Chapter - 1 INTRODUCTION TO ANDROID

What is Open Handset Alliance (OHA)

The OHA is a consortium whose goal is to develope open standards for mobile device, promote innovation in mobile phones and provide better experience for consumers at a lower cost.

It's a consortium of 84 companies such as google, samsung, AKM, synaptics, KDDI, Garmin, Teleca, Ebay, Intel etc.

It was established on 5th November, 2007, led by Google. It is committed to advance open standards, provide services and deploy handsets using the Android Plateform.

- **Building a better phone for consumers**
  Today, there are 1.5 billion television sets in use around the world. 1 billion people are on the Internet. But nearly 3 billion people have a mobile phone, making it one of the world's most successful consumer products. Building a better mobile phone would enrich the lives of countless people across the globe. The Open Handset Alliance™ is a group of mobile and technology leaders who share this vision for changing the mobile experience for consumers.

- **Innovating in the open**
  Each member of the Open Handset Alliance is strongly committed to greater openness in the mobile ecosystem. Increased openness will enable everyone in our industry to innovate more rapidly and respond better to consumers' demands. Our first joint project as a new Alliance is Android™. Android was built from the ground up with the explicit goal to be the first open, complete, and free platform created specifically for mobile devices.
- **Making the vision a reality**
  Android is not just a blueprint for the faraway future, but a complete platform that will give mobile operators, handset manufacturers, and developers everything they need to build innovative devices, software and services. We are committed to commercially deploy handsets and services using the Android Platform

What is Android?

- Android is an open source and Linux-based **Operating System** for mobile devices such as smartphones and tablet computers.
- Android was developed by the *Open Handset Alliance*, led by Google, and other companies.
- Android offers a unified approach to application development for mobile devices which means developers need only develop for Android, and their applications should be able to run on different devices powered by Android.
- The first beta version of the Android Software Development Kit (SDK) was released by Google in 2007 where as the first commercial version, Android 1.0, was released in September 2008. On June 27, 2012, at the Google I/O conference, Google announced the next

Android version, 4.1 **Jelly Bean**. Jelly Bean is an incremental update, with the primary aim of improving the user interface, both in terms of functionality and performance. The source code for Android is available under free and open source software licenses. Google publishes most of the code under

➢ History of Android

The history and versions of android are interesting to know. The code names of android ranges from A to Q → A to J currently, such as **Aestro**, **Blender**, **Cupcake**, **Donut**, **Eclair**, **Froyo**, **Gingerbread**, **Honeycomb**, **Ice Cream Sandwitch**, **Jelly Bean**, **KitKat**and **Lollipop**. Let's understand the android history in a sequence. marshmallow, Nougat, Oreo, Pie

1) Initially, **Andy Rubin** founded Android Incorporation in Palo Alto, California, United States in October, 2003.

2) In 17th August 2005, Google acquired android Incorporation. Since then, it is in the subsidiary of Google Incorporation.

3) The key employees of Android Incorporation are **Andy Rubin**, **Rich Miner**, **Chris White** and **Nick Sears**.

4) Originally intended for camera but shifted to smart phones later because of low market for camera only.

5) Android is the nick name of Andy Rubin given by coworkers because of his love to robots.

6) In 2007, Google announces the development of android OS.

7) In 2008, HTC launched the first android mobile.

➢ Android Features

Feature	Description
Beautiful UI	Android OS basic screen provides a beautiful and intuitive user interface.
Connectivity	GSM/EDGE, IDEN, CDMA, EV-DO, UMTS, Bluetooth, Wi-Fi, LTE, NFC and WiMAX.
Storage	SQLite, a lightweight relational database, is used for data storage purposes.
Media support	H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC, AAC 5.1, MP3, MIDI, Ogg Vorbis, WAV, JPEG, PNG, GIF, and BMP
Messaging	SMS and MMS
Web browser	Based on the open-source WebKit layout engine, coupled with Chrome's V8 JavaScript engine supporting HTML5 and CSS3.

