



# Web View 구현





# 학습 목표

## 교육 목표

❖ Web View 구현



Navigation voice



Traffic view



Street View





# Web View

## Web View

### 기능

◆ Web Pages를 보여 주는 기능 제공

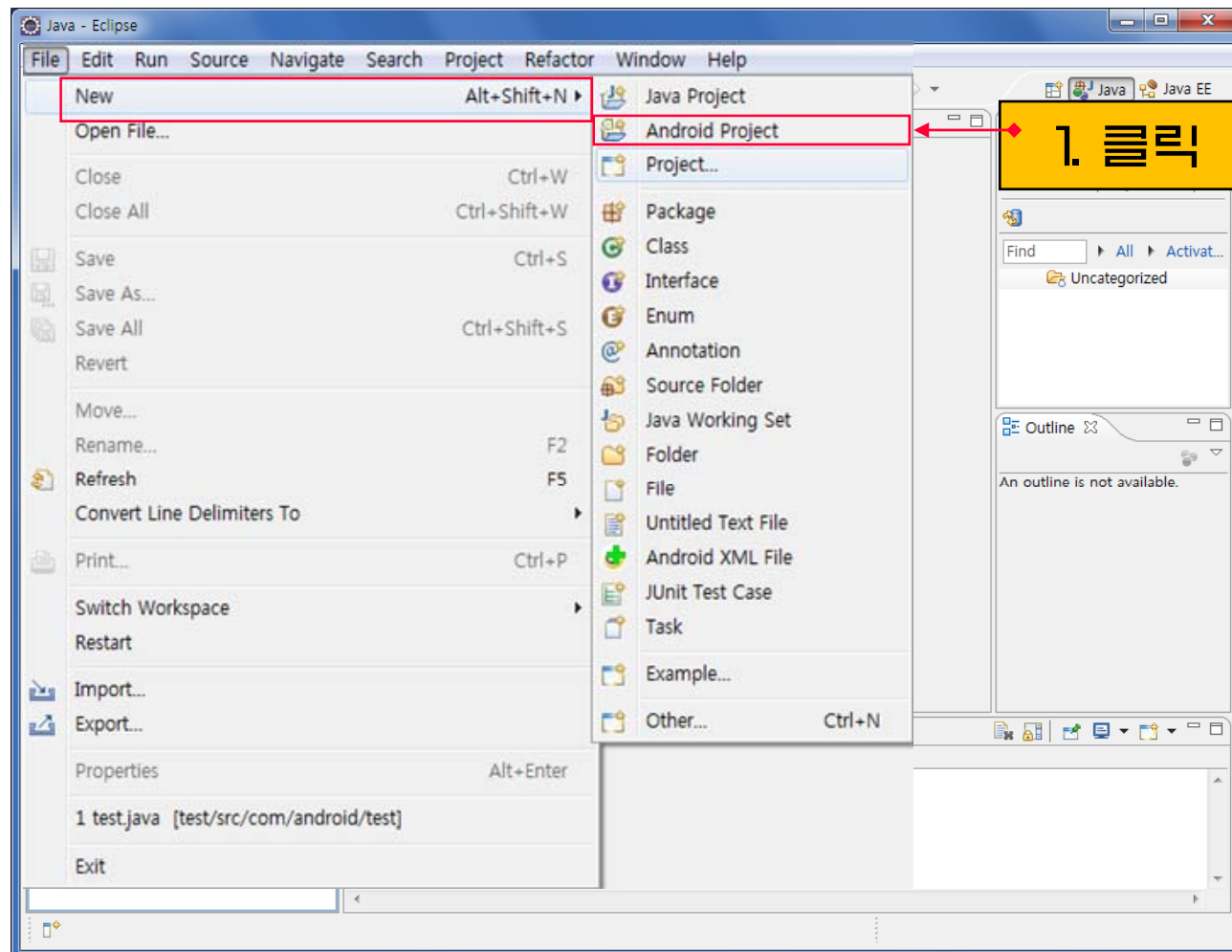




# Web View 구현 (1)

## ■ Android 프로젝트 생성

❖ 프로젝트 명 : Android\_WebView





# Web View 구현 (2)

New Android Project

Creates a new Android Project resource.

