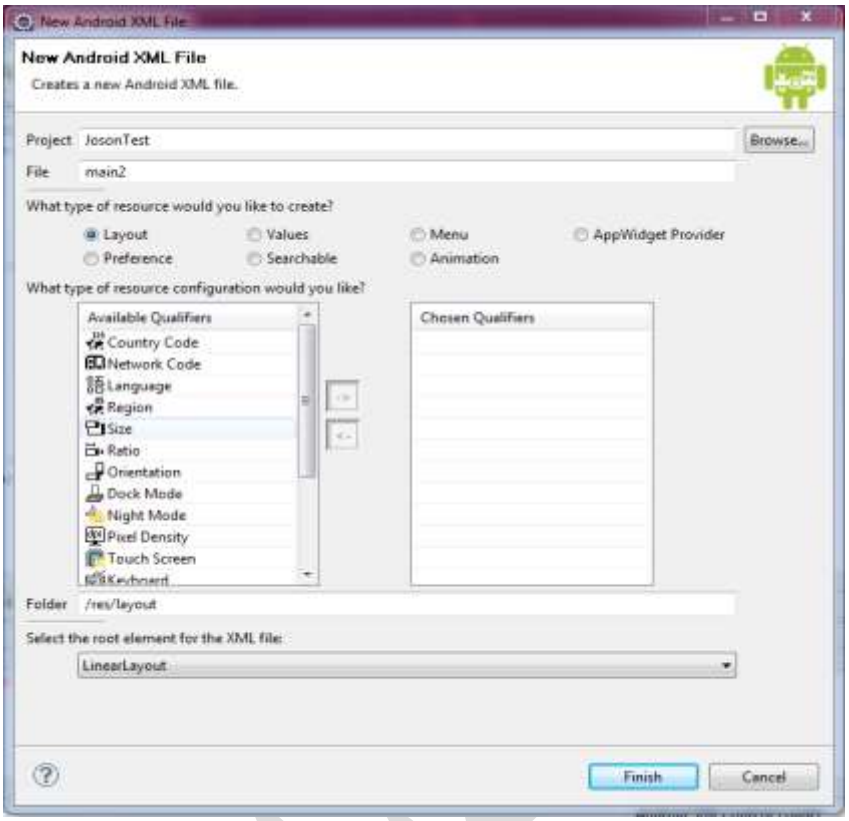
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
      package="com.example"
      android:versionCode="1"
      android:versionName="1.0">

    <application android:icon="@drawable/icon"
android:label="@string/app_name">
        <activity android:name=".Activity1"
                  android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category
android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
             <activity android:name="Activity2"></activity>

    </application>
</manifest>
```

3. اضافة الشاشة الخاصة ب Act i vi t y2

اقف علي ملف l ayout واضغط علي ذر الماوس الايمن تظهر لك الشاشة التالية

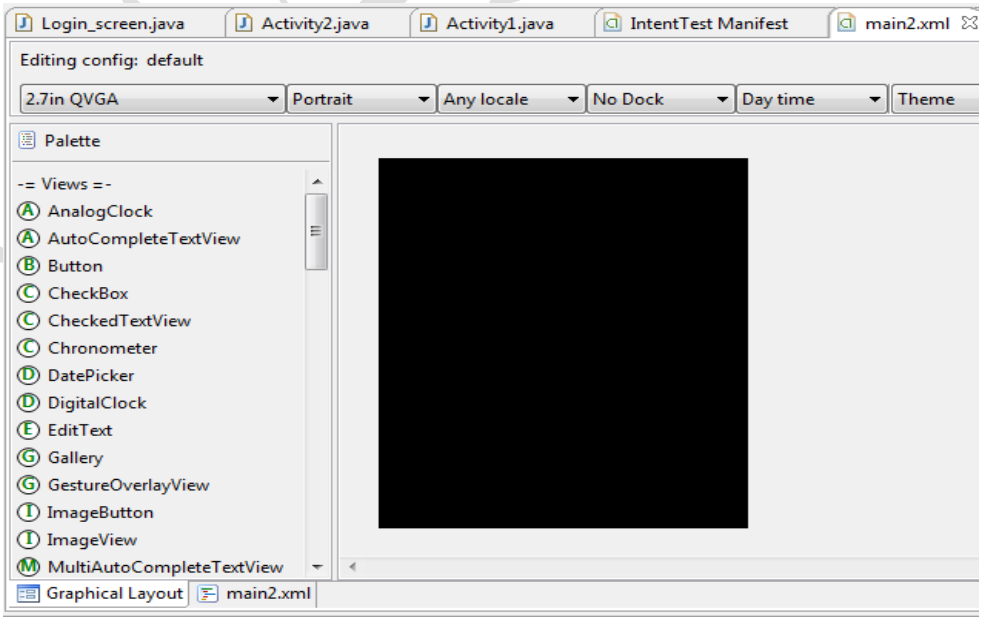



بعد اختيار Andr oi d XML Fi l e تظهر لك الشاشة التالية:




قم بملي حقل Fi l e بالاسم ملف l ayout، في هذا المثال ادخلنا mai n2، بعد الضغط علي در

f i ni sh سوف يتم انشاء ملف الواجهة mai n2، كما موضح في الصورة التالية:



في البدء كما في الصورة ستكون الشاشة فارقة، يجب إضافة vi ews لكي تظهر علي الشاشة، فمثلا اذا ردنا عرض رسالة في الشاشة سوف نقوم بادراج Text Vi ew من قائمة ال vi ews الموجودة علي يسار الصورة اعلاه.

الان بقي اخر خطوة وهي الربط بين ال Act i vi t y2 و ملف الواجهة mai n2. xml الذي قم بانشاءة، لكي تقوم بهذه المهمة، قم بتعديل ملف الجافا Act i vi t y2. j ava، التعديل موضح فيمايلي:

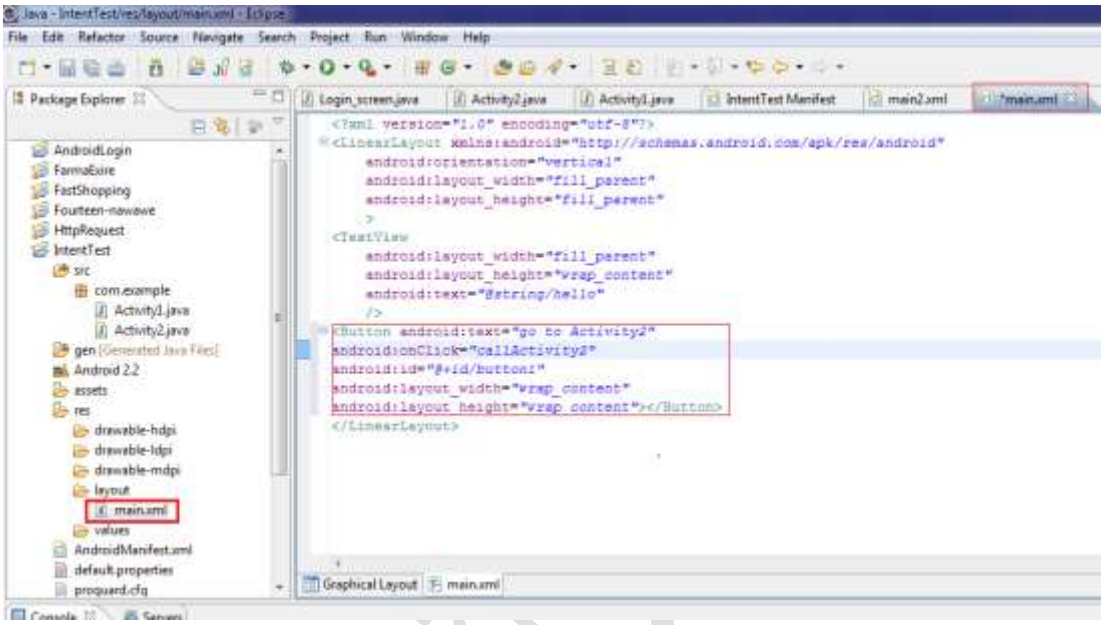
```
package com.example;