GSM is Global System for Mobile (Global)
EDGE Enhanced Data Rates for GSM Evolution
IDEN Integrated Digital Enhanced Network

CDMA Code Division Multiple Access
GSM Global System for Mobile
SMS short message service smsc → centre
MMS multimedia messaging service

Multi-touch	Android has native support for multi-touch which was initially made available in handsets such as the HTC Hero.
Multi-tasking	User can jump from one task to another and same time various application can run simultaneously.
Resizable widgets	Widgets are resizable, so users can expand them to show more content or shrink them to save space
Multi-Language	Supports single direction and bi-directional text.
GCM	Google Cloud Messaging (GCM) is a service that lets developers send short message data to their users on Android devices, without needing a proprietary sync solution.
Wi-Fi Direct	A technology that lets apps discover and pair directly, over a high-bandwidth peer-to-peer connection.

➢ **Android Versions, Codename and API**

Alpha Aestro → septembes 23, 2008
Beta Blender → Feb. 9, 2009

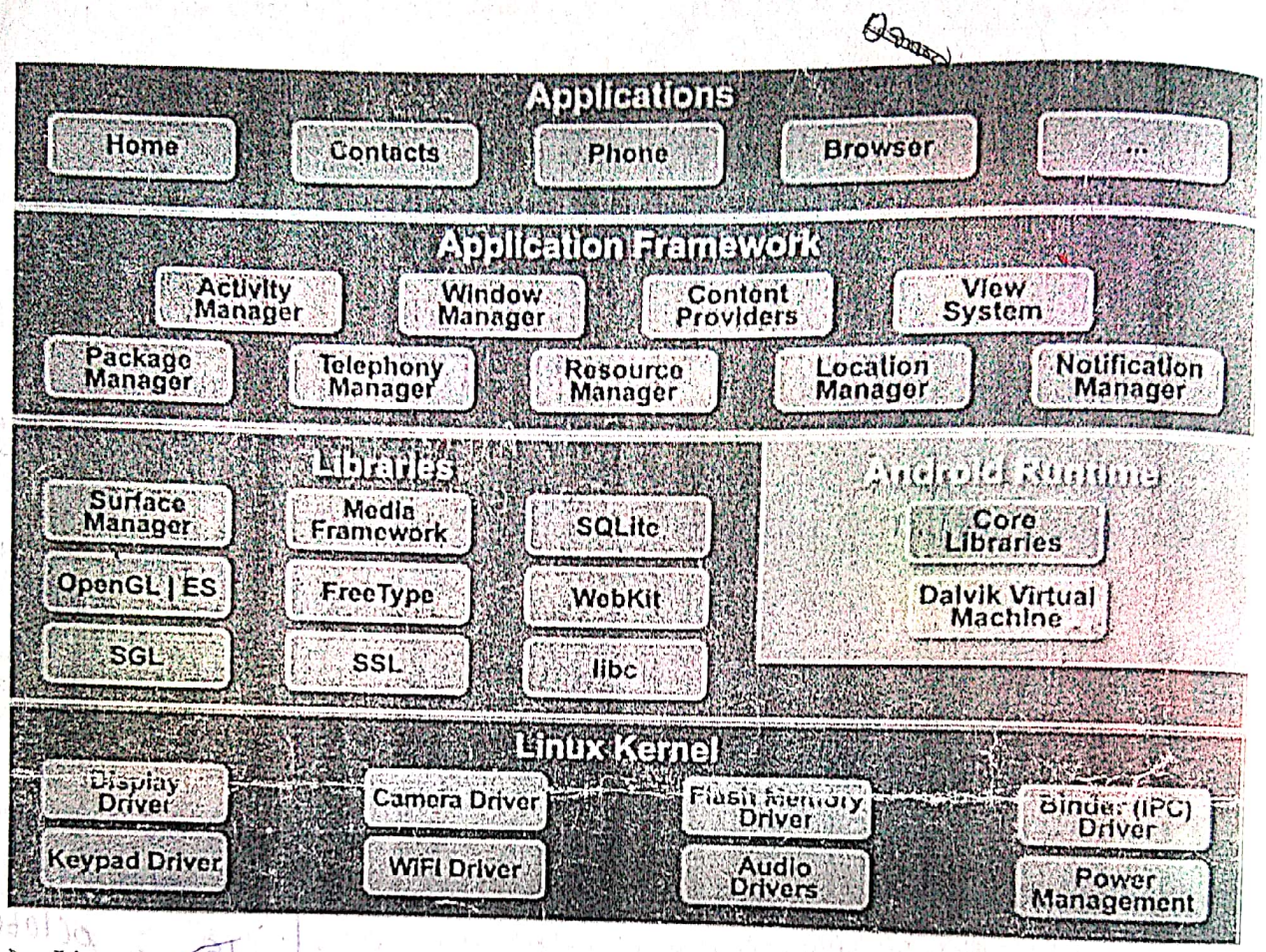
Version	Code name	API Level	
1.5	Cupcake	3	April 27, 2009
1.6	Donut	4	septembes 15, 2009
2.1	Éclair	7	octobes 26, 2009
2.2	Froyo	8	may 20, 2010
2.3	Gingerbread	9 and 10	decembes 6, 2010
3.1 and 3.3	Honeycomb	12 and 13	Feb, 22, 2011
4.0	Ice Cream Sandwitch	15	octobes 18, 2011
4.1, 4.2 and 4.3	Jelly Bean	16, 17 and 18	July 9, 2012
4.4 4.44	KitKat	19	octobes 31, 2013
5.0 5.1.1	Lollipop	21 - 22	Novembes 12, 2014
6.0 6.0.1	Marsh mallow	23	octobes 5, 2015
7.0 - 7.1.2	Nougat	24-25	August 22, 2016
8.0	Oreo	26	August 21, 2017
9.0	Pie	28	August 6, 2018
10.0	Android 10	29	septembes 3, 2019

