

CS 646 Android Mobile Application Development
Spring Semester, 2015
Doc 7 Source Control, Selection
Feb 10, 2014

Copyright ©, All rights reserved. 2015 SDSU & Roger Whitney, 5500 Campanile Drive, San Diego, CA 92182-7700 USA. OpenContent (<http://www.opencontent.org/openpub/>) license defines the copyright on this document.

From Last Class

Fragment Life Cycle - Order Of Methods

When first created

Activity onCreate
Fragment Constructor
Fragment onAttach
Fragment onCreate
Fragment onCreateView
Activity onStart
Fragment onStart
Activity onResume

When recreated

Fragment Constructor
Activity onCreate
Fragment onAttach
Fragment onCreate
Fragment onCreateView
Activity onStart
Fragment onStart
Activity onResume

onBackPressed ()

Activity method

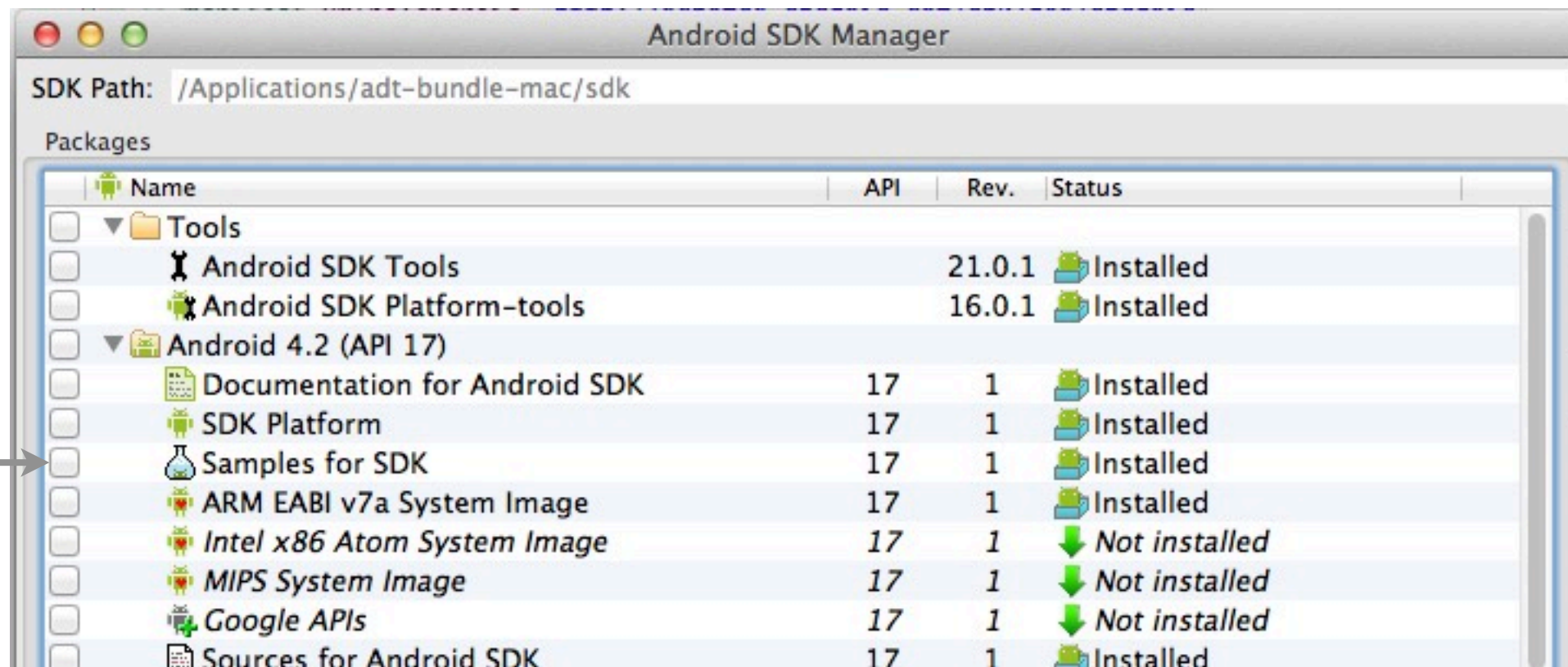
Called when user presses on back button

Call super if you want to finish current activity

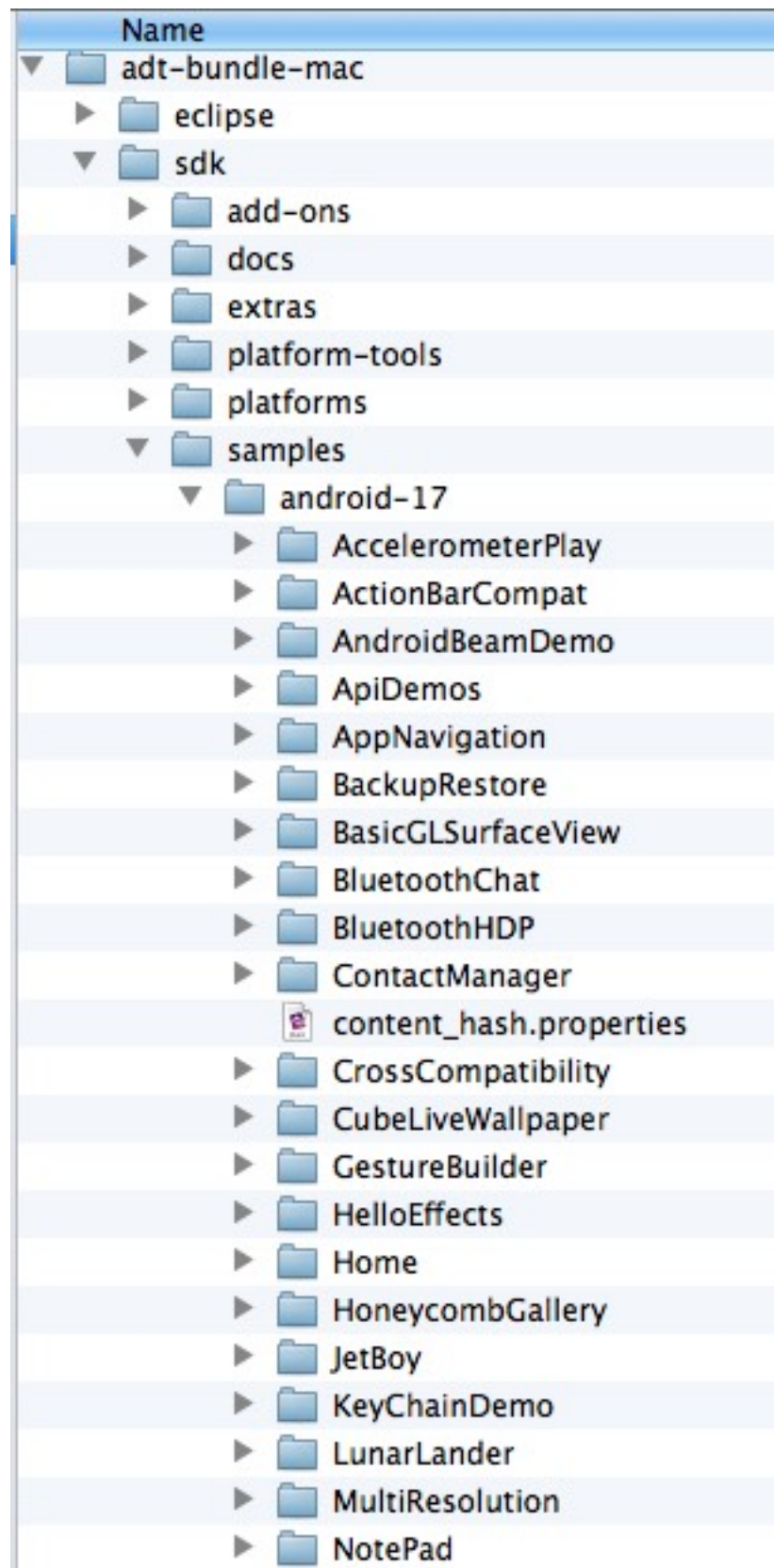
Examples

Google Examples

In SDK Manager



Samples are in sdk/samples



Source Control

Source Control

Track changes in software

Maintain software in one location

Multiple people can make updates

Common Free Source Control Systems

CVS

Concurrent Versions System

Command line interface in Unix

Various interfaces in Window

Subversion

Claims to be a better CVS

Many commands are same as CVS

Git

Created by Linus Torvald
Distributed Version control

Mercurial

Python based
Distributed version control

Mercurial

<http://mercurial.selenic.com/>

Runs on:

Mac OS X

Unix

Linux

Windows

Command line interface

GUI interface for windows

Eclipse plugins

Can use locally with no server

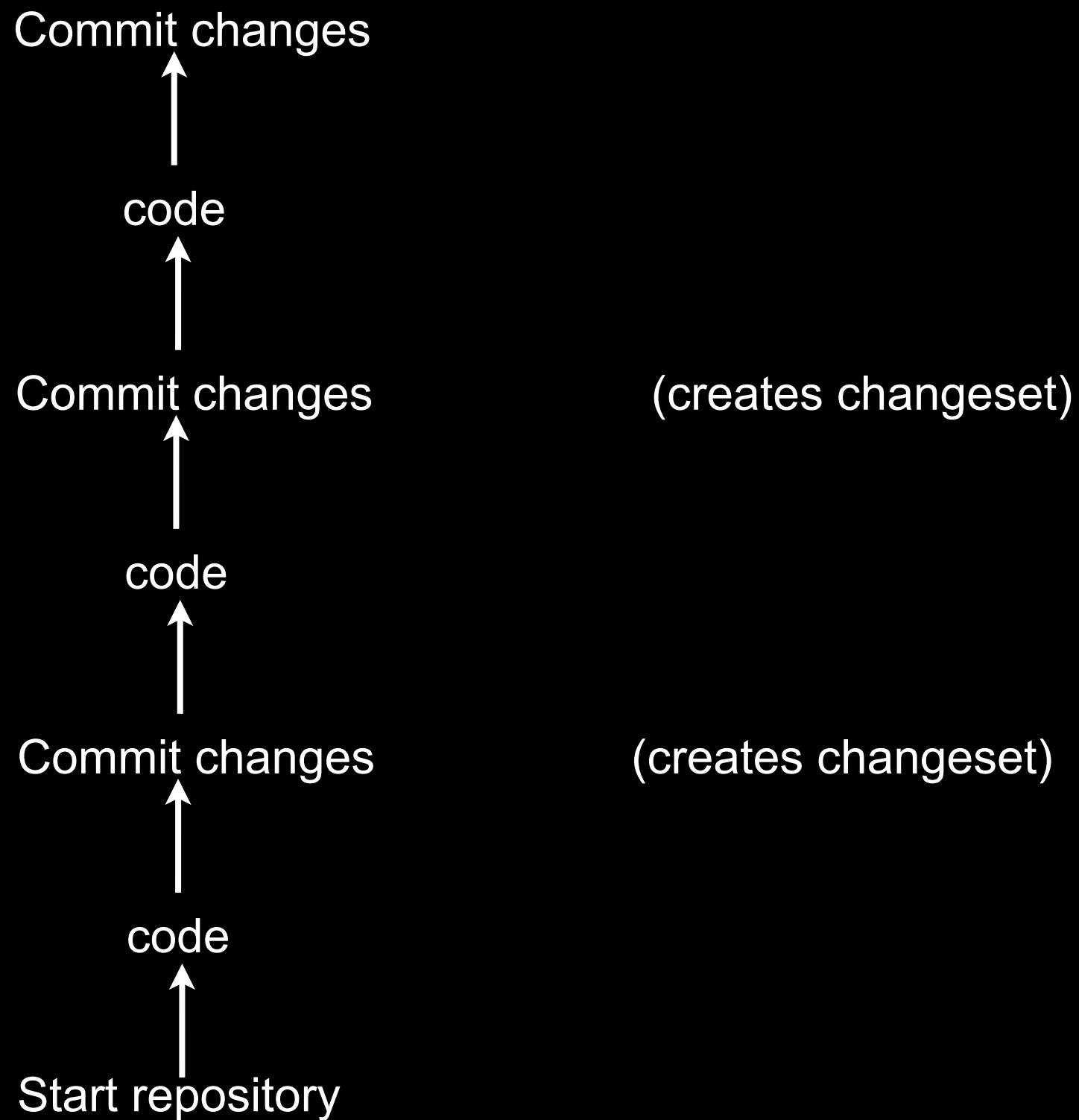
Mercurial in Eclipse

https://bitbucket.org/mercurialeclipse/main/wiki/Installation_and_Configuration#!installation-and-configuration