Project name:

Contents

☒ Create new project in workspace

☐ Create project from existing source

☒ Use default location

Location:

☐ Create project from existing sample

Samples:

Build Target

Target Name	Vendor	Platform	API ...
<input type="checkbox"/> Android 2.0.1	Android Open Source Project	2.0.1	6
<input checked="" type="checkbox"/> Android 2.1	Android Open Source Project	2.1	7
<input type="checkbox"/> Google APIs	Google Inc.	2.0.1	6

Standard Android platform 2.1

Properties

Application name:

Package name:

☒ Create Activity:

Min SDK Version:

2. Android\_WebView 입력

3. 클릭

4. Android\_WebView 입력

5. com.inhatec.android\_WebView 입력

6. Android\_WebView 입력

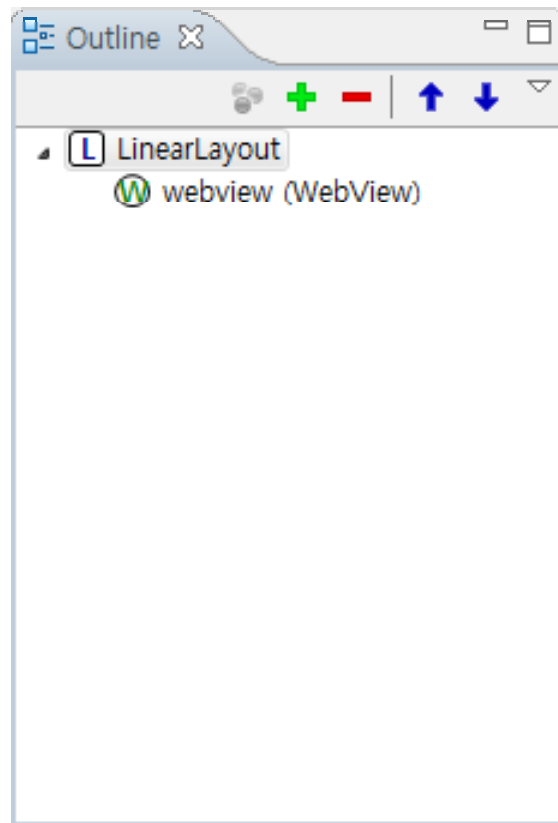
7. 클릭



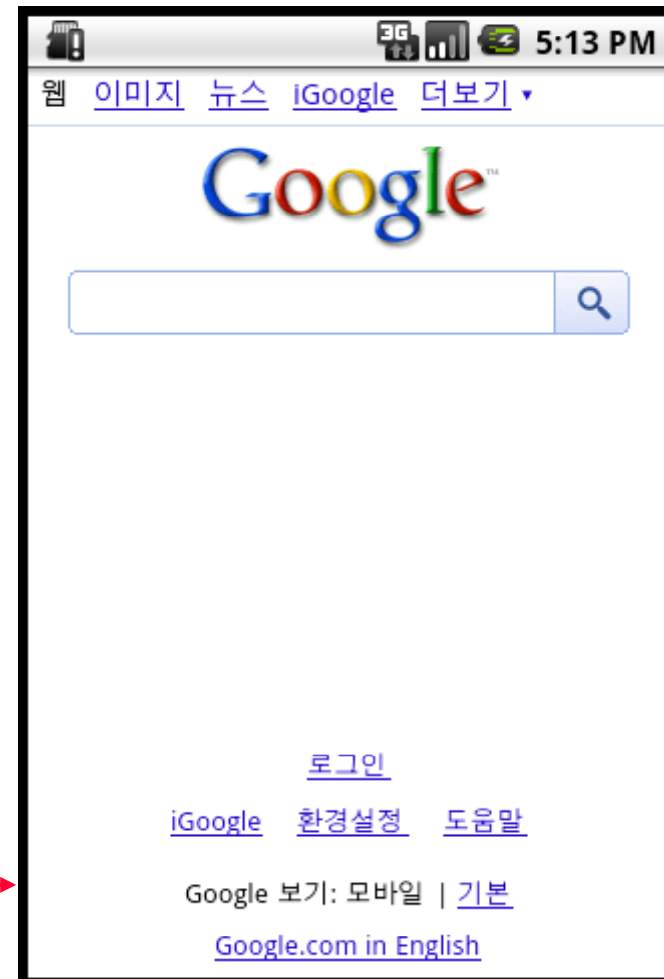


# Web View 구현 (3)

## ■ UI 설계



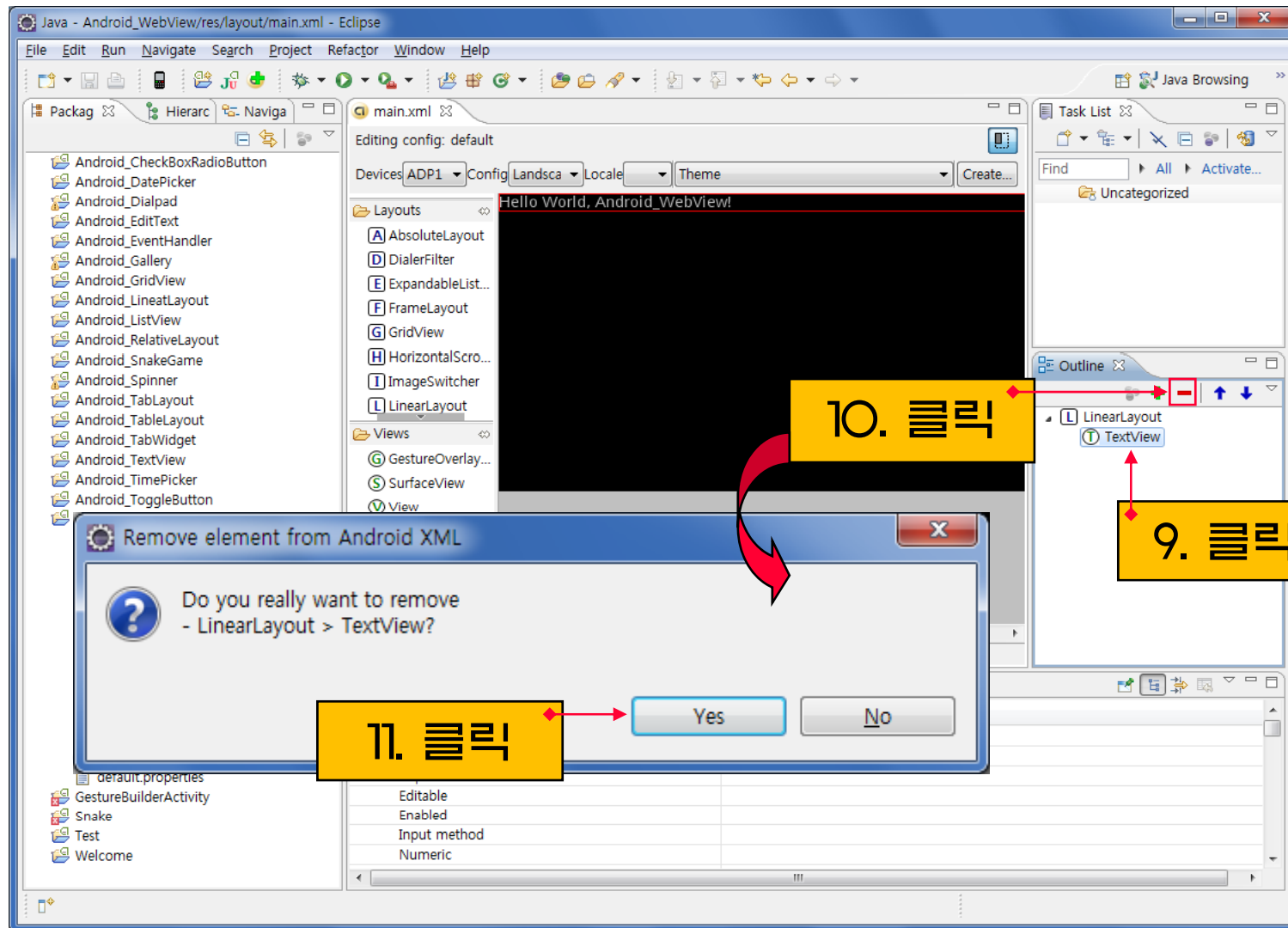
8. UI 설계 및 속성 지정





# Web View 구현 (4)

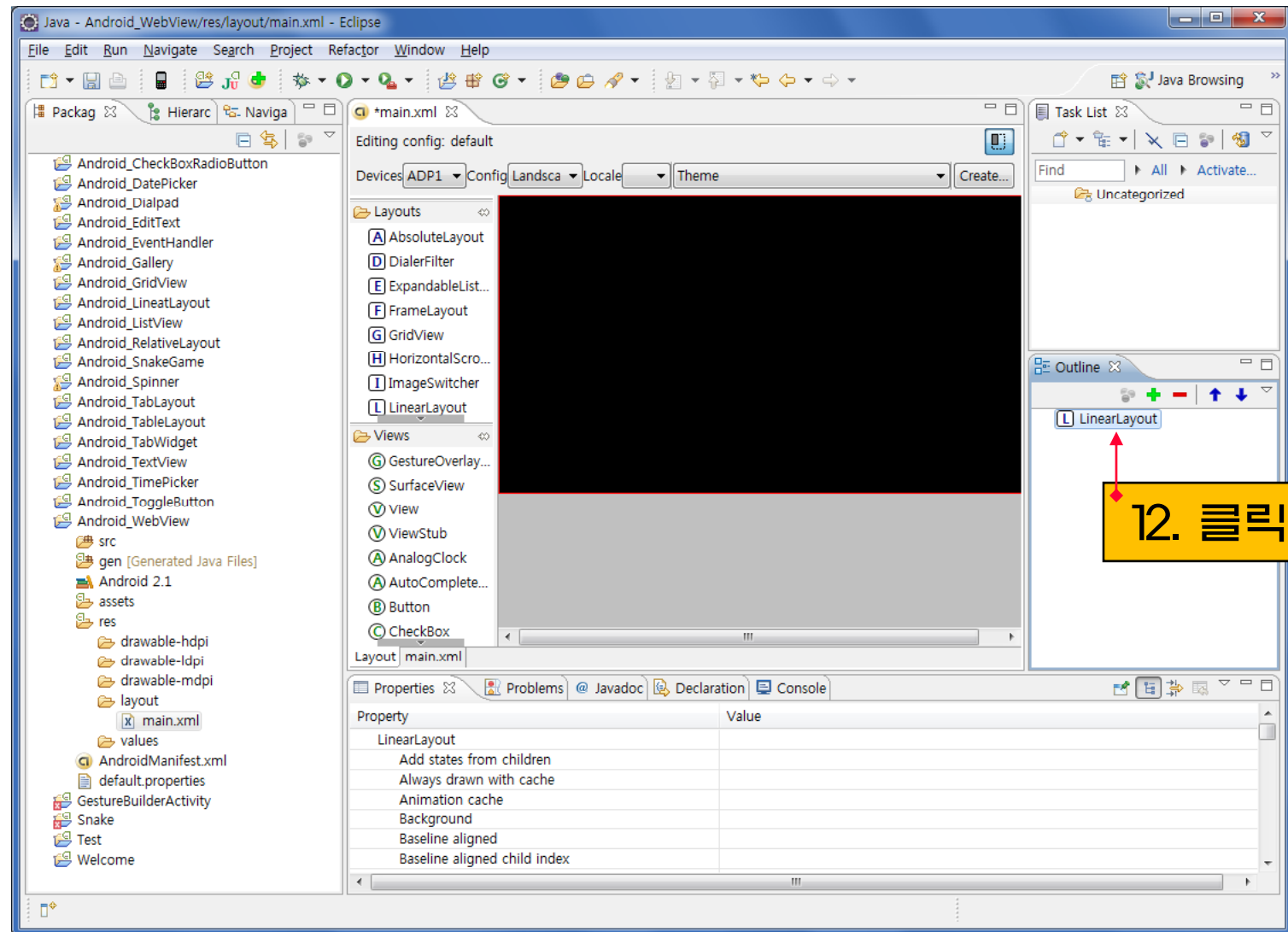
## ❖ Text View 삭제





# Web View 구현 (5)

## ❖ Linear Layout 속성 지정







# Web View 구현 (6)

## ❖ Linear Layout 속성 지정

컨트롤	속성 지정
LinearLayout	<ul style="list-style-type: none"><li>• xmlns:android="http://schemas.android.com/apk/res/android"</li><li>• android:orientation="vertical"</li><li>• android:layout_height="wrap_content"</li><li>• android:layout_width="wrap_content"&gt;</li></ul>





# Web View 구현 (7)

## ■ Main.xml 수정

### ❖ Web View 추가

```
*main.xml
1<?xml version="1.0" encoding="utf-8"?>
2<LinearLayout
3    xmlns:android="http://schemas.android.com/apk/res/android"
4    android:orientation="vertical"
5    android:layout_height="wrap_content"
6    android:layout_width="wrap_content">
7
8    <WebView
9        android:id="@+id/webview"
10       android:layout_width="fill_parent"
11       android:layout_height="fill_parent"
12    />
13</LinearLayout>
```