EV-DO Evolution - Data optimized → Evolution - Data only
UMTS Universal mobile Telecommunications services
WiFi wireless Fidelity
WLAN wireless Local Area Network
LTE Long Term Evolution
3GPP 3rd Generation Partnership Project
NFC Near Field Communication is a short-range wireless connectivity technology.
WiMAX Worldwide Interoperability for microwave Access

IPC + Inter-Process Communication
OpenGL +
EST Embedded System

Android Architecture

→ Android operating system is a stack of software components which is roughly divided into five sections and four main layers as shown below in the architecture diagram.



➢ **Linux kernel**

At the bottom of the layers is Linux - Linux 2.6 with approximately 115 patches. This provides basic system functionality like process management, memory management, device management like camera, keypad, display etc. Also, the kernel handles all the things that Linux is really good at such as networking and a vast array of device drivers, which take the pain out of interfacing to peripheral hardware.

➢ **Libraries**

On top of Linux kernel there is a set of libraries including open-source Web browser engine WebKit, well known library libc, SQLite database which is a useful repository for storage and sharing of application data, libraries to play and record audio and video, SSL libraries responsible for Internet security etc.

➢ **Android Runtime**

This is the third section of the architecture and available on the second layer from the bottom. This section provides a key component called **Dalvik Virtual Machine** which is a kind of Java Virtual Machine specially designed and optimized for Android. The Dalvik VM makes use of Linux core features like memory management and multi-threading, which is intrinsic in the Java language. The Dalvik VM enables every Android application to run in its own process, with its own instance of the Dalvik virtual machine. The Android runtime also provides a set of core libraries which enable Android application developers to write Android applications using standard Java programming language.

SGL - Scalable Graphics Library
SSL - Secure Sockets Layer
libc - Custom C Library

➢ **Application Framework**

The Application Framework layer provides many higher-level services to applications in the form of Java classes. Application developers are allowed to make use of these services in their applications.

~~Applications~~

You will find all the Android application at the top layer. You will write your application to be installed on this layer only. Examples of such applications are Contacts Books, Browser, Games etc.

❖ How to setup Android for Eclipse IDE

In this page, you will learn what softwares are required for running an android application on eclipse IDE. Here, you will be able to learn how to install the android SDK and ADT plugin for Eclipse IDE. Let's see the list of software required to **setup android for eclipse** IDE manually.

1. Install the JDK
2. Download and install the Eclipse for developing android application
3. Download and Install the android SDK
4. Intall the ADT plugin for eclipse
5. Configure the ADT plugin
6. Create the AVD
7. Create the hello android application

1) Install the Java Development Kit (JDK)

For creating android application, JDK must be installed if you are developing the android application with Java language. download the JDK

2) Download and install the Eclipse IDE

For developing the android application using eclipse IDE, you need to install the Eclipse. you can download it from this location download the Eclipse. Eclipse classic version is recommended but we are using the Eclipse IDE for JavaEE Developers.

3) Download and install the android SDK

First of all, download the android SDK. In this example we have installed the android SDK for windows (.exe version).

Now double click on the exe file, it will be installed. I am using the android 2.2 version here.