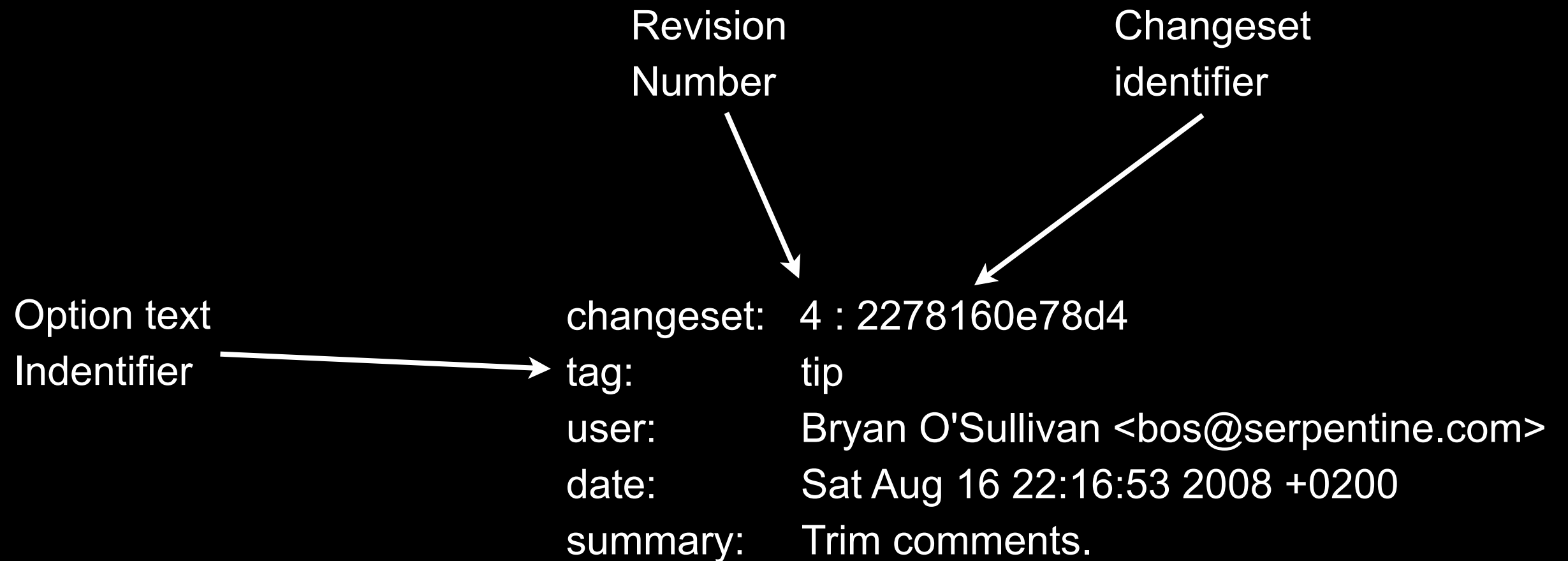
requires Mercurial installed

<http://mercurial.selenic.com/>

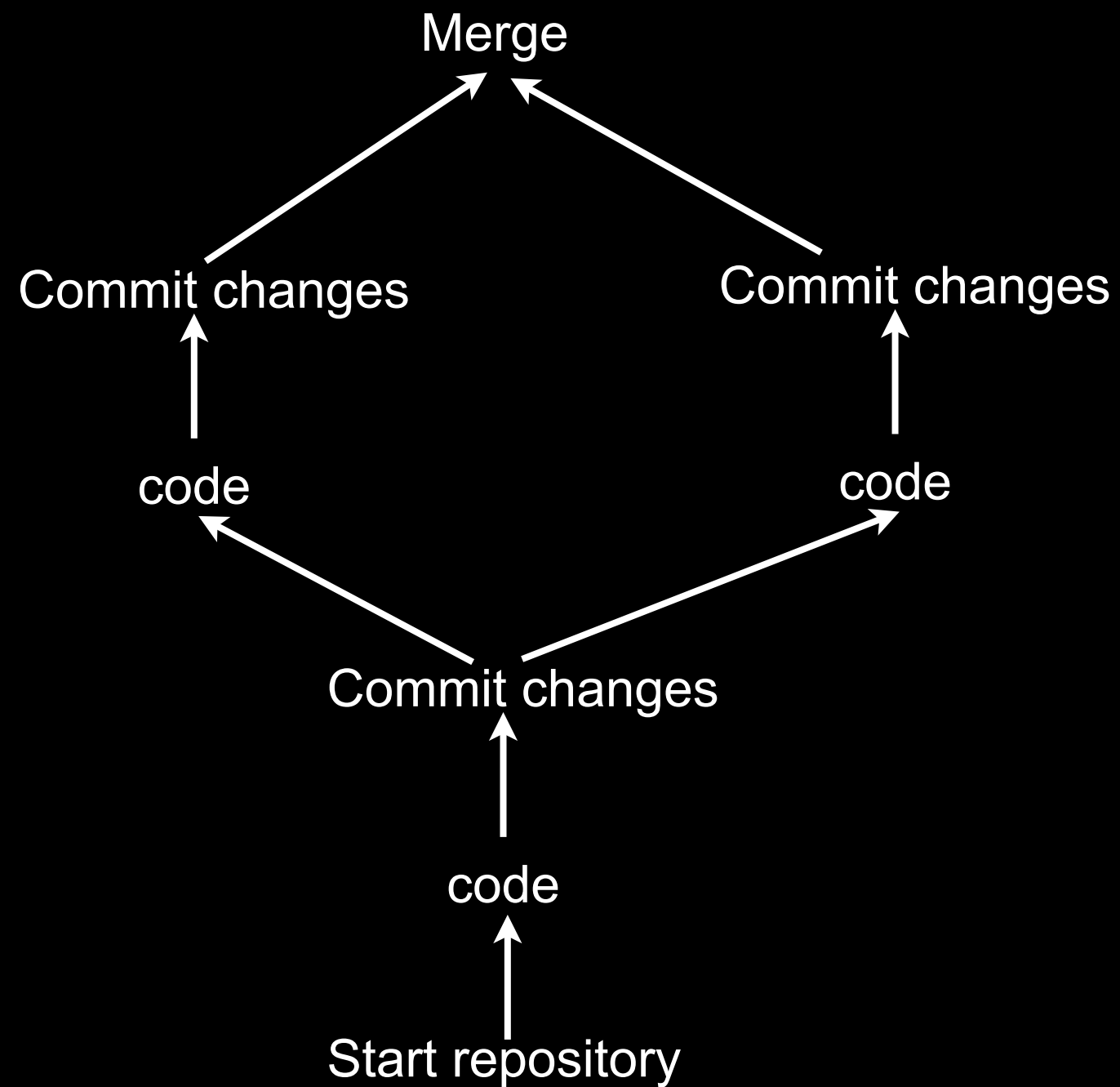
Simple Workflow



Changesets



Branch in one Repository



Basic Source Control Operations

Starting a new project

Adding code to a project

Modifying existing code

Retrieving past versions of code

Handling conflicts in code

Creating code branches

Merging code branches

Cloning repositories

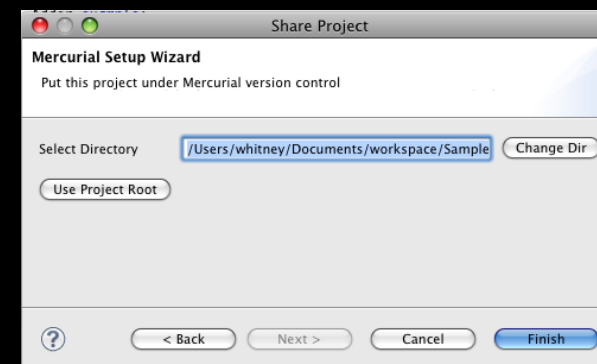
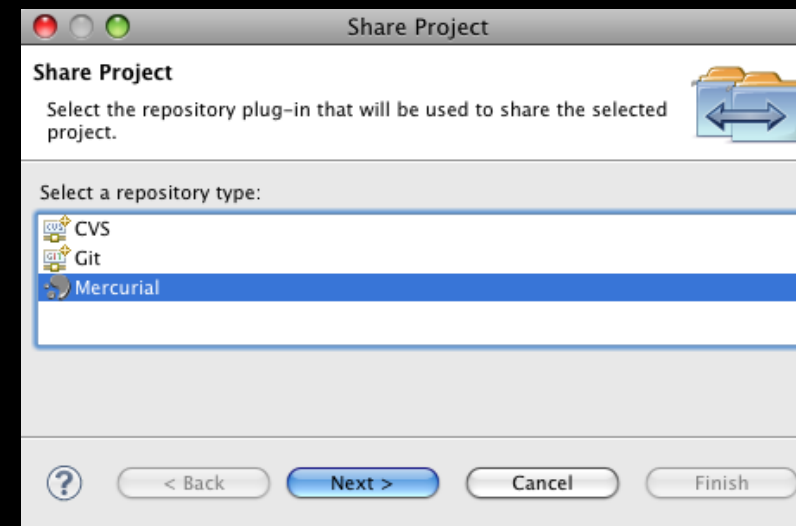
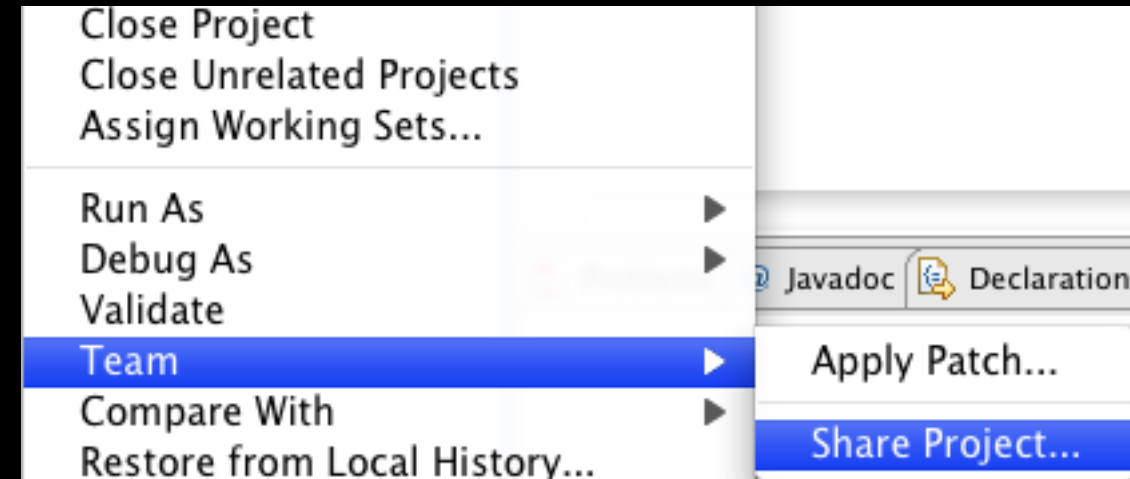
Pulling repositories

Pushing repositories

Mercurial Commands

Creating a Repository

hg init myproject



Repository & Working Directory

Repository

In .hg directory

Contains

- history of changes

- list of files part of project

Working Directory

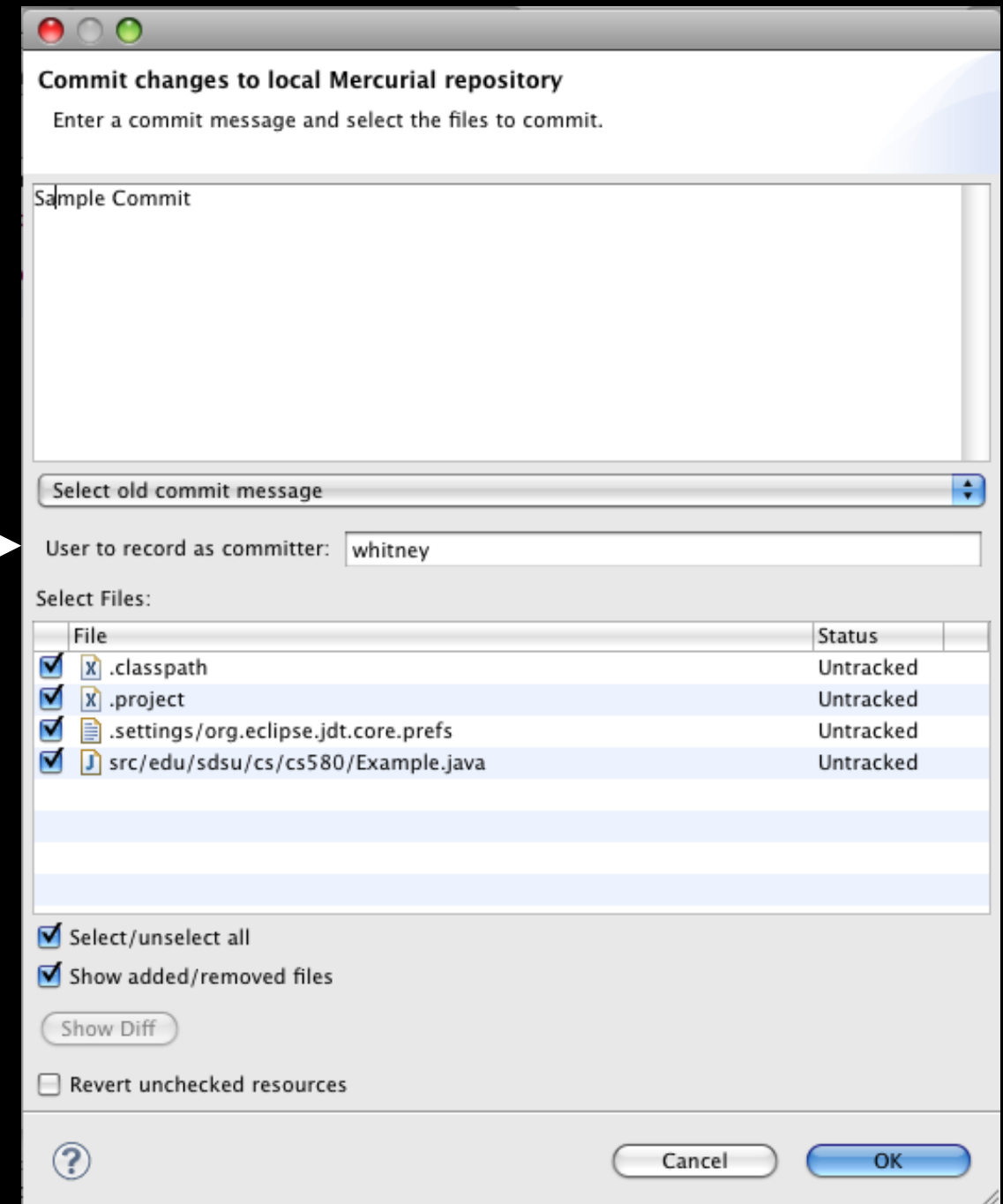
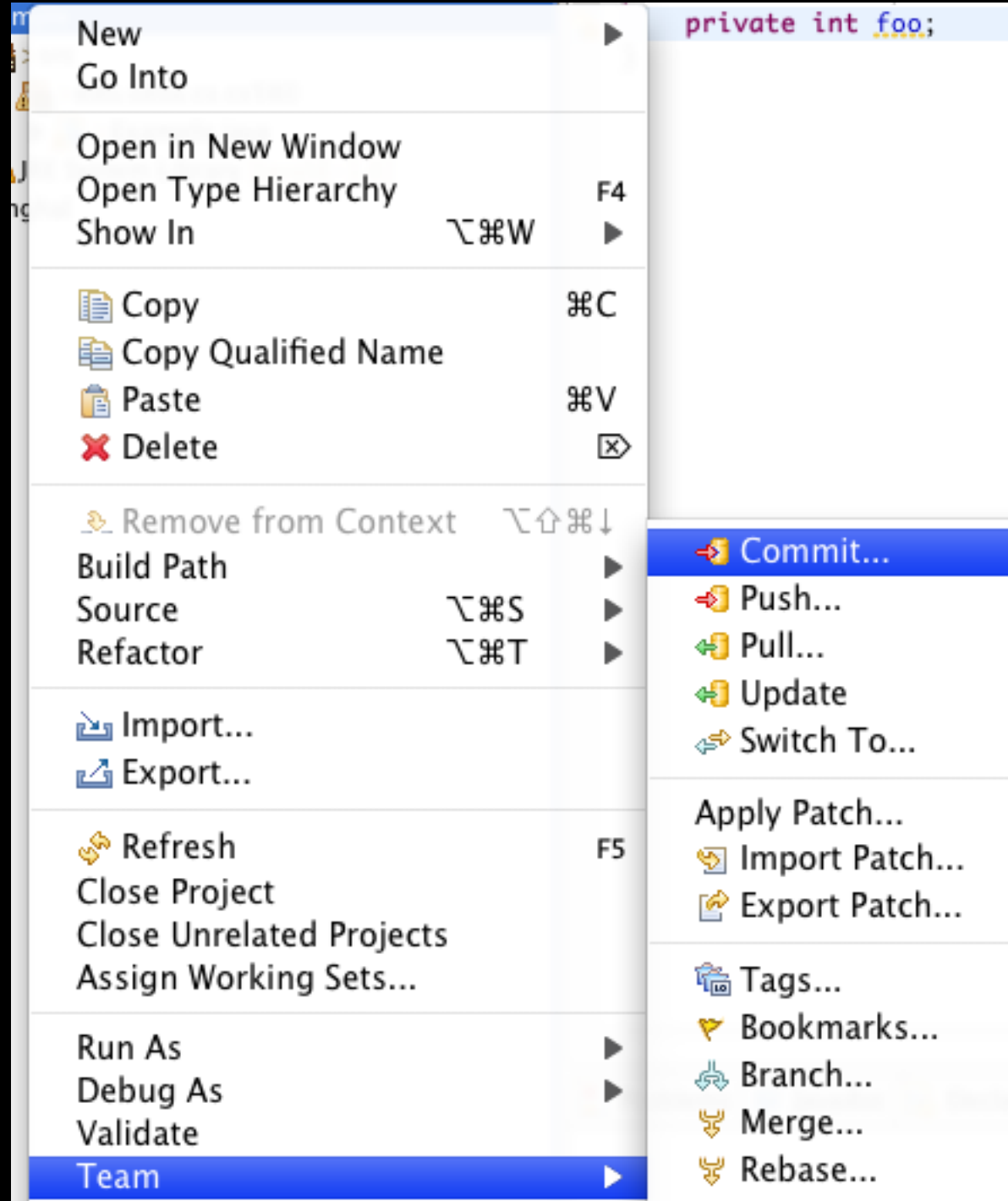
Contains