13. WebView 추가 및 속성 지정

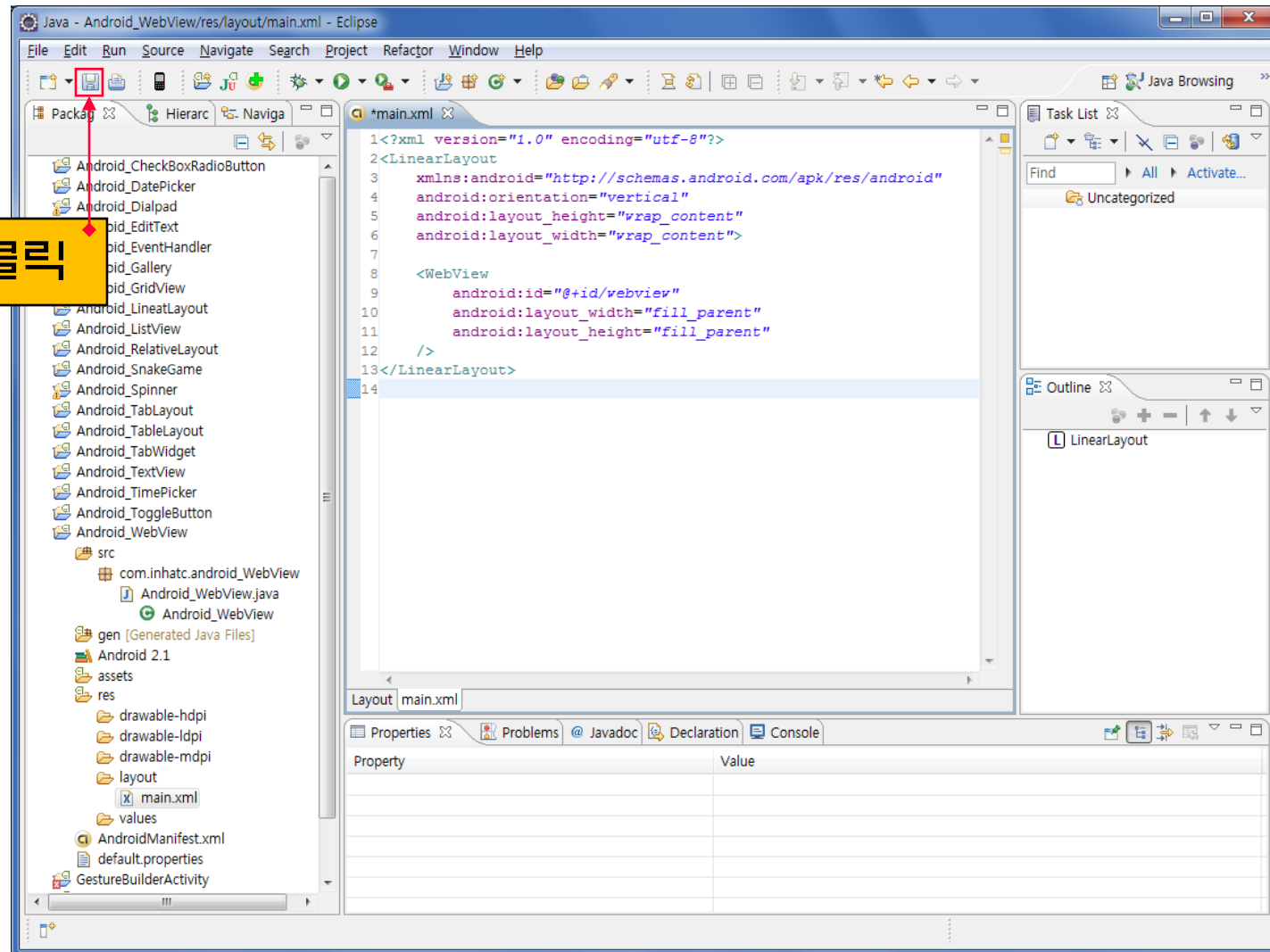




# Web View 구현 (8)

## ❖ Main.xml 저장

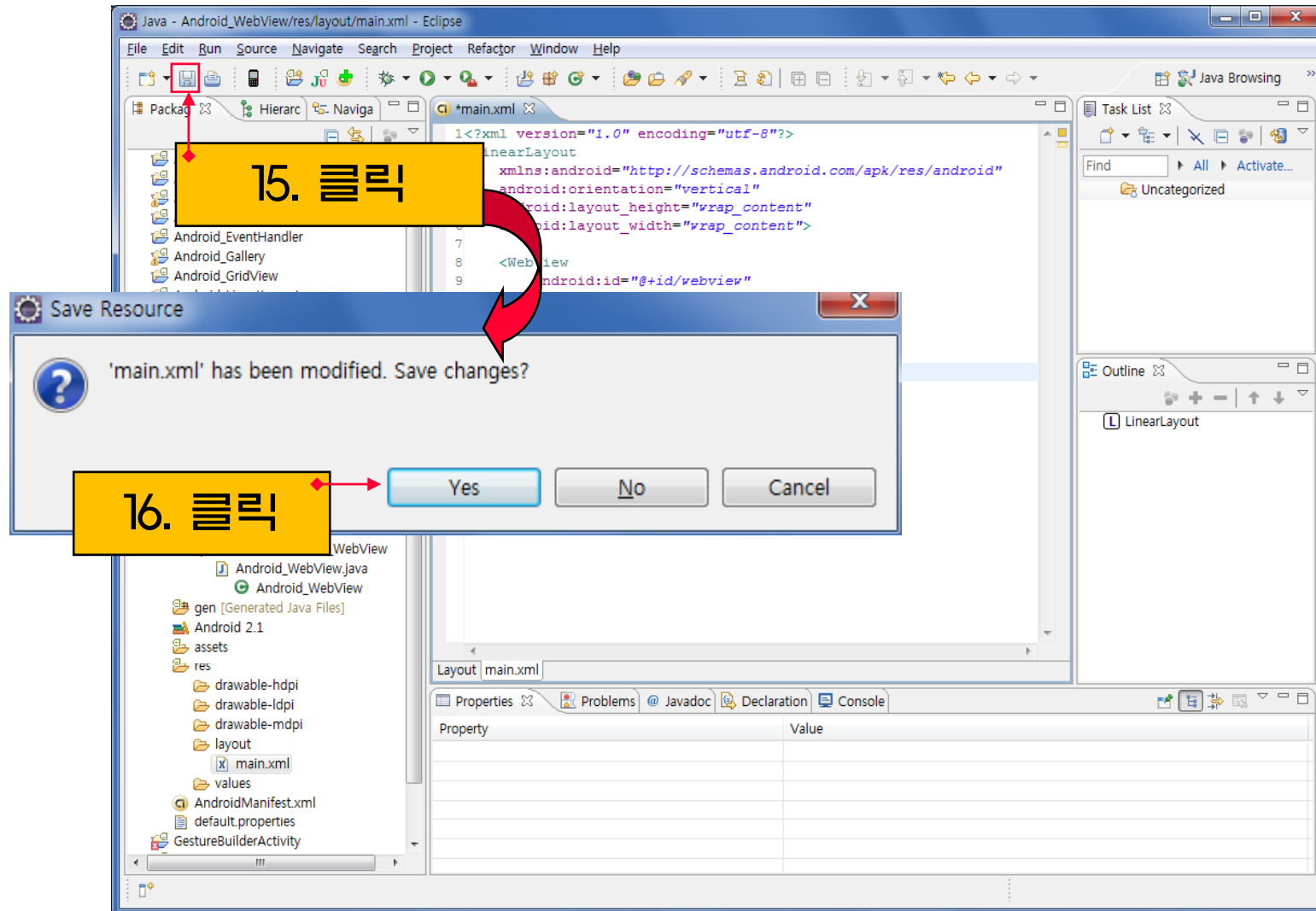
14. 클릭





# Web View 구현 (9)

## ❖ Main.xml 저장





# Web View 구현 (10)

## ■ Android\_WebView.java

```
*Android_WebView.java X
1 package com.inhatc.android_WebView;
2
3 import android.app.Activity;
4 import android.os.Bundle;
5 import android.webkit.WebView;
6
7 public class Android_WebView extends Activity {
8     private WebView objWebView;    //WebView 객체 선언
9
10
11     /** Called when the activity is first created. */
12     @Override
13     public void onCreate(Bundle savedInstanceState) {
14         super.onCreate(savedInstanceState);
15         setContentView(R.layout.main);
16
17         objWebView = (WebView) findViewById(R.id.webview);
18         objWebView.getSettings().setJavaScriptEnabled(true);
19         objWebView.loadUrl("http://www.google.com");
20     }
21 }
```

17. Coding



# Web View 구현 (11)

## ❖ AndroidManifest.xml 수정

```
*Android_WebView Manifest
1<?xml version="1.0" encoding="utf-8"?>
2<manifest xmlns:android="http://schemas.android.com/apk/res/android"
3    package="com.inhatec.android_WebView"
4    android:versionCode="1"
5    android:versionName="1.0">
6    <uses-permission android:name="android.permission.INTERNET" />
7    <application
8        android:icon="@drawable/icon"
9        android:label="@string/app_name">
10        <activity
11            android:name=".Android_WebView"
12            android:label="@string/app_name">
13            <intent-filter>
14                <action
15                    android:name="android.intent.action.MAIN" />
16                <category
17                    android:name="android.intent.category.LAUNCHER" />
18            </intent-filter>
19        </activity>
20    </application>
21    <uses-sdk android:minSdkVersion="7" />
22</manifest>
```