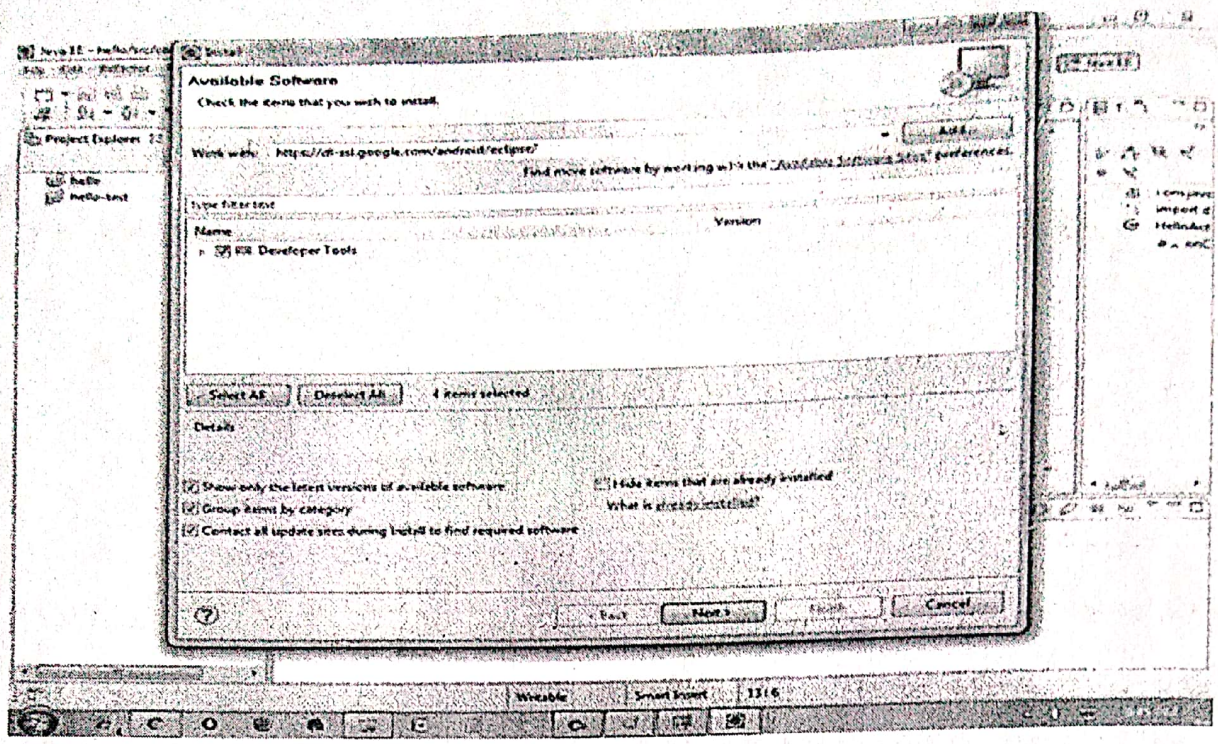
4) Download the ADT plugin for eclipse

ADT (Android Development Tools) is required for developing the android application in the eclipse IDE. It is the plugin for Eclipse IDE that is designed to provide the integrated environment.

For downloading the ADT, you need to follow these steps:

1) Start the eclipse IDE, then select **Help > Install new software...**

2) In the **work with** combo box, write **https://dl-ssl.google.com/android/eclipse/**



3) **select the checkbox** next to Developer Tools and **click next**

4) You will see, a list of tools to be downloaded here, **click next**

5) **click finish**

6) After completing the installation, restart the eclipse IDE

5) Configuring the ADT plugin

After the installing ADT plugin, now tell the eclipse IDE for your android SDK location. To do so:

1. Select the **Window menu > preferences**
2. Now select the android from the left panel. Here you may see a dialog box asking if you want to send the statistics to the google. Click **proceed**.
3. Click on the browse button and locate your SDK directory e.g. my SDK location is C:\Program Files\Android\android-sdk .
4. Click the apply button then OK.