- Project files

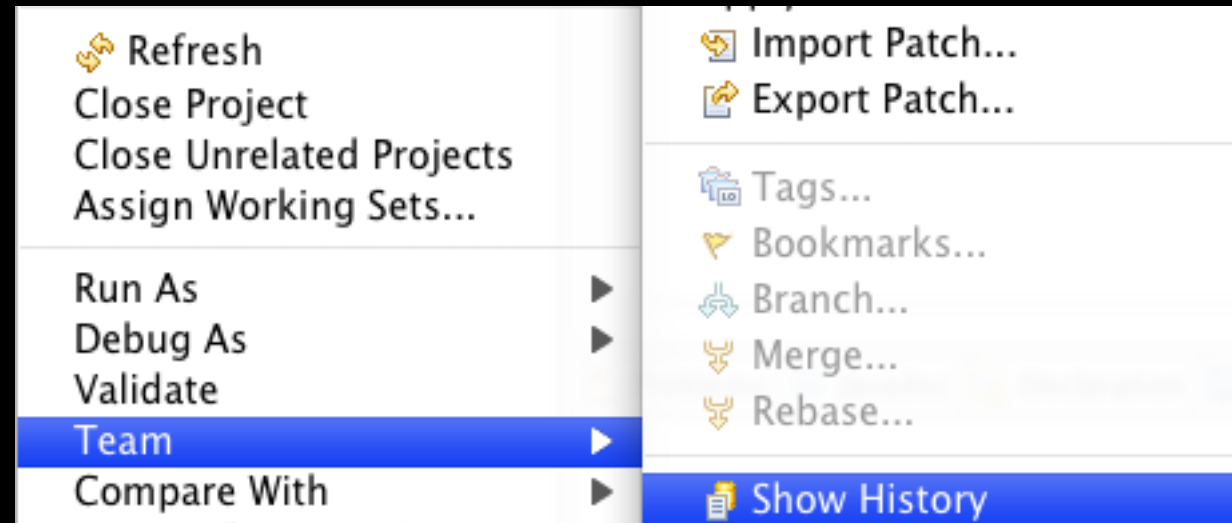
- Project directories

- .hg

Adding/Committing in Eclipse

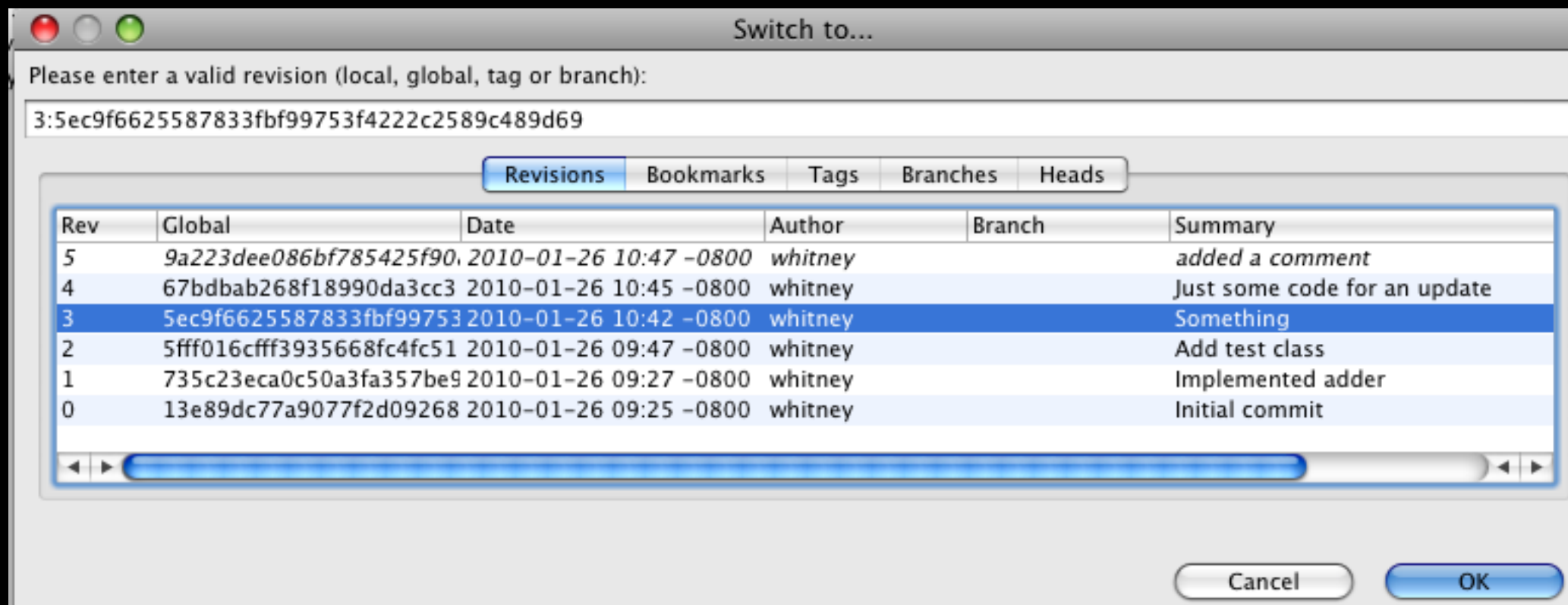
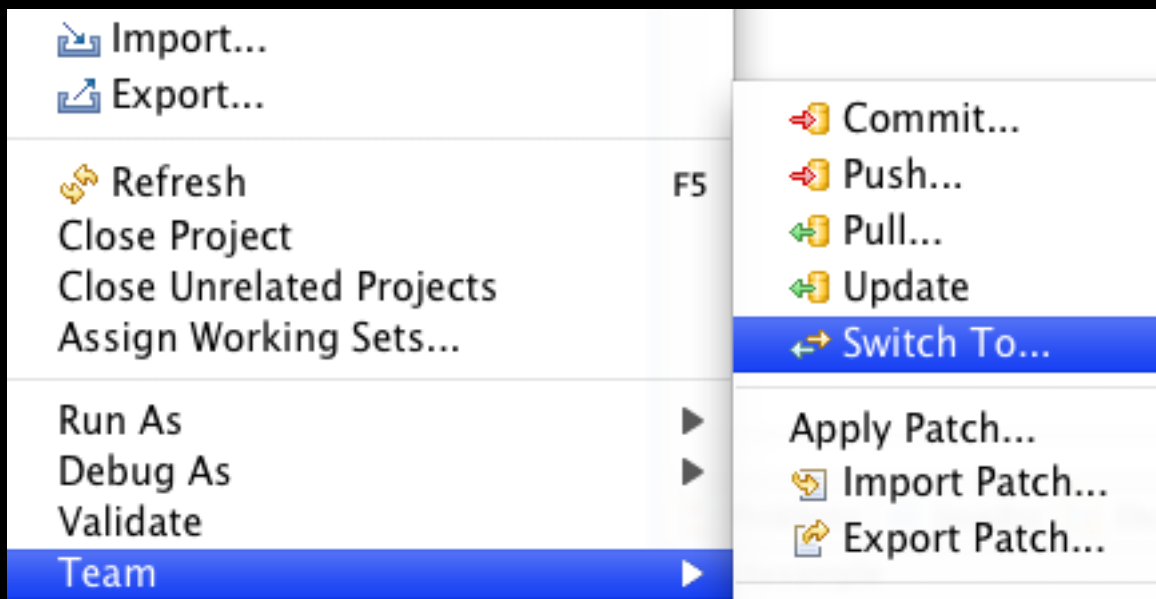