import android.app.Activity;

import android.os.Bundle;

public class Activity2 extends Activity {

   /** Called when the activity is first created. */

   @Override

   public void onCreate(Bundle savedInstanceState) {

      super.onCreate(savedInstanceState);

      setContentView(R.layout.main2 ); // we change the conent view from main.xml to main2.xml

   }

}
```

**الان نريد اضافة رز في الشاشة الاولي main.xml و عن\ الضغط علي الزر نقوم بنداء**

**الشاشة الثانية main2.xml:**

خطوتين:

اضف button في main.xml و إضافة خاصية onClick



كتاب دالة onClick في ملف الجافا Activity1.java

```
public void callActivity2(View v){

        Intent i = new Intent(this, Activity2.class);
        startActivity(i);
}
```

بهذه الخطوات نقوم بتنفيذ المشروع، وعند الضغط علي الزر Button شوف يتم الانتقال من الشاشة

الاولي الي الشاشة الثانية عن طريق Intent.

### 5.4 Context:

عبارة عن مركز الاوامر للدوال التي تقوم بتنفيزها، فمثلا اذا كنت في Activity معين في البرنامج واردت تنفيذ دوال موجود في Activity اخري ففهذة الحالة تحتاج استخدام ال Context.

### 5.5 التعرف علي ملف Manifest File

الاساس لاي تطبيق اندرويد هو ال Manifest File يكون ضمن ملفات مشروع اندرويد، اسمه AndroidManifest File في هذا الملف نقوم بتعريف الاشياء التي بداخل تطبيق اندرويد، نقصد بالاشياء هي المكونات الاساسية لاي تطبيق اندرويد وهي Activity، Service، broadcast recover و content provider ، اضافة الي الصلاحيات التي يحتاجها تطبيق اندرويد لكي يعمل علي الجهاز الحقيقي مثل صلاحية الاتصال بالانترنت، عندما تنشي مشروع جديد باستخدام بيئة تطوير مثل eclips، يتم انشاء ملف Manifest بسيط بصورة تلقائية ، بعد ذلك يقوم مطور اندرويد بتعديلة كلما اضاف تعلمات برمجية جديدة تحتاج الي صلاحيات او اضاف واحد من المكونات الاساسية لبرنامج اندرويد، المثال ادناه يوضح ملف AndroidManifest بسيط يحتوي علي بعض ال tags الاساسية:

ملحوظة: ملف Mani f est هو عبارة عن XML Fi l e بالتالي ما هو الا مجوعة من t ag و

At t r i but es .