6) Create an Android Virtual Device (AVD)

For running the android application in the Android Emulator, you need to create and AVD. For creating the AVD:

1. Select the **Window menu > AVD Manager**
2. Click on the **new** button, to create the AVD
3. Now a dialog appears, write the AVD name e.g. myavd. Now choose the target android version e.g. android2.2.
4. click the **create AVD**

7) create and run the simple android example

How to make android apps

In this page, you will know how to create the simple hello android application. We are creating the simple example of android using the Eclipse IDE. For creating the simple example:

1. Create the new android project
2. Write the message (optional)
3. Run the android application

❖ Hello Android Example

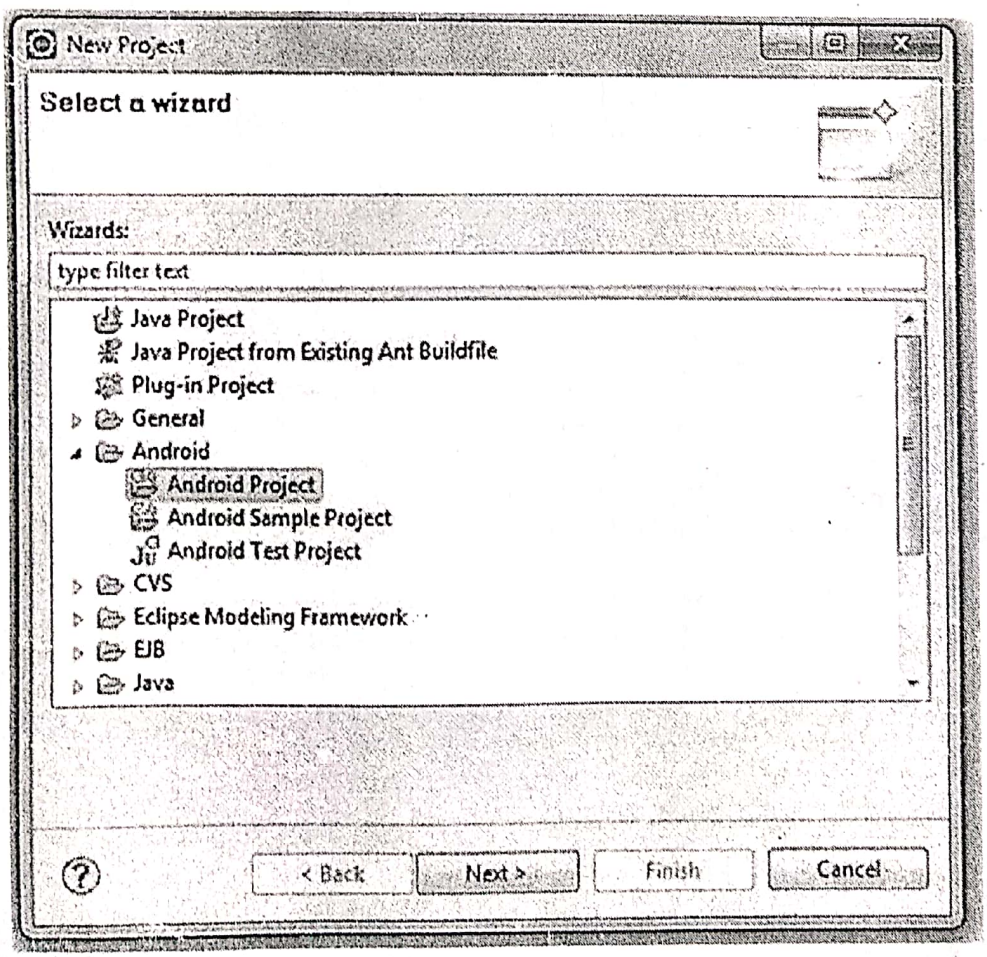
You need to follow the 3 steps mentioned above for creating the Hello android application.