18. XML code 수정 확인



# Web View 구현 (12)

❖ R.java

```
*R.java X
1+/* AUTO-GENERATED FILE. DO NOT MODIFY.
7
8 package com.inhatc.android_WebView;
9
10 public final class R {
11     public static final class attr {
12     }
13     public static final class drawable {
14         public static final int icon=0x7f020000;
15     }
16     public static final class id {
17         public static final int webview=0x7f050000;
18     }
19     public static final class layout {
20         public static final int main=0x7f030000;
21     }
22     public static final class string {
23         public static final int app_name=0x7f040001;
24         public static final int hello=0x7f040000;
25     }
26 }
```

19. id 클래스 변수 추가 확인





# Web View 구현 (13)

## ❖ Android 프로젝트 실행

The screenshot shows the Eclipse IDE interface. The 'Run' menu is open, and the 'Run As' option is selected. A yellow box with the text '20. 클릭' (Click) points to the 'Run' menu item. The 'Run As' dialog box is open, showing the 'Select a way to run 'Test':' list. The 'Android Application' option is selected, and a yellow box with the text '21. 클릭' (Click) points to it. The 'Description' section shows 'Runs an Android Application'. A yellow box with the text '22. 클릭' (Click) points to the 'OK' button. The background shows the AndroidManifest.xml file being edited.

20. 클릭

21. 클릭

22. 클릭





# Web View 구현 (14)

## 실행 결과





# 실습 1 : Web View 구현 (1)

## ■ Android\_WebView (실습 시간 : 30분)

- ❖ Android\_WebView.java 파일을 프로그래밍하여 아래 그림과 같이 forward / backward 기능을 구현하시오.





# Web View 구현 (15)

## ❖ Linear Layout 속성 변경

컨트롤	속성 지정
LinearLayout	<ul style="list-style-type: none"><li>• xmlns:android=http://schemas.android.com/apk/res/android</li><li>• android:orientation="vertical"</li><li>• android:layout_height="fill_parent"</li><li>• android:layout_width="fill_parent"&gt;</li></ul>

25. 속성 변경





# Web View 구현 (16)

## ❖ Android\_WebViewClient 클래스 추가

```
*Android_WebView.java X
1 package com.inhatec.android_WebView;
2
3 import android.app.Activity;
4 import android.os.Bundle;
5 import android.view.KeyEvent;
6 import android.webkit.WebView;
7 import android.webkit.WebViewClient;
8
9 public class Android_WebView extends Activity {
10     private WebView objWebView;    //WebView 객체 선언
11
12     private class Android_WebViewClient extends WebViewClient {
13         @Override
14         public boolean shouldOverrideUrlLoading(WebView view, String url){
15             view.loadUrl(url);
16             return true;
17         }
18         @Override
19         public boolean shouldOverrideKeyEvent (WebView view, KeyEvent event){
20             int keyCode = event.getKeyCode();
21             if ((keyCode == KeyEvent.KEYCODE_DPAD_LEFT) && objWebView.canGoBack()){
22                 objWebView.goBack();
23                 return true;
24             }
25             else if ((keyCode == KeyEvent.KEYCODE_DPAD_RIGHT) && objWebView.canGoForward()){
26                 objWebView.goForward();
27                 return true;
28             }
29             return false;
30         }
31     }
```

26. Code 추가





# Web View 구현 (17)

## ❖ Android\_WebView 클래스 Code 수정

```
Android_WebView.java
32  /** Called when the activity is first created. */
33  @Override
34  public void onCreate(Bundle savedInstanceState) {
35      super.onCreate(savedInstanceState);
36      setContentView(R.layout.main);
37
38      objWebView = (WebView) findViewById(R.id.webview);
39      objWebView.setWebViewClient(new Android_WebViewClient()); //code 수정
40      objWebView.getSettings().setJavaScriptEnabled(true);
41      objWebView.loadUrl("http://www.google.com");
42  }
43  }
44
```

27. Code 추가





# Web View 구현 (18)

## ❖ Android 프로젝트 실행

The screenshot shows the Eclipse IDE interface. The 'Run' menu is open, and the 'Run As' option is selected. A yellow box with the text '28. 클릭' (Click) points to the 'Run' menu item. The 'Run As' dialog box is open, showing the 'Select a way to run 'Test':' list. The 'Android Application' option is selected. A yellow box with the text '29. 클릭' (Click) points to the 'Android Application' option. The 'Description' section shows 'Runs an Android Application'. A yellow box with the text '30. 클릭' (Click) points to the 'OK' button. The background shows the AndroidManifest.xml file being edited.