```
<?xml version=”1.0” encoding=”utf-8”?>
<manifest
xmlns:android=”http://schemas.android.com/apk/res/android”
    package=”com.adil.game”
    android:versionCode=”1”
    android:versionName=”1.0”>
  <application android:icon=”@drawable/icon”
android:label=”@string/app_name”>
     <activity android:name=”.FirstActivity”
           android:label=”@string/app_name”>
       <intent-filter>
          <action android:name=”android.intent.action.MAIN” />
          <category android:name=”android.intent.category.LAUNCHER”
/>
       </intent-filter>
     </activity>

  </application>
</manifest>
```

طبعا ملف Mani f est هو مجموعة من ال t ags وال At t r i but es، t ags الاساسية

الموضحة في المثال

اعلاة هي:

- Manifest tage
- Application tage
- Activity tage

عادل مبارك احمد

عادل مبارك احمد

# الباب السادس: استخدام المكتباب البرمجية الشائعة

تابع التدريب العملي علي قناتي علي يوتيوب

http://www.youtube.com/c/AdilMobarek

# 6 الباب السادس: استخدام المكتباب البرمجية الشائعة

في هذه الوحدة سوف نتعرف علي بعض المكتبات البرمجية الشائعة و المستخدمة بكثرة في تبيقات اندرويد مع شرح بعض الامثلة، في الجزء التالي سوف نتعرف علي المكتبات الخاصة بالاتصال بالشبكة.

## 6. 1 استخدام مكتبات اندرويد الخاصة بالاتصال بالشبكة Andr oi d Net wor ki ng

كتابة تطبيق اندرويد تحتوي علي الاتصال بالشبكة يجعل التطبيق اكثر مرونتا، ويضيف له ميزات كثيرة عن التطبيقات الاخري مثلا اذا كان لديك جزء من العمليات التي تحتاج الي سرعة معالج وذاكرة كبيرة، يمكنك ان تستخدم الاتصال بالشبكة لتكون المعالجة علي الخادم، كذلك الاستفادة من حجم وحدة التخذين العالية لدي الخادم لتخذين البيانات، ففي كثير من الاحيان قد تجد نفسك مضطرا لاستقدام مكتبات الاتصال في الشبكة في تطبيق اندرويد لعدة اغراض منها مثلا لعرض اخر تحديث لبيانات معينة او ارسال البيانات للخادم لمعالجتها او لاخذ البيانات من واجهة المستخدم و تخذينها في الخادم.

لاستخدام البروتوكولات والعمليات المختلفة الخاصة بالاتصال بالشبكة، يوجد العديد من الاصناف cl asses التي يودي هذه المهام موجودة في الحذمة j ava. net ، في هذه الجزئية سوف نركذ علي نوع مشهر جدا في تطبيقات الموبايل وهو Ht t p Pr ot ocol.

### 6. 1. 1 الوصول الي الانترنت عن طريق ht t p

يستخدم هذا البروتوكول لنقل البيانات من والي الشبكة، فيمكنة التعامل مع معظم انواع البيانات، اضافة الي تامين البيانات المرسلة عن طريق SSL.

قراء البيانات من الويب:

قراء البيانات من الويب بسيط جدا، مثلا اذا اردت قراءة بيانات من موقع الكتروني معين ولديك عنوان الويب لهذه البيانات ، يمكنك استخدام URL Class، لقراءة كمية معينة من النصوص من ملف موجود في خادم الويب Web Server، فيما يلي مثال بسيط لقراء البيانات من الويب:

لانك تستخدم بروتوكول HTTP للاتصال بالويب يجب عليك اضافة صلاحية الوصول الي الانترنت عن طريق اضافة Permission في ملف AndroidManafist.xml .

**فتح اتصال http مع خادم الويب باستخدام url:**

نستخدم الصنف HttpURLConnection لفتح الاتصال بالخادم كمافي المثال التالي:

```
HttpURLConnection httpConn =(HttpURLConnection) conn;
httpConn.setAllowUserInteraction(false);
httpConn.setInstanceFollowRedirects(true);
httpConn.setRequestMethod("GET");
httpConn.connect()
response =httpConn.getResponseCode();
    if (response==HttpURLConnection.HTTP_OK) {
    in=httpConn.getInputStream();
    }
```

**الوصول الي حالة الاتصال بالشبكة**

## Retrievingg the android network statues

اداة تطوير تطبيقات اندرويد Android SDK تدعم ادوات لجمع المعلومات عن حالة الشبكة الحالية هذة مهم جدا لتعرف هل الاتصال بالشبكة حاليا متاح قبل ان تحاول استخدام موارد الشبكة، فمثلا اذا اردت ارسال رسالة معينة عن طريق بروتوكول TCP ، لابد من ان تتاكد ان الاتصال تم بالجهاذ الاخر بنجاح، الصنف Class يسمي ConnectivityManager class يوجد لدية مجموعة من الدوال للوصول الي حالة الشبكة الحالية، المثال التالي يوضح هل الشبكة متاحة للاستخدام ومتصلة:

```
import android.net.ConnectivityManager
import android.net.NetworkInfo
... //
ConnectivityManager cm = (ConnectivityManager )
getSystemService(Context.CONNECTIVITY_SERVICE );
  NetworkInfo ni=
cm.getNetworkInfo(ConnectivityManager.TYPE_WIFI )
boolean isWifiAvail = ni.isAvailable() ;
boolean isWifiConn = ni.isConnected() ;
ni = cm.getNetworkInfo(ConnectivityManager.TYPE_MOBILE  )
boolean isMobileAvail = ni.isAvailable ()
boolean isMobileConn = ni.isConnected ()
  status.setText("WiFi\nAvail = "+ isWifiAvail
+ nConn = " + isWifiConn\"
+ nMobile\nAvail = "+ isMobileAvail\"
nConn = " + isMobileConn \"
```

## 6.2 مكتبات اندرويد لانشاء وادارة قاعة البيانات المحلية Local Dat abase

قاعدة البيانات الخاصة بتطبيق معين يمكن ان تكون علي علي الخادم Ser ver او علي زاكرة الهاتف

وهي ما تسمي قاعدة البيانات المحلية، قاعدة البيانات المحلية لها اهمية خاصة في عالم برمجيات الهاتف المحمول، ويرجع ذلك الي طبيعة الهاتف المحمول حيث ان الاتصال بالشبكة في بعض الاوقات غير متاح اضافة الي سرعة نقل البيانات من والي الخادم بطيئة نسبيا، لذلك من الحلول الزكية ان يكون لديك نسخة من قاعدة البيات محليا في ذاكرة الهاتف يتم اللجوء اليها في حالة عدم توفر الاتصال بقاعدة البيانات الاساسية، ويتكم تحديث قاعدة البيانات المحلية من وقت الاخر. المثال التالي يوضح تخذين واسترجاع البينات باستخدام SQLI t e.

عادل مبارك احمد

**اولا: كتاب الصنف Dat abaseHandl er:**