1) Create the New Android project

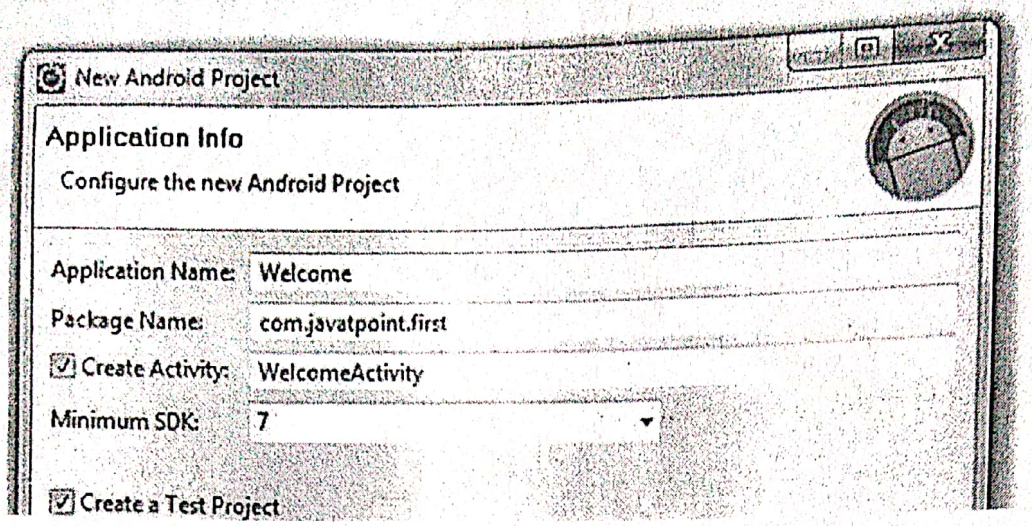
For creating the new android project:

1) Select **File > New > Project...**

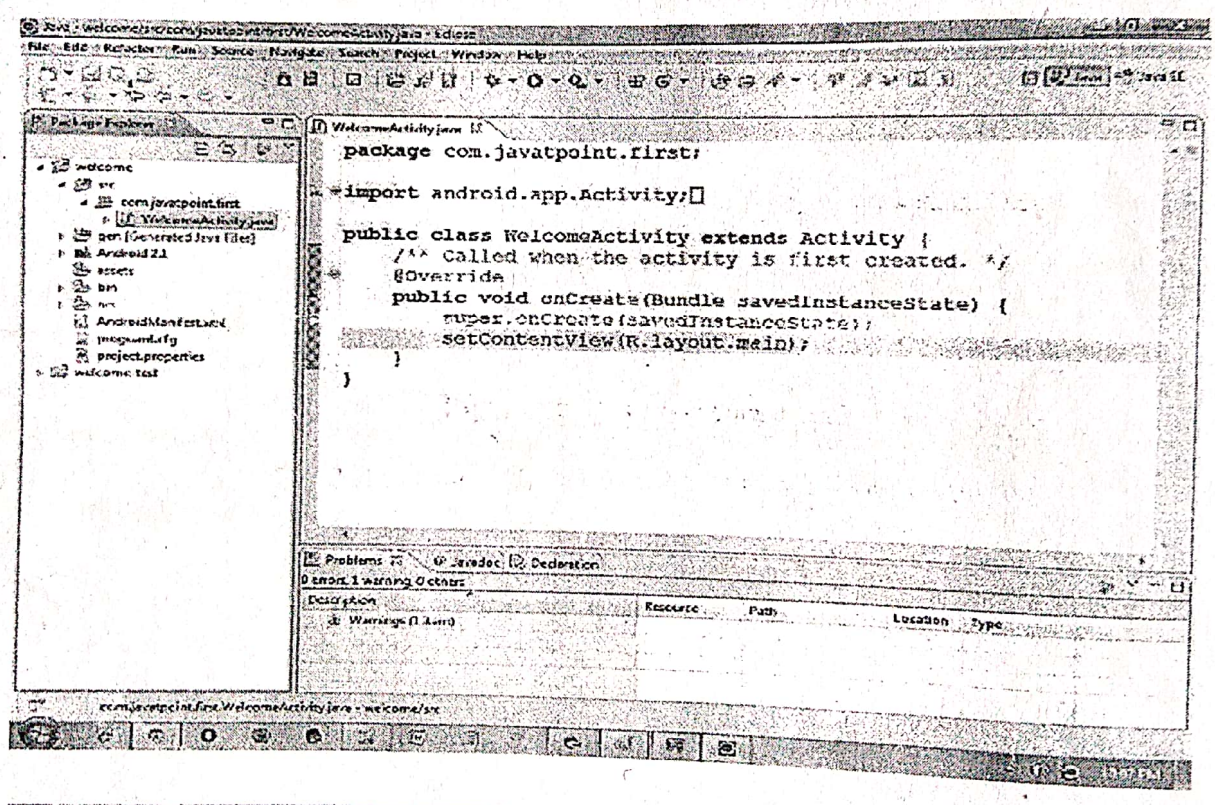
2) Select the android project and click **next**



3) Fill the Details in this dialog box and click **finish**



Now an android project have been created. You can explore the android project and see the simple program, it looks like this:



2) Write the message

For writing the message we are using the TextView class. Change the onCreate method as:

1. TextView textview=new TextView(this);
2. textview.setText("Hello Android!");
3. setContentView(textview);

Let's see the full code of MainActivity.java file.