28. 클릭

29. 클릭

30. 클릭



# Web View 구현 (19)

## 실행 결과



31. 클릭



32. motoroi 입력

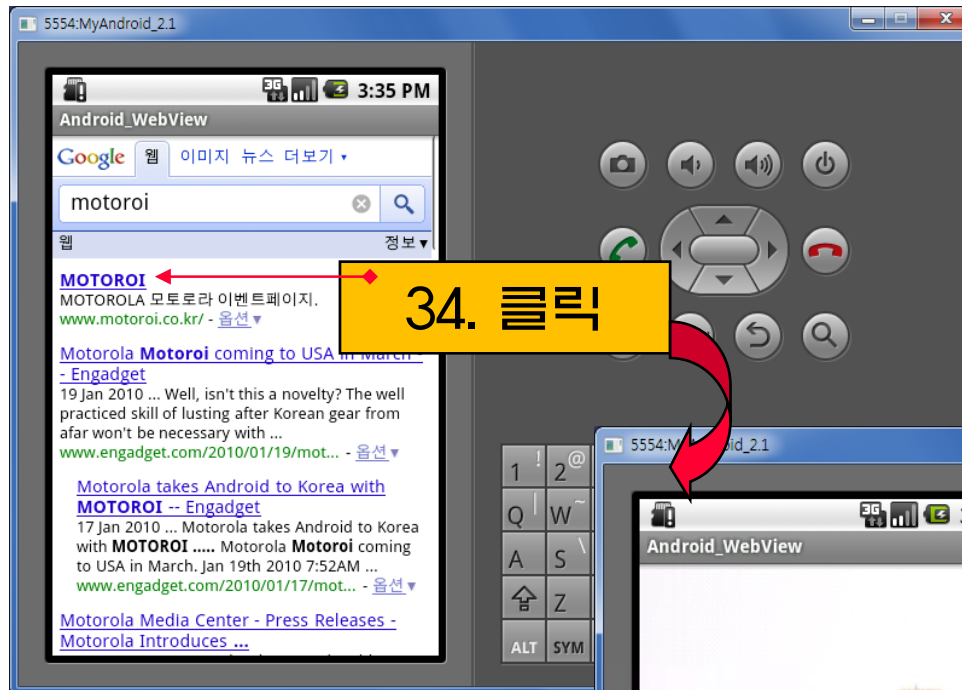
33. 클릭







# Web View 구현 (20)



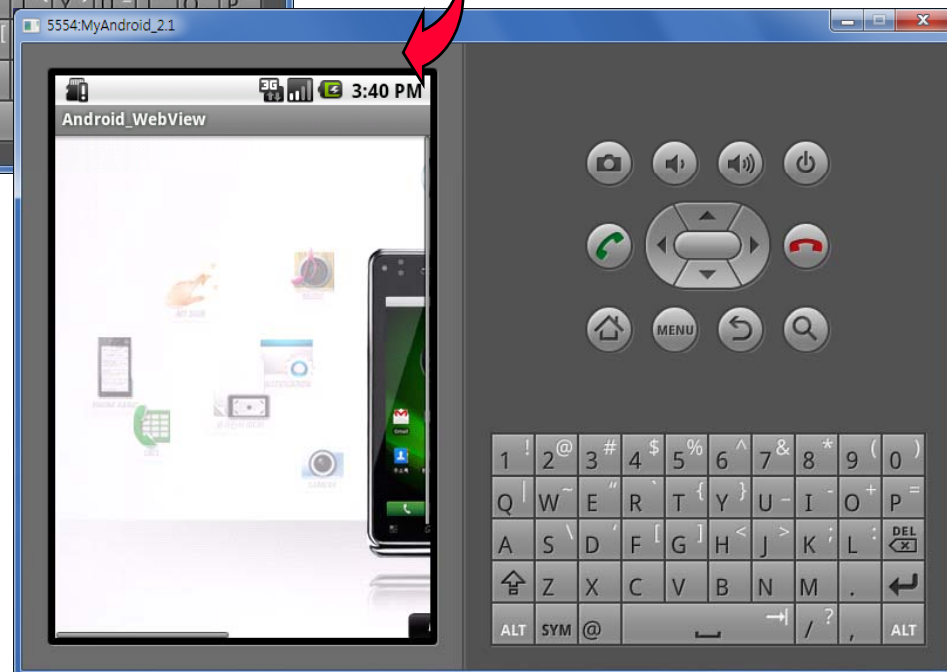




# Web View 구현 (21)



36. 클릭





# 학습 요약

## ■ Web View 구현



open handset alliance