```
import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;
import java.util.ArrayList;
import java.util.List;
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

      public class DatabaseHandler extends SQLiteOpenHelper {

          // All Static variables
          // Database Version
          private static final int DATABASE_VERSION = 1;

          // Database Name
          private static final String DATABASE_NAME = "contactsManager";

          // Contacts table name
          private static final String TABLE_CONTACTS = "contacts";

          // Contacts Table Columns names
          private static final String KEY_ID = "id";
          private static final String KEY_NAME = "name";
          private static final String KEY_PH_NO = "phone_number";

         public DatabaseHandler(Context context) {
              super(context, DATABASE_NAME, null, DATABASE_VERSION);
          }

          // Creating Tables
          @Override
          public void onCreate(SQLiteDatabase db) {
              String CREATE_CONTACTS_TABLE = "CREATE TABLE " +
        TABLE_CONTACTS + "("
                      + KEY_ID + " INTEGER PRIMARY KEY," + KEY_NAME + "
          TEXT,"
                      + KEY_PH_NO + " TEXT" + ")";
              db.execSQL(CREATE_CONTACTS_TABLE);
          }
```

```
// Upgrading database
          @Override
          public void onUpgrade(SQLiteDatabase db, int oldVersion, int
newVersion) {
              // Drop older table if existed
              db.execSQL("DROP TABLE IF EXISTS " + TABLE_CONTACTS);

              // Create tables again
              onCreate(db);
          }

          public void addContact(String phone, String name) {
              SQLiteDatabase db = this.getWritableDatabase();

              ContentValues values = new ContentValues();
              values.put(KEY_NAME, name); // Contact Name
              values.put(KEY_PH_NO, phone); // Contact Phone Number

              // Inserting Row
              db.insert(TABLE_CONTACTS, null, values);
              db.close(); // Closing database connection
          }
           // Getting All Contacts
```

```
        public String getPhone() {
          // Select All Query
          String selectQuery = "SELECT  * FROM " + TABLE_CONTACTS;
         String phone =null;// new String();

          SQLiteDatabase db = this.getWritableDatabase();
          Cursor cursor = db.rawQuery(selectQuery, null);

          // looping through all rows and adding to list
          if (cursor.moveToFirst()) {
              do {
                  phone=cursor.getString(2);
              } while (cursor.moveToNext());
          }


          return phone;
      }
          public int getContactsCount() {
              String countQuery = "SELECT  * FROM " + TABLE_CONTACTS;
              SQLiteDatabase db = this.getReadableDatabase();
              Cursor cursor = db.rawQuery(countQuery, null);
              cursor.close();

              // return count
              return cursor.getCount();
          }


}
```

```
<uses-permission android:name="android.permission.READ_CONTACTS" />
```

**ثانيا: استخدام الصنف DatabaseHandler**

***الخطوة1: انشاء ال object***

```
DatabaseHandler databaseHandler;
    databaseHandler = new DatabaseHandler(context);
```

***الخطوة 2: نداء الدوال***

```
databaseHandler.addContact(phone, "name");
```

عادل مبارك احمد

## 6.3 مكتبات اندرويد للتعامل مع الوسائط المتعددة Andr oi d Mul t i medi a

أ. تشغيل ملف صوتي موجود في ذاكرة الهاتف

```
import android.media.MediaPlayer;
import android.os.Bundle;
import android.os.Environment;
import android.app.Activity;
import android.view.Menu;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;

. public class MainActivity extends Activity {
.    Button start,pause,stop;
.    @Override
.    protected void onCreate(Bundle savedInstanceState) {
.       super.onCreate(savedInstanceState);
.       setContentView(R.layout.activity_main);
.
.       start=(Button)findViewById(R.id.button1);
.       pause=(Button)findViewById(R.id.button2);
.       stop=(Button)findViewById(R.id.button3);
.       //creating media player
.       final MediaPlayer mp=new MediaPlayer();
.       try{
.            //you can change the path, here path is external directory(e.g. sdcard) /Music/
maine.mp3
.       mp.setDataSource(Environment.getExternalStorageDirectory().getPath()+"/Music/mai
ne.mp3");
.
.       mp.prepare();
.       }catch(Exception e){e.printStackTrace();}
.
.       start.setOnClickListener(new OnClickListener() {
.          @Override
.          public void onClick(View v) {
.             mp.start();
.          }
.       });
.       pause.setOnClickListener(new OnClickListener() {
.          @Override
.          public void onClick(View v) {
.             mp.pause();
.          }
.       });
.       stop.setOnClickListener(new OnClickListener() {
.          @Override
.          public void onClick(View v) {
.             mp.stop();
.          }
.       });
.    }
. }
```

ب. تشغيل ملف صوتي من الويب

```
try {
   MediaPlayer player = new MediaPlayer();
    player.setAudioStreamType(AudioManager.STREAM_MUSIC);
    player.setDataSource("http://xty/MRESC/images/test/xy.mp3"
        );
    player.prepare();
          player.start();