hg log



/junitexample						
Graph	Changeset	Tag	Branch	User	Date	Summary
●	5:9a223dee086b	tip		whitney	2010-01-26 10:47 -08	added a comment
●	4:67bdbab268f1			whitney	2010-01-26 10:45 -08	Just some code for an update
●	3:5ec9f6625587			whitney	2010-01-26 10:42 -08	Something
●	2:5fff016cfff3			whitney	2010-01-26 09:47 -08	Add test class
●	1:735c23eca0c5			whitney	2010-01-26 09:27 -08	Implemented adder
●	0:13e89dc77a90			whitney	2010-01-26 09:25 -08	Initial commit

Reverting to different Version

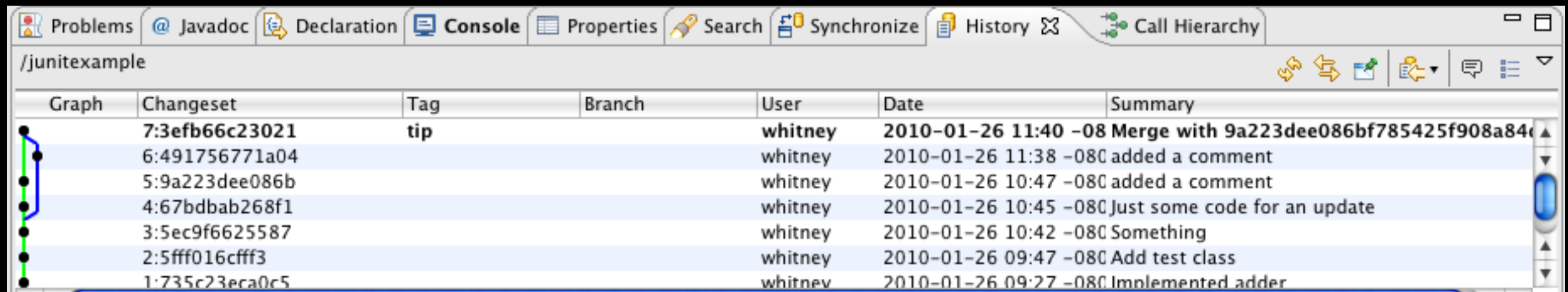


Creating Branches

/junitexample						
Graph	Commit	Tag	Branch	User	Date	Summary
	6:751735771a04	tip		whitney	2010-01-26 11:38 -08	added a comment
	5:9a223dee086b			whitney	2010-01-26 10:47 -08	added a comment
	4:67bdbab268f1			whitney	2010-01-26 10:45 -08	Just some code for an update
	3:5ec9f6625587			whitney	2010-01-26 10:42 -08	Something
	2:5fff016cfff3			whitney	2010-01-26 09:47 -08	Add test class
	1:735c23eca0c5			whitney	2010-01-26 09:27 -08	Implemented adder
	0:13e89dc77a90			whitney	2010-01-26 09:25 -08	Initial commit

Merging - without conflicts

use the merge item in the team menu
and commit



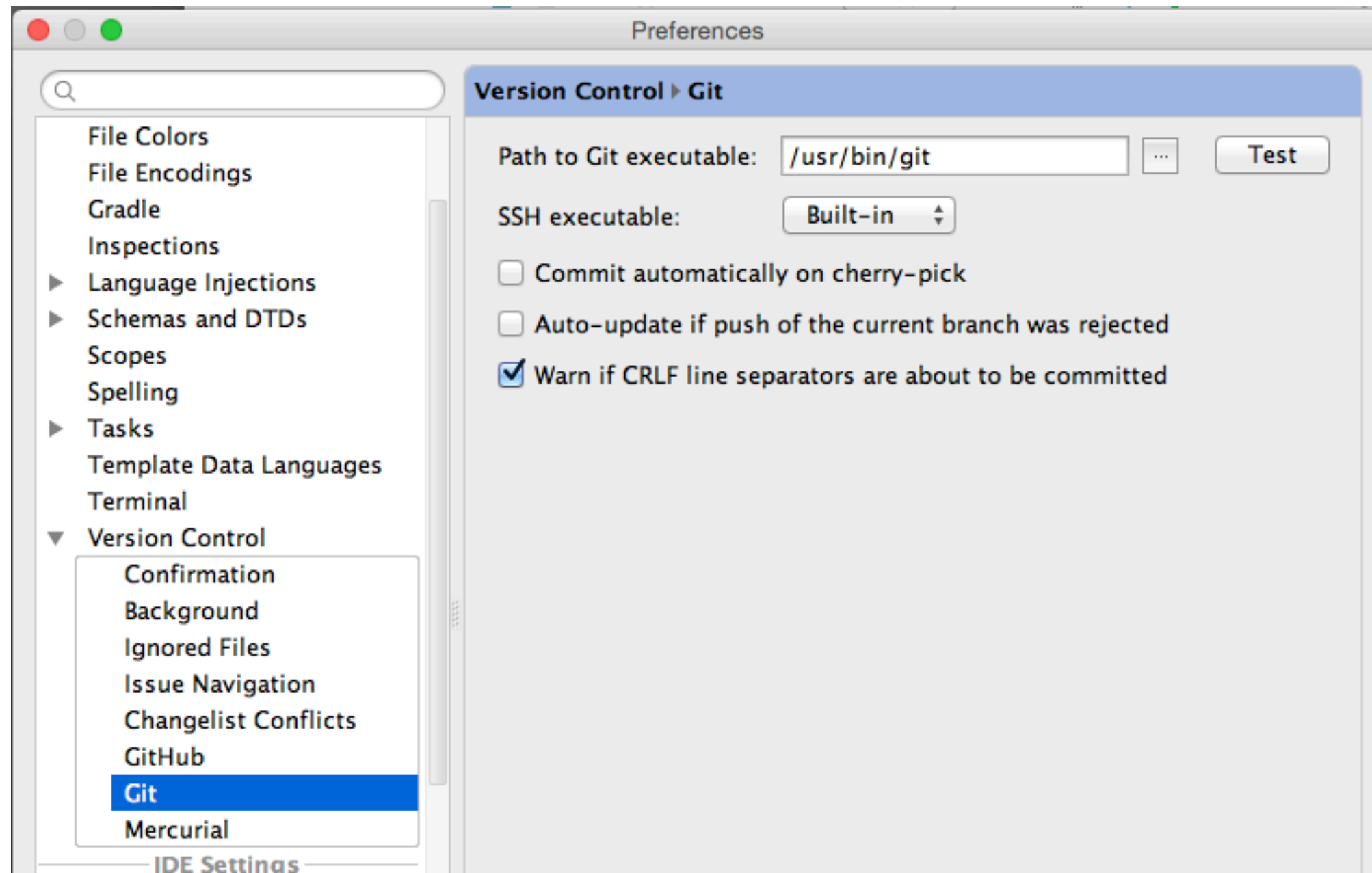
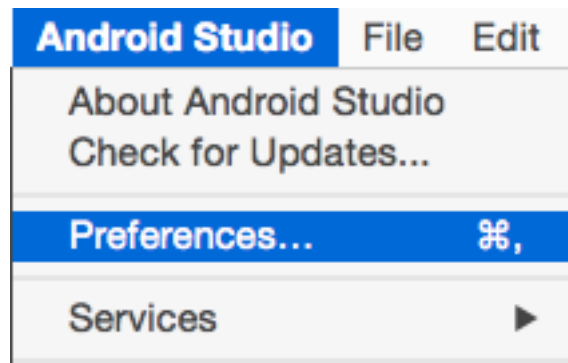
Git in Android Studio