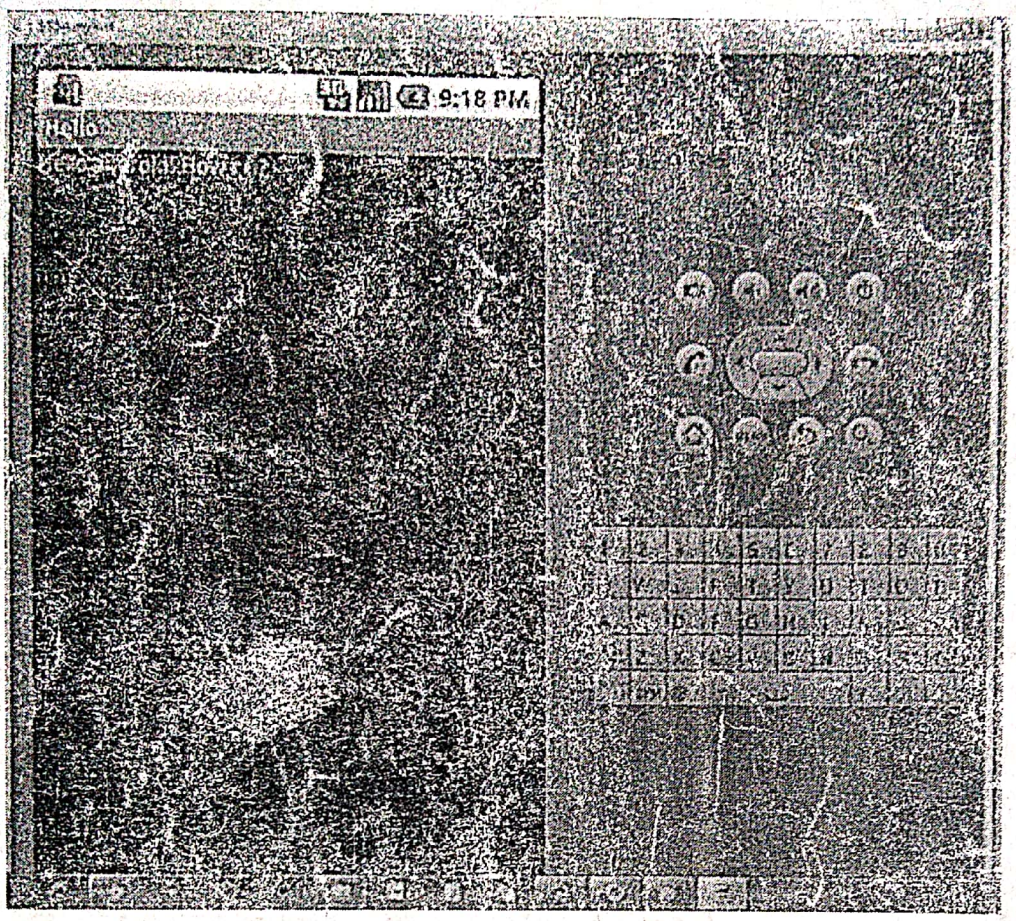
```
package com.example.helloandroid;
import android.os.Bundle;
import android.app.Activity;
import android.view.Menu;
import android.widget.TextView;
public class MainActivity extends Activity {
  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    TextView textview=new TextView(this);
    textview.setText("Hello Android!");
    setContentView(textview);
  }
  @Override
  public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is present.
    getMenuInflater().inflate(R.menu.activity_main, menu);
    return true;
  }
}
```

To understand the first android application, visit the next page (internal details of hello android example).

3) Run the android application

To run the android application: **Right click on your project > Run As.. > Android Application**

The android emulator might take 2 or 3 minutes to boot. So please have patience. After booting the emulator, the eclipse plugin installs the application and launches the activity. You will see something like this:



Types of mobile operating system

1. Symbian → Nokia
2. Android
3. Apple ios
4. Blackberry
5. windows os
6. BADA - Samsung wave, Samsung wave2, & Wave 3
7. Palm
8. maemo
9. MeeGo
10. Vardict