 } catch (Exception e) {
   // TODO: handle exception
 }
```

## ❖مراجعة للجافا

تابع التدريب العملي علي قناتي علي يوتيوب

http://www.youtube.com/c/AdilMobarek

عادل مبارك احمد

# مراجعة للجافا

في هذه الوحدة سوف نقوم بمراجعة بعض المفاهيم الرئسية الخاصة بلغة جافا والتي من المؤكد سوف تحتاج استخدامها في تطوير تطبيقات اندرويد، سوف نقوم بمراجعة الدروس المهمة وهي:

1. Variable declaration
2. Java method
3. Class and Object
4. Constructors
5. Inheritance

بما ان اللغة التي سوف نكتب بها تطبيقات اندرويد في هذ الكتاب هي الجافا، وجدت من المهم اضافة هذا الفصل، سوف نشرح المفاهيم السابقة، مع شرح بعض الامثلة.

**اولا: تعريف المتغيرات في الجافا**

المتغير هو عبارة عن اسم لموقع محجوز من ذاكرة الكمبيوتر ram، في لغة البرمجة جافا هناك ثلاث انواع من المتغيرات وهي:

Local Variables

هي المتغيرات التي تم تعريفها داخل دالة method

Instance Variables

هو اي متغير قمت بتعريفه داخل class ولكن خارج الدالة method ، ولم يعرف static.

Static Variables

هو اي متغير عرف static، دائما يكون داخل class، و خرج الدالة method حيث لا يمن تعريف متغير static داخل دالة method .

**مثال يوضح انواع المتغيرات**

```
1.  class A{
2.  int data=50; //instance variable
3.  static int m=100; //static variable
4.  void method(){
5.  int n=90; //local variable
6.  }
7.  }//end of class
```

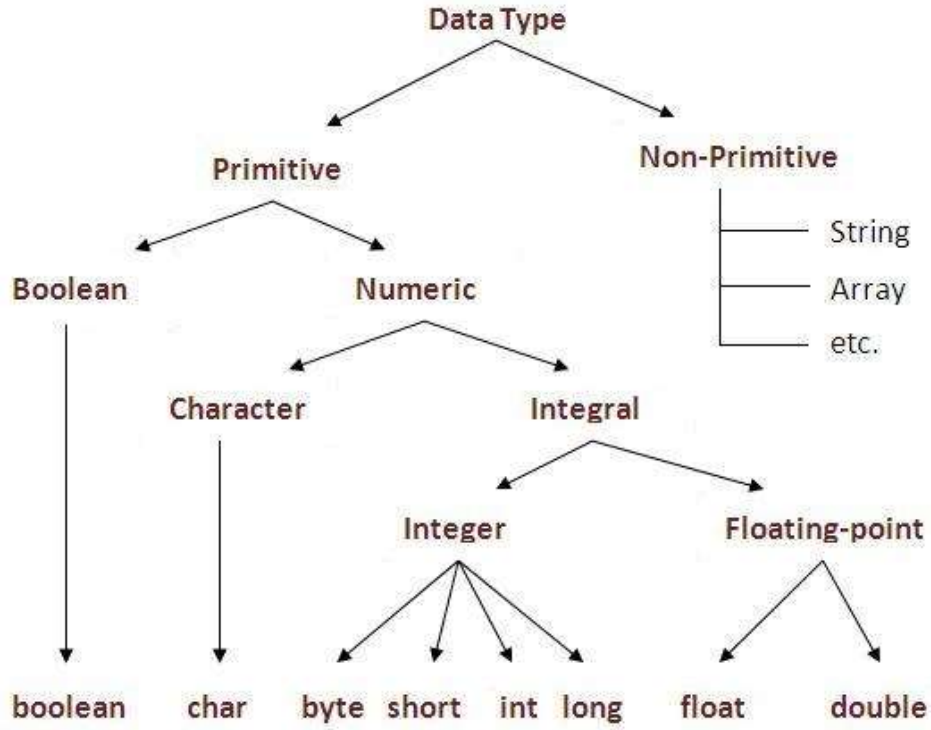
كما ان هناك نوعين رئيسيين من انواع البيانات وهي:

primitive data types

و non-Primitive او complex data types

المخطط التالي يوضح انواع البيانات في لغة البرمجة جافا:

عادل مبارك احمد




من ناحية قواعد تعريف المتغيرات syntax، أي متغير في لغة جافا يحتي علي ثلاث اشياء رئسيبة وهي

إسم المتغير : variable name

نوع البيانات:data type

قيمة المتغير: value

مثال:

```
i nt  x=10;
```

في العبارة اعلاه قمنا بتعرف متغير var i abl e إسمة x، من نوع i nt ،

وقيمته تساوي 10.

مثال: في سطر البرنامج التالي سوف اقوم بتعريف متغير من نوع i nt

```
i nt  x;

x=10;
```

**Doubl e:** عبارة عن متغير رقمي ولاكن يقبل الفاصلة العشرية،

الصيغة:

```
doubl e gr ade=3. 5;
```

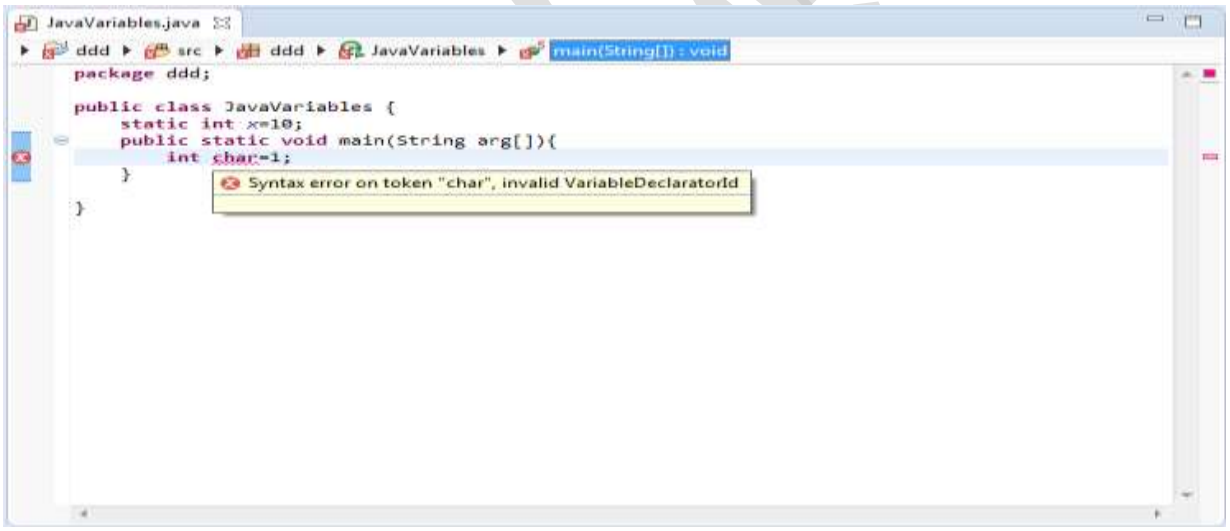
اود ان اشير هنا ان كل المتغيرات في لغة البرمجة جافا هي كلمات محجوزة او تسمي key wor ds، بمعني ان هذه المتغيرات لا يمكن ان تستخدم كاسم

متغير او اسم كلاس او اسم obj ect او اسم دالة، فمثلا اذا قمنا بكتابة

العبارة البرمجية الاتية:

i nt char =1;

سوف تظهر لك رسالة الخطاء الاتية:




- *الصورة اعلاه توضح رسالة الخطاء في بيئة التطوير ecl i pse*

في المثال التالي سوف نقوم بكتابة برنامج جافا يحوي علي صيغة تعريف المتغيرات:

```
publ i c cl ass St udent I nf o{

Publ i c st at i c voi d mai n( St r i ng ar g[ ] ) {

St r i ng st udent Name=”Osman”;  / / St r i ng

i nt  age=1;     / / i nt

doubl e gr ade=3. 5;  / / doubl e

St r i ng

cour ses[ ] ={ “mat h”, ”Ar abi c”, ”Engl i sh”, ”I T”} ;

bool ean pass=t r ue;

char  f i r st Let t er =’o’;

Syst em out . pr i nt l n( “St udent  name: ”+

st udent Name) ;
```

```
System.out.println("first char is: "+
firstLetter);
System.out.println("Student age is: "+ age);
System.out.println("Student courses are: "+
courses);
if (pass)
System.out.println("Student result is:
pass");
else
System.out.println("Student result is:
fail");

}//main
```

عادل مبارك احمد

```
}//class
```

ذكرنا في ما سبق ان المتغير هو عبارة عن اسم موقع في ذاكرة الكبيوتر يحتوي علي قيمة وهي عبارة عن قيمة المتغير، الان قد يتبادر الي ذهنك سؤال وهو ما هو حجم الذاكرة المحجوزة للمتغير، بالتاكيد هذا يعتمد علي نوع النتغير، الجدول التالي يوضح ذلك:

| Data Type | Default Value | Default size |
|---|---|---|
| boolean | False | 1 bit |
| char | '\u0000' | 2 byte |
| byte | 0 | 1 byte |
| short | 0 | 2 byte |
| int | 0 | 4 byte |

| | | |
|---|---|---|
| l ong | 0L | 8 byt e |
| f l oat | 0. 0f | 4 byt e |
| doubl e | 0. 0d | 8 byt e |

## Java Met hods

في الجافا، ال met hod مثل f unct i ons في لغات البرمجة الاخري،

تستخدم لتمثل سلوك ال obj ect.

**فوائد استخدام الدوال :**

- **اعادة استخدام الشفرة البرمجية** Code Reusabi l i t y
- زيادة كفاءة الكود Code Opt i mi zat i on

**مثال:**

```
1.  class Student {
2.   int rollno;
3.   String name;
4.
5.   void insertRecord(int r, String n){  //method
6.    rollno=r;
7.    name=n;
8.   }
9.
10.  void displayInformation(){System.out.println(rollno+" "+name);}//method
11.
12.  public static void main(String args[]){
13.   Student s1=new Student();
14.   Student s2=new Student();
15.
16.   s1.insertRecord(111,"Karan");
17.   s2.insertRecord(222,"Aryan");
18.
19.   s1.displayInformation();
20.   s2.displayInformation();
```

```
21.
22.   }
23.   }
```

# Cl ass and obj ect

ال cl ass هو مجموعة من ال obj ect s التي لها خصائص مشتركة، وهو عبارة عن القالب التي ينشئ منه ال obj ect ،

الوحدة البرمجية الكامة لبرنامج الجافا تسمي cl ass، يبداء ال cl ass في الجافا ، بعبارة cl ass، وفتحة قوس الكلاس، وينتهي، بقوس مقلق يدل علي نهاية الكلاس، و ما بين قوس البداية والنهائة نقوم بتعريف المتغيرات والدوال. مثال

```
1. cl ass <cl ass_name>{
2.    dat a member ;
3.    met hod;
4. }
```

**الcl ass في الجافا يمكن ان يحتوي علي:**

- ✓ المتغيرات dat a member

- ✓ الدوال
- ✓ دالة البناء const r uct o
- ✓ Cl ass وi nt er f ace

يحتوي اي برنامج جافا علي الاقل علي cl ass واحد، ودالة واحدة، ابسط صورة لبرمامج جافا هو ان يحتوي علي cl ass واحد داخلة الدالة mai n، والتي يبداء منها تنفيذ برنامج الجافا.

مثال لبرنامج جافا بسيط:

المثال التالي يوضح برنامج جافا بابسط صورة:

```
1.  publ i c cl ass Hel l Wor l d{
2.      publ i c st at i c voi d mai n( St r i ng
    ar g[ ] ) {
3.      Syst em out . pr i nt ( “Hel l o Wor l d! ”) ;
```

```
4.      }
5.      }
```

## شرح المثال:

في السطر الاول قمنا بتعريف cl ass اسمة Hel l Wor l d ثم قمنا بفتح قوس ال cl ass

في السطر الثاني اضفنا دالة mai n وهي التي يبداء منها تنفيذ البرنامج ونداء الدوال الاخري

في السطر الثالث قمنا بطباعة رسالة علي الشاشة بنداء الدالة pr i nt ( )

في السطر الرابع والخامس قمنا بإقلاغ قوس الدالة mai n ثم قوس cl ass.

*ملحوطة: لتنفيذ البرنامج اعلاة لاب بد من ان يكون اسم الملف بنفس اسم ال cl ass

في لغات البرمجة القديمة التي لاتدم خاصية الobj ect or i ent ed ، كان عملية اعادة استخدام الدوال معقد،

## Java Object

ال object هو عبارة عن نسخة من الclass، يسمي instance، في لحظة التنفيذ، يمكن تشبيه ال class بانه قالب ثلج، اما ال object في مكعب الثلج الذي يمكن استخلاصة من قالب الثلج، بمعني اخر، يمكنني كتابة class واحد فيه خصائص attribute، وسلوك behavior تسمي ال methods ، في كل مرة انشي object من الclass يكون لديه نفس الخصائص ولكن بقيمة مختلفة، كما ان object يكون لدية السلوك الذي حدد في الclass، ولكن بقيم مختلفة، لاهمية هذه المفاهيم، سوف ناخذ مثال عملي لتوضيح الفكرة، مثلا اذا اردنا انشاء برنامج لرسم علي الشاشة مربع، هنا سوف نحتاج الي class واحد ولنقل اسم DrawBox. هنا نريد تحديد ما هي خصائص الكلاس attributes وما هي سلوكه methods ،

الخصائص: طول المربع، عرض المربع

السلوك: عملية الرسم

```
Public class DrawSqr{

Int width;
```

```
Int hight;

Public void draw(){

//drawsqr code

}

}
```

عن انشاء اي object من هذا الclass، يكون لدية، نفس الخصائص، اي لديه طول وعرض، ونفس السلوك اي عملية الرسم علي الشاشة void draw()

الان فهمت!، سوف نشرح كيفية انشاء object من هذا الCLASS

صيغة انشاء object في لغة البرمجة جافا:

الان انشي class اخر اسمه MyDraw وفي داخل دالة main اضف الاسطر الاتية:

```
ClassName objectName=new ClassName();

DrawBox object1=new DrawBox();

Object1.width=10;

Object1.hight=10;
```

```
Object1.draw();
```

## Constructors

تسمي دالة البناء، بمعني ان هذة الدالة تستقبل قيم، من خلال هذة القيمة نقوم ببناء ال object،

قلت ان ال constructor هي دالة لبناء ال object ؟ نعم، ولكن لنها داله لديها خصائص معينة، وهي:

- لها نفس اسم ال class
- ليس لديها نوع بيانات مرجع return type

- تستقبل مدخلات، هذة المدخلات تستخدم لاعطاء القيم لخصائص الclass، التي تسمي class attributes

سوف نقوم بإضافة دالة البناء constructor الي المثال class اعلاة الذي اسمه .drawSqr

```
Public class DrawSqr {

Int width;

Int hight ;

/*coni structor*/

DrawSqr( int width, int hight ){

this.width=width;

this.hight =hight ;

}

Public void draw( ){
```

```
//drawsqr code

}

}

}
```

اذا لم تقوم باضافة constructor الي الclass سوف تم تنفيذ نوع من دالة البناء تسمي دالة البناء الضمني او التلقائي default constructor، وهي عبارة عن اسم الclass وقوسين دائريين، والصيغة العامة كالاتي

```
ClassName object1=new ClassName();
```

*Default constructor*

اما اذا قمت بكتابة constructor داخل الclass فيجب نداءها عن انشاء object بالمدخلات المحددة، ولا سوف تاتيك رسالة خطاء:

```
ClassName object1=new ClassName( argument 1,
                                 argument 2, ...) ;
```

Calling the constructor

## Inheritance

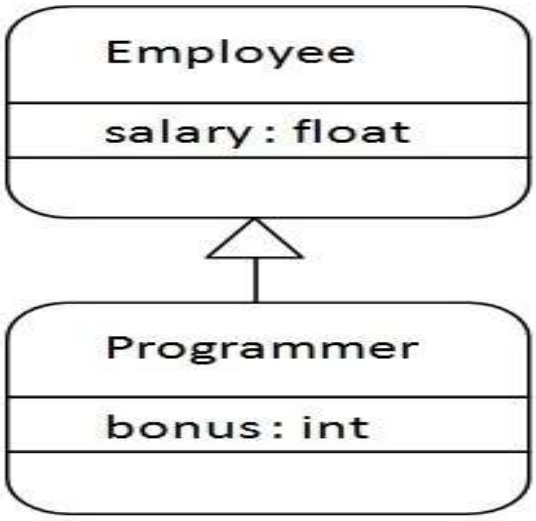
الوراثة في الجافا هي الوسيلة التي يمكن للobject من ان يتحصل علي كل المتغيرات والدوال الموجودة في ال parent object.

الان بقي لدينا موضوع واحد من مواضيع الجافا التي شرحناها في هذا الكتاب، وهو موضوع الوراثة i nher i t ance، مفهوم الوراثة اضاف ميزة جديدة وهي يمكنك ان تستفيد من cl asses اخري قم بكتابتها انت في السابق، او كتبها اي مبرمج اخر، علي سبيل المثال اذا كان لديك cl ass اسمة Per son، خصائصة هي الاسم name والعمرage، واردت كتاب cl ass جديد اسمة st udent ، يمكن ان ترث من الcl ass الذي كتبته في السابق لانشاء الcl ass الجديد st udent:

**الصيغة:**

