

下载开发资源

1 下载最新的 Android SDK

<http://code.google.com/android/download.html>

选择 Windows 版本的 SDK，下载后直接解压缩就可以使用，为了在控制台中直接使用 SDK 中的工具，把 SDK 的目录添加到 Windows 的 path 环境变量中。

2 下载 JDK6，（JDK5 开发的时候会有一些 Bug）

<http://java.sun.com/javase/downloads/index.jsp>

安装后，设置 J D K 环境变量（java_home,classpath,path）

3，下载 Eclipse 3.2，3.3 版本（开发 Android 的 IDE）

<http://www.eclipse.org/downloads/>

更新 Eclipse，下载 ADT（Android 开发插件）

1. 启动 Eclipse，选择 **Help > Software Updates > Find and Install....**
2. 弹出对话框后，选择 **Search for new features to install** 点击 **Next**.
3. 点击 **New Remote Site**.
4. 在编辑框中输入更新插件的网址：
`https://dl-ssl.google.com/android/eclipse/`. 点击 **OK**.
5. 看到一个更新的列表，点击 **sFinish**.
6. 在返回的对话框中选择 **Android Plugin > Eclipse Integration > Android Development Tools** 并且点击 **Next**.
7. 选择 **Accept terms of the license agreement**, 点击 **Next**.
8. 点击 **Finish**.
9. 点击 **Install All**.
10. 重新启动 Eclipse.
11. 重启 Eclipse 后设置 Android 属性中的 Android SDK 目录(不包括 SDK 下的 Tools, 比如 `D: \android_XXXXX`):

- a. 选择 **Window > Preferences...** 打开编辑属性窗口
- b. 选择 **Android** 属性面板

- c. 加入 AndroidSDK 的目录点击 **Browse...**
 - d. 点击 **Apply**, 和 **OK**
12. 设置 Eclipse 的 JavaJDK 的属性设置。
- A 选择 **Window > Preferences...** 打开编辑属性窗口
 - B 选择 **Java** 属性面板
 - C 选择 Java 编译器为 6.0
 - D 点击 **Apply**, 和 **OK**

使用 Eclipse 开发 Android 应用

使用 EclipseIDE 开发一个 Android 应用。首先是创建一个 Android 工程，然后是设置应用启动配置。然后可以编辑代码，启动，调试应用。

创建一个 Android 工程

ADT 插件提供了一个新的工程向导帮助我们快速的建立 Android 工程和编辑现存的 Android 项目的代码。

按照下面的步骤建立 Android 工程:

1. 选择 **File > New > Project**
2. 选择 **Android > Android Project**, and 点击 **Next**
3. 设置工程的用途
 - 选择 **Create new project in workspace** 可以建立一个新的 Android 工程，输入工程的名字，代码的开发包的名字，应用界面的类名字，应用程序的标题的名字。
 - 选择 **Create project from existing source** 可以根据现存的代码建立一个工程。通过这个选项可以建立并运行现存的 Google 提供的 Example Application （在 Android SDK 的目录中可以找到）
4. 点击 **Finish**.

ADT 插件将会为你的工程建立这些文件和目录:

- **src/** 代码文件存放的目录.
- **res/** 资源文件目录.
- **AndroidManifest.xml** 应用工程的主配置文件.

建立一个启动配置

在使用 Eclipse 调试程序之前，必须为工程建立一个启动配置：

1. 选择 **Run > Open Run Dialog... or Run > Open Debug Dialog...**
2. 在左边的类型中选择 **Android Application** 点击 **New**.
3. 输入一个配置的名字.
4. 在 **Android** 选项中，选择一个应用的界面最为应用启动的主界面.
5. 点击 **Apply** 保存配置，点击 **Run** 或者 **Debug** 开始调试程序.

调试应用程序

一旦建立了工程启动配置就可以按照如下的步骤调试程序：从 Eclipse 的主菜单中选择 **Run > Run or Run > Debug** 。

如果需要当然还可以重新配置启动参数，（比如选择其他的程序界面作为启动的主界面）

Hello, Android!