<http://git-scm.com/>

Downloads

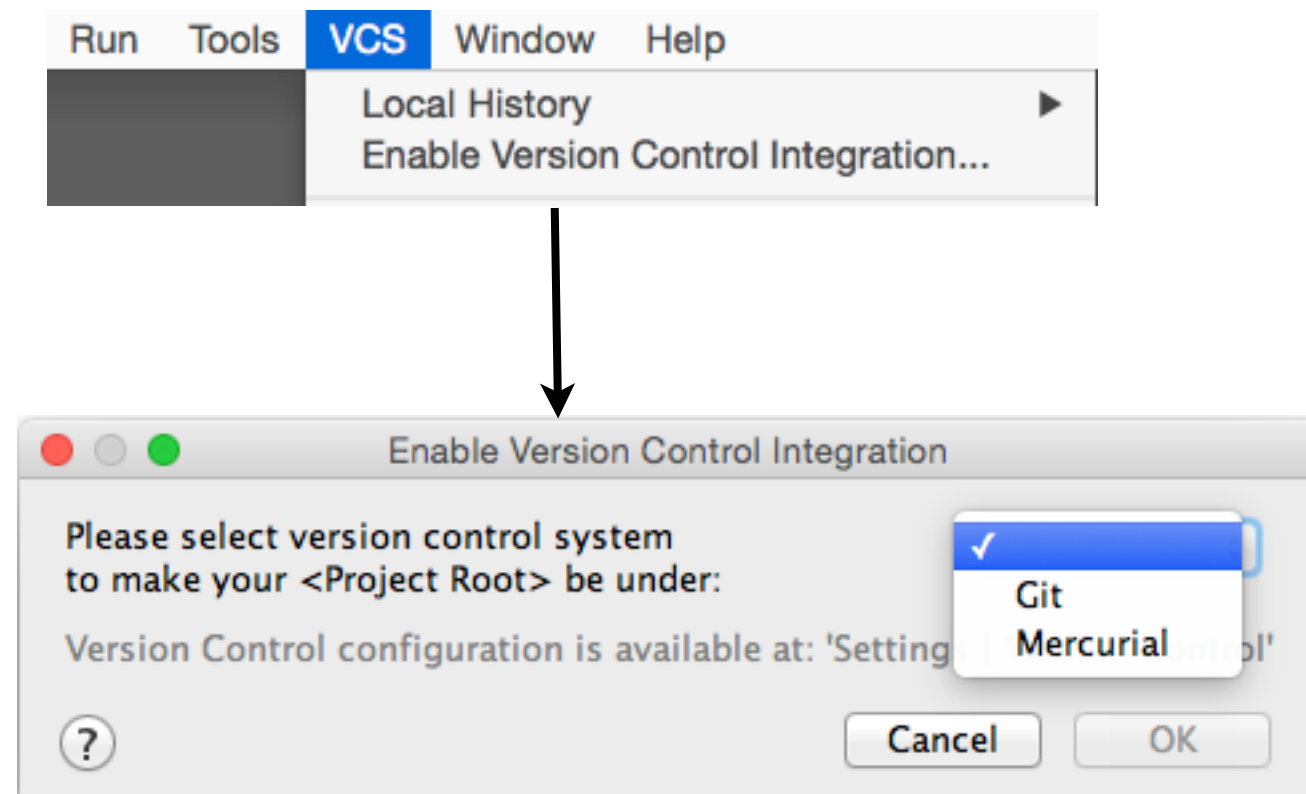
Documentation

Letting Android Studio know Git's Location



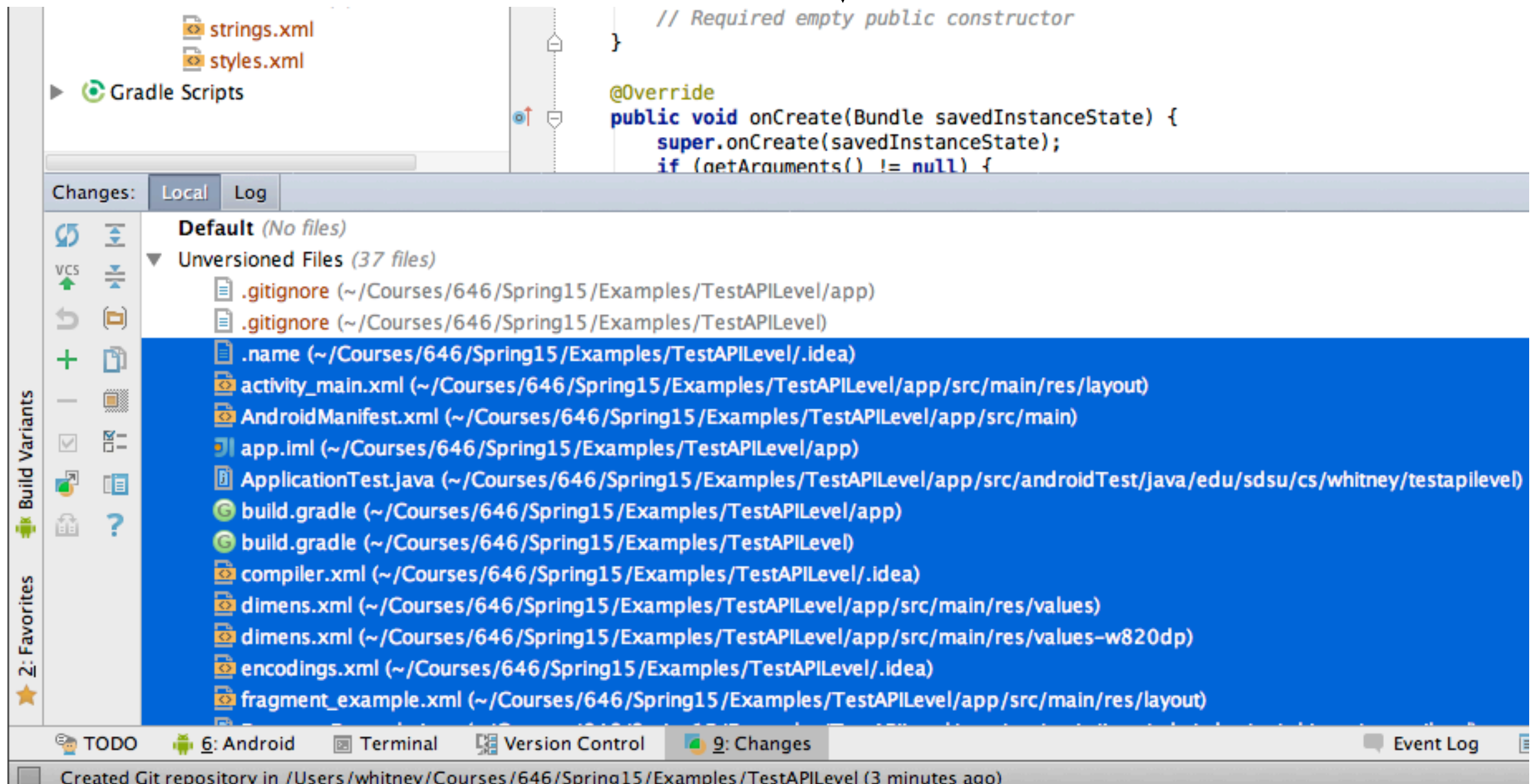
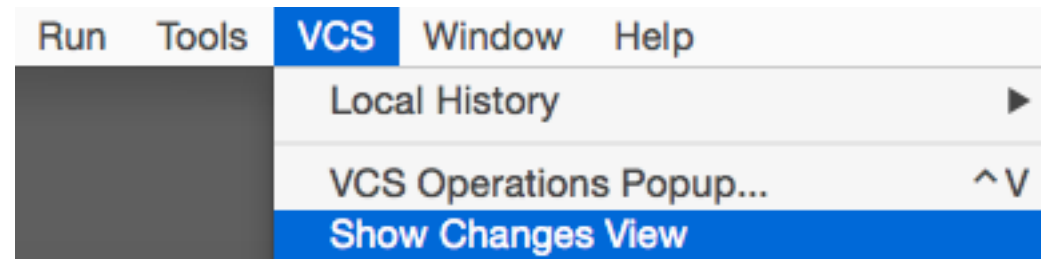
Installing git in a Project

This needs to be done only once in a project

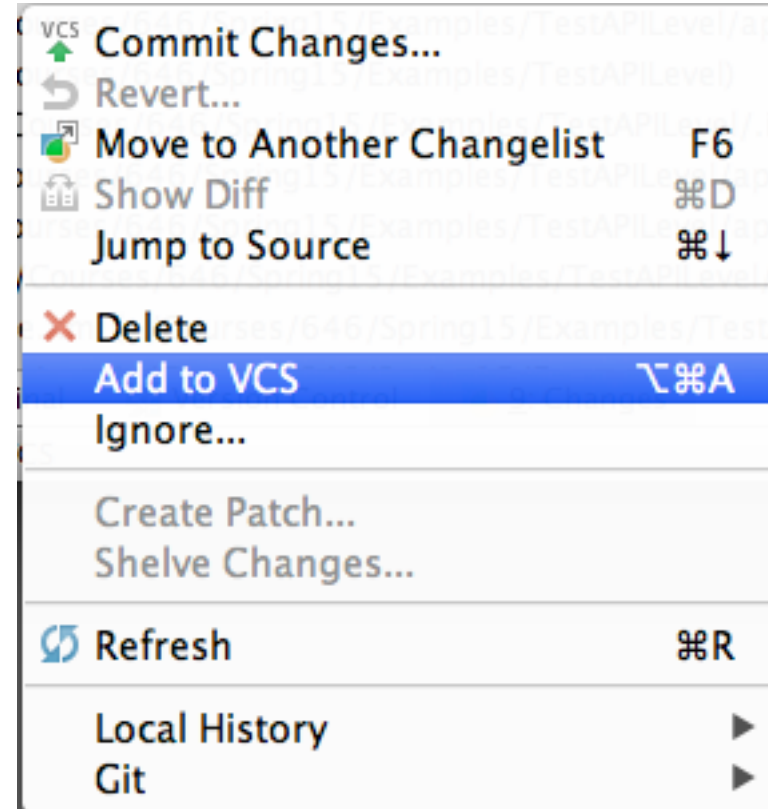


Adding Files

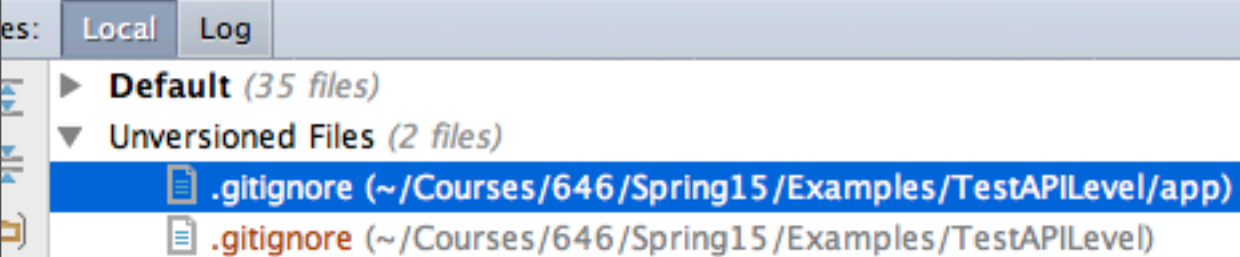
You need to add files to git, first select them