```
1. cl ass Subcl ass- name ext ends Super cl ass-
   name
2. {
3.    / / met hods and f i el ds
4. }

5. cl ass  Pr ogr ammer  ext ends Empl oyee
```

```
6.  {
7.      / / met hods and f i el ds
8.  }
```




- مخطط يوضح مفهوم الوراثة

يمكن التعبير عن المخطط اعلاة بالبرنامج التالي:

```
1.    cl ass Empl oyee{
2.     f l oat  sal ar y=40000;
```

```
3.    }
4.    class Programmer extends Employee{
5.     int bonus=10000;
6.     public static void main(String args[]){
7.       Programmer p=new Programmer();
8.       System.out.println("Programmer salary
   is:"+p.salary);
9.       System.out.println("Bonus of Programme
   r is:"+p.bonus);
10.   }
11.   }
```

في لغات البرمجة هناك نوعين من الوراثة وهما الوراثة من اب واحد SINGLE INHARITANCE والوراثة المتعدد MULTIBLE

الجافا تدعم النةع الاول فقط من الوراثة وهي SINGLE INHARATICE، وذلك للحفاظ علي اللغة سهلة، لان الوراثة المتعددة تذيد من تعقيد الcode.inharitance

## المراجع

- http://developer.android.com/
- Amdroid Application Development for dummies 2nd Edition
- Android™ Wireless Application Development Second Edition , Shane Conder , Lauren Darcey
- beginning android app development
- **Beginning Android2,** 2010 by Mark L. Murphy.
- Learning Android by Marko Gargenta , 2011 Marko Gargenta.
- - Professional android 4 application development rk, Reto Meier.

- Programming Android, Zigurd Mednieks, Laird Dornin, G. Blake Meike, and Masumi Nakamura
- professional_android_4_application_development u
- http://www.wiseandroid.com/

عادل مبارك احمد

مجال تطوير تطبيقات الهاتف المحمول يعتبر من الفروع الهامة في مجال تقنية المعلومات و علوم الحاسوب، وذلك يعود الي التوسع الكبير في مستخدمين الهاتف المحمول، نحن الآن في عصر تقنية المعلومات، لقد انتقلت معظم بلدان العالم، من ثورة الاتصالات التي تهتم بنقل البيانات من شخص الي آخر، الي ثروة تقنية المعلومات، عزيزي القاري في هذا الكتاب سوف تتعرف علي اساسيات تطوير تطبيقات اندرويد، والتي تهتم ببرمجة وتشبيك الأجهزة النقالة التي تعمل علي نظام تشغيل أندرويد، والتي تضم عدد كبير من الأجهذة منها الهاتف المحمول، الهواتف الذكيه، وبقية الحاسبات النقالة التي تستخدم نظام تشغيل اندرويد. هذا الكتاب موجه الي شريحة المبتدئن في هذا المجال حيث يعطيهم المفاهيم الاساسية التي تمكنهم من الدخول في هذا المجال.

المؤلف: عادل مبارك

© جميع حقوق المؤلف محفوظة – إيداع رقم 2015/977م - الخرطوم

السيرة الذاتية للمؤلف

# Adil Mobarek

Software Engineers

Omdurman, Khartoum
(+249) 123827778 – adil.mobarek@gmail.com

## Work Experience

**Software developer**

*STC* *July 2012 – Present*

Software developer at Sudan Taxation Champer, software developing projects including:

- Data migration software.
- STC Website.
- Chugue cheduling software.

**Mobile Software developer**

*SUST- data center* *May 2011 - March 2012*

Software developer at Sudan University/ data center, software developing projects including:

- Meeting Management System.
- Student Result mobile App.

**Teaching assistant (collaborator)**

Sudan University- College of computer science

May *2011 - March 2012*

## Education

**Degree / Master**

*University of Khartoum- School of math* *15 Jan 2015*

*Software engineering Track.*

**Degree / B.Sc. (HONOUR) In computer Science**

*Sudan University- College of computer science Oct. 2010*

*Second class / division on*

**Degree/ diploma**

*Diploma in network design and Administrator*

*Oracle database Administration, Sudacad training center 2015*

**LINUX OS,** *Linux, Guard for Soft, 2015*
**PL/SQL.** *Guard for Soft, Dec 2015.*
**Oracle Database Administration 11g**
*Oracle Database Administration 11g under Linux, Guard for Soft, 2015.*

***TOT****- at NEW TRAIN training center- Khartoum - 2016*

## Personal Information

**Objective**

To work in a professional environment with a growing organization and utilize my creativity and innovative thinking for benefit of the organization and myself.

**About me**

- Name : Adil Mobarek Ahmed
- Address: Khartoum, SudanOmdurman
- Mobile: 0123827778
  E-Mail: adil.mobarek@gmail.com.
- Date of Birth: 6 Feb 1988

## Skill Set

**Software Engineering**

- Agile Software Development Methodology
- Verification and Validation
- OO Analisis and Design
- System Analisis and Design

**Programming Languages**

- Java se, java ee
- Android,
- SQL, PL/SQL
- HTML, PHP, CSS, Java script and

عادل مبارك احمد