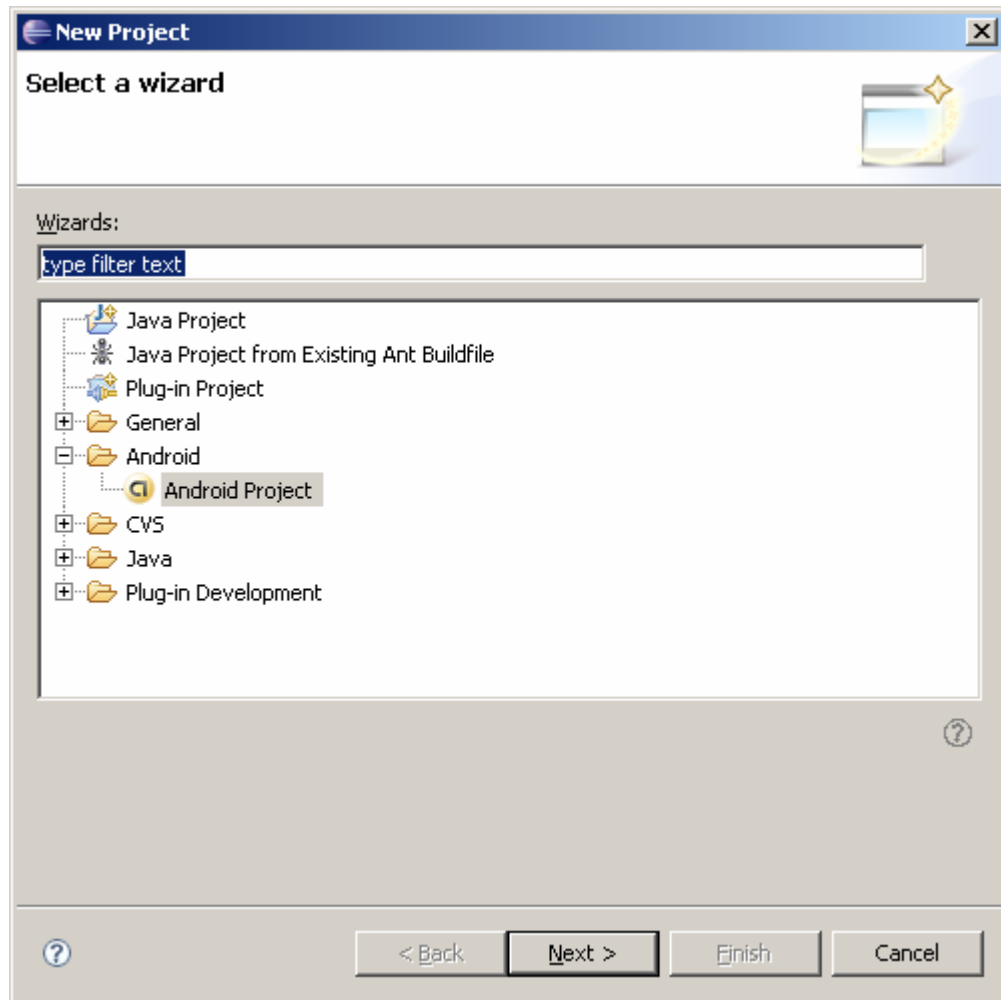
环境配置完成后就可以开始为 Android 编写第一个 HelloWorld 了。一个一般的 Android 应用开发的步骤包括如下几个步骤：

- 建立工程
- 构造 UI
- 运行程序

建立工程

如何建立工程不再赘述，下面是建立 HelloWorld 工程的抓图.

1. **Create a new Android Project**



选择 "Android Project", 点击 Next button.

2. 填入工程的相关的名字

Here's what each field on this screen means:

Project Name	在计算机中存储工程的目录的名字
Package Name	包名—— 参考 Java 相关的概念 比如 com.china.hello
Activity Name	UI 界面窗口的类名，从 Activity 继承而来
Application Name	应用的标题名字

3. 便捷自动产生的代码:

产生的代码类似:

```
package com.china.hello;
```

```

import android.app.Activity;

import android.os.Bundle;

public class HelloChina extends Activity {

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

    public void onCreate(Bundle icle) {

        super.onCreate(icle);

        setContentView(R.layout.main);

    }

}

```

构造 UI

这是最终的 HelloWorldAndroid 的代码，下面逐一解释。

```
package com.china.hello;
```

```

import android.app.Activity;
import android.os.Bundle;
import android.widget.TextView;
public class HelloChina extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle icle) {
        super.onCreate(icle);
        //setContentView(R.layout.main);
        TextView tv = new TextView(this);
        tv.setText("这是一个测试Android的helloWorld");
        setContentView(tv);

    }
}

```

}在类前面添加 import android.widget.TextView; (Java 中的概念，不多做介绍)

在 Android 中，用户界面控件被封装成了各种 **Class** 叫做 **Views**。一个 **View** 是一个可以显示的控件对象，比如 **RadioButton**，**Animation**，**TextLable** 等。其中的一个简单的控件是 **TextView**:

```
TextView tv = new TextView(this);
```

传入 **TextView** 构造函数的参数是一个 **Context** 对象，通过这个对象可以使用系统提供的功能接口，比如加载资源，访问数据库和共享数据等等。**Activity** 类从 **Context** 类继承而来，所以 **Activity** 本身是一个 **Context**（Java 中的继承概念）。**TextView** 对象构建以后就可以设置要显示的数据了。

```
tv.setText("这是一个测试Android的helloWorld");
```

最后是连接 **TextView** 到屏幕，类似这样:

```
setContentView(tv);
```

setContentView() 方法可以控制具体哪一个控件和系统的 UI 联系起来（我的理解是设置为主显示 **View**）。如果没有设置，屏幕中将会显示空白。

运行程序: Hello, Android

点击 **Run** 就可以看到如下的窗口（如何配置启动参数，参考前面）
模拟器启动起来，应用程序启动起来了。



注：还会陆续提供 **Android** 开发的其他的主题的开发文档，欢迎讨论交流。希望本文档能对你有所帮助。