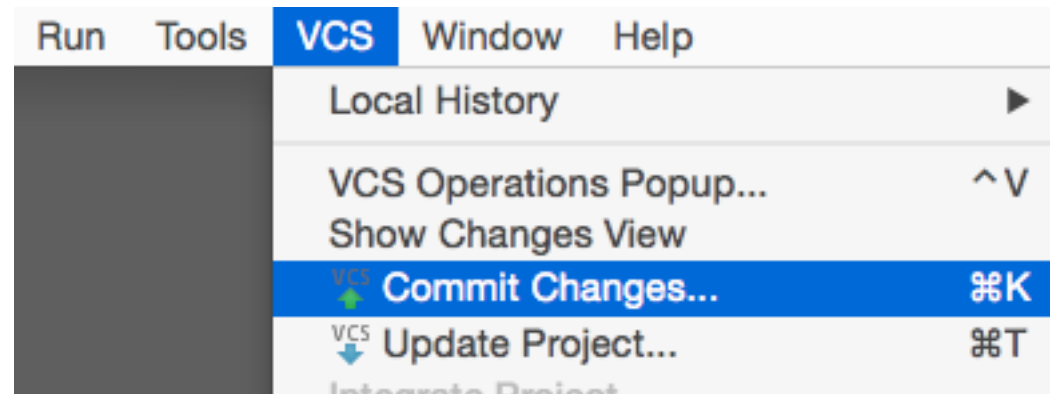
Now Add them



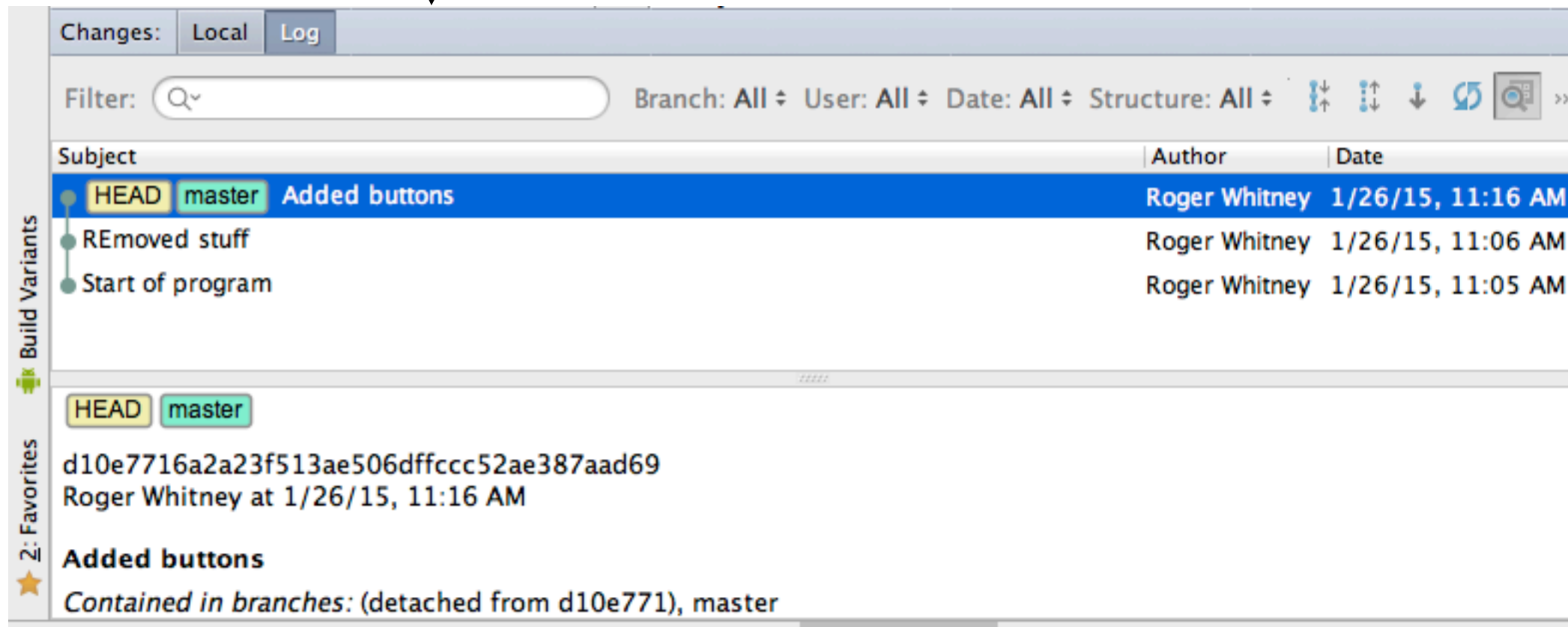
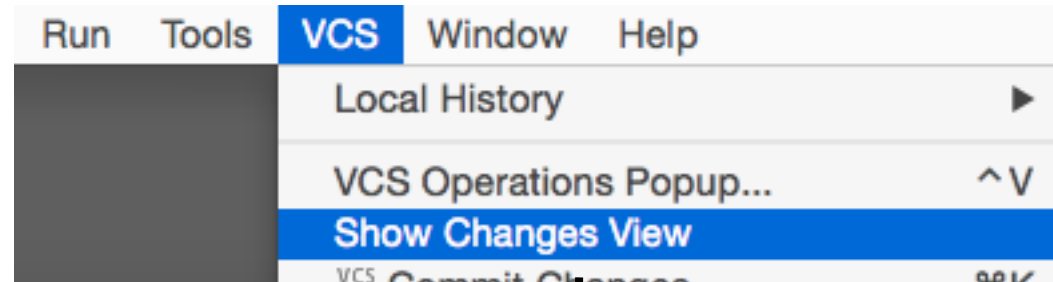
```
super.onCreate(savedInstanceState);  
if (getArguments() != null) {
```



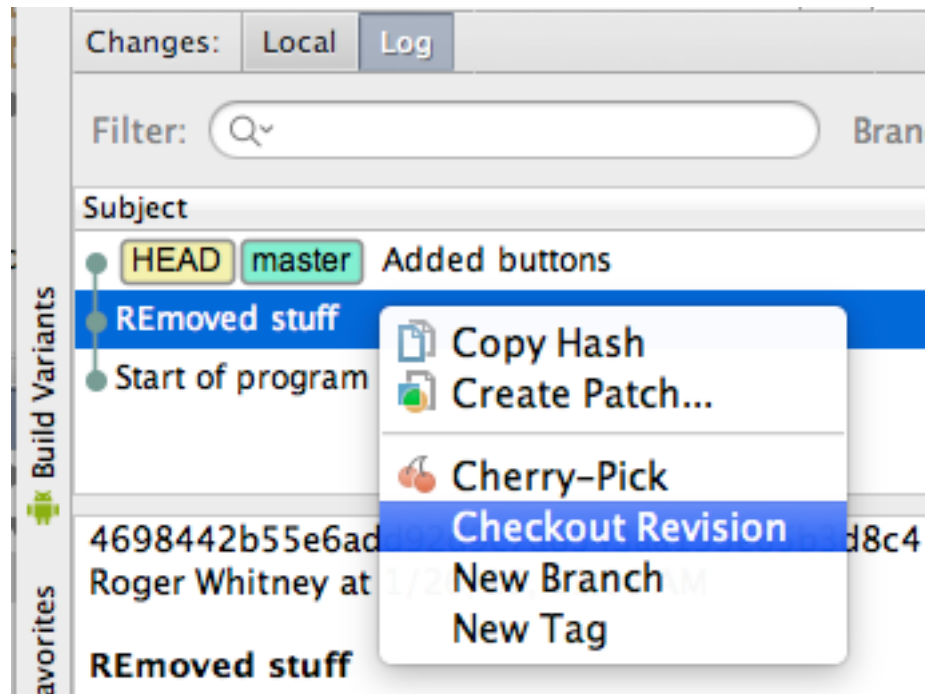
Finally Commit



Viewing Commits



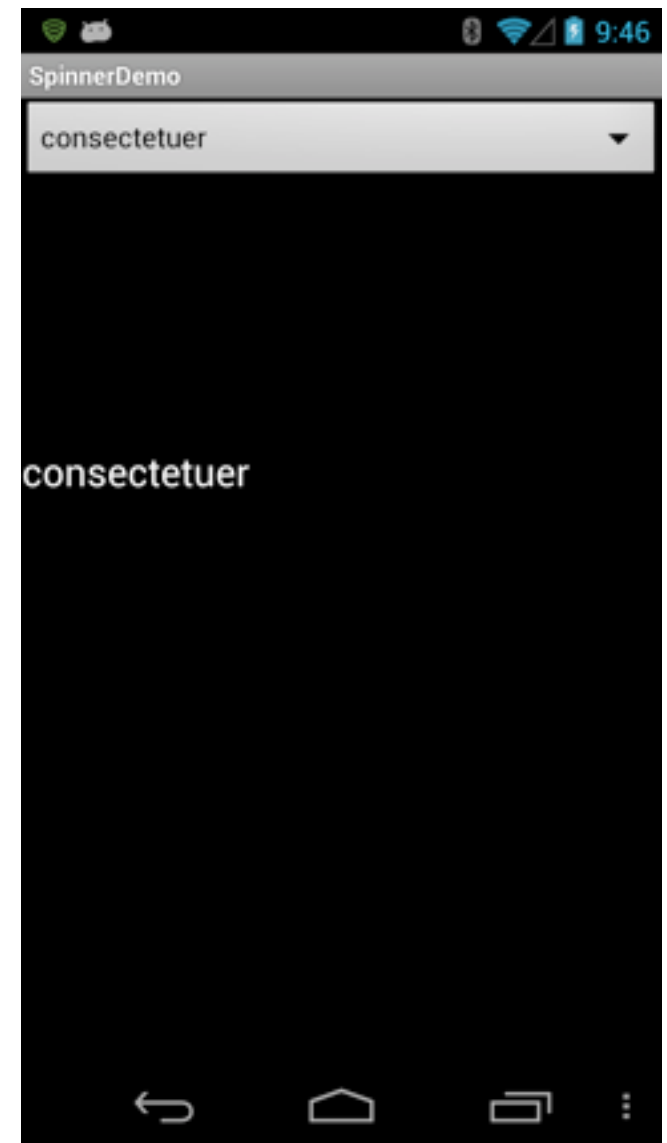
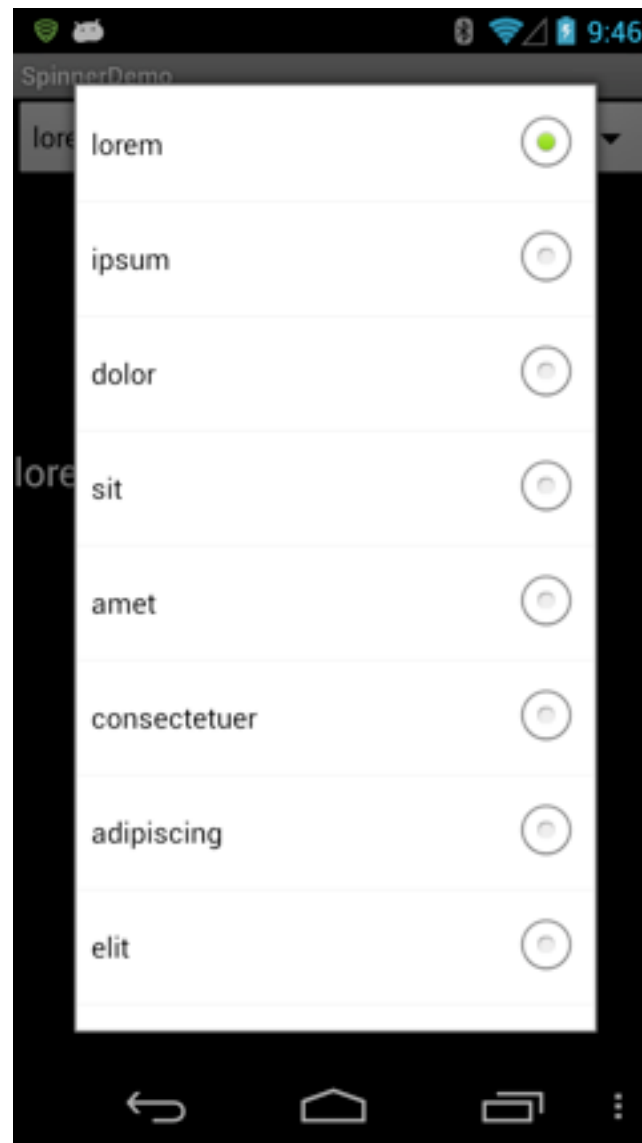
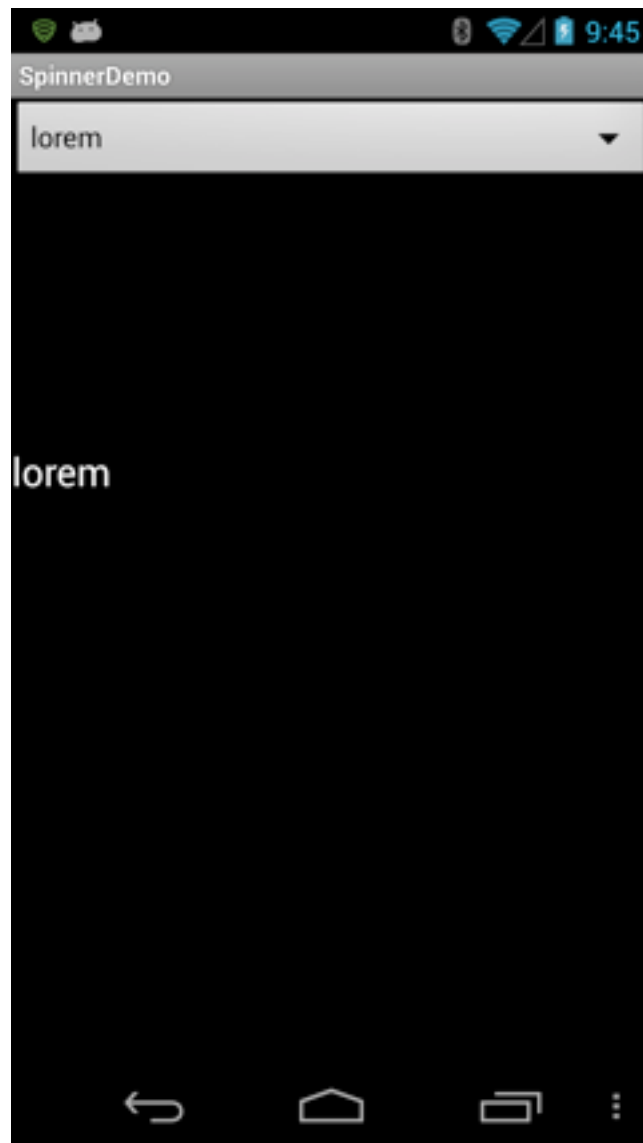
Checking out older Versions



(right click)

Selection

Spinner



Data Sources

Some UI elements (spinner, lists, gridView) need a data source

ArrayAdaptor

You provide data in an array

Adaptor provides it in proper format for UI element

Adapter Pattern



ArrayAdapter

```
setListAdapter(new ArrayAdapter<String>(this,  
    android.R.layout.simple_list_item_1, items));
```

We have an array of Strings

ListView uses a list of Views

ArrayAdapter

Adapts (converts) what we have to what ListView needs

Creates a view for each element in the array

Other Adapters

CursorAdapter

Converts database cursor for display in selection view

SimpleAdapter

Converts XML resources

ArrayAdapter

Can

- Add

- Remove

- Filter

- Sort

```
public ArrayAdapter (Context context, int textViewResourceId, T[] objects)
```

```
public ArrayAdapter (Context context, int resource, int textViewResourceId, T[] objects)
```

Supply view to populate

AdapterView.OnItemSelectedListener

Need OnItemSelectedListener

When user select item in Spinner following methods are called

```
public void onItemSelected(AdapterView<?> parent, View v, int position, long id)  
public void onNothingSelected(AdapterView<?> parent)
```


Layout

```
<TextView
    android:id="@+id/selection"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_alignParentTop="true"
    android:layout_marginTop="195dp"
    android:textAppearance="?android:attr/textAppearanceLarge" />
```

```
<Spinner
    android:id="@+id/spinner"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_alignParentTop="true"
    android:drawSelectorOnTop="true" />
```

```
</RelativeLayout>
```

SpinnerDemo Class

```
public class SpinnerDemo extends Activity
    implements AdapterView.OnItemClickListener {
    private TextView selection;
    private static final String[] items={"lorem", "ipsum", "dolor",
                                         "sit", "amet","consectetuer", "adipiscing", "elit", "morbi", "vel"};

    public void onCreate(Bundle icle) {
        super.onCreate(icle);
        setContentView(R.layout.main);
        selection=(TextView)findViewById(R.id.selection);

        Spinner spin=(Spinner)findViewById(R.id.spinner);
        spin.setOnItemClickListener(this);

        ArrayAdapter<String> adapter=new ArrayAdapter<String>(this,
                                                                android.R.layout.simple_spinner_item, items);
        adapter.setDropDownViewResource(
            android.R.layout.simple_spinner_dropdown_item);
        spin.setAdapter(adapter);
    }
```

SpinnerDemo Class

```
public void onItemSelected(AdapterView<?> parent, View v, int position, long id) {  
    selection.setText(items[position]);  
}
```

```
public void onNothingSelected(AdapterView<?> parent) {  
    selection.setText("");  
}
```

ArrayAdapter

```
public ArrayAdapter (Context context, int textViewResourceId, T[] objects)
```

context

The current context

textViewResourceId

The resource ID for a layout file containing a TextView to use when instantiating views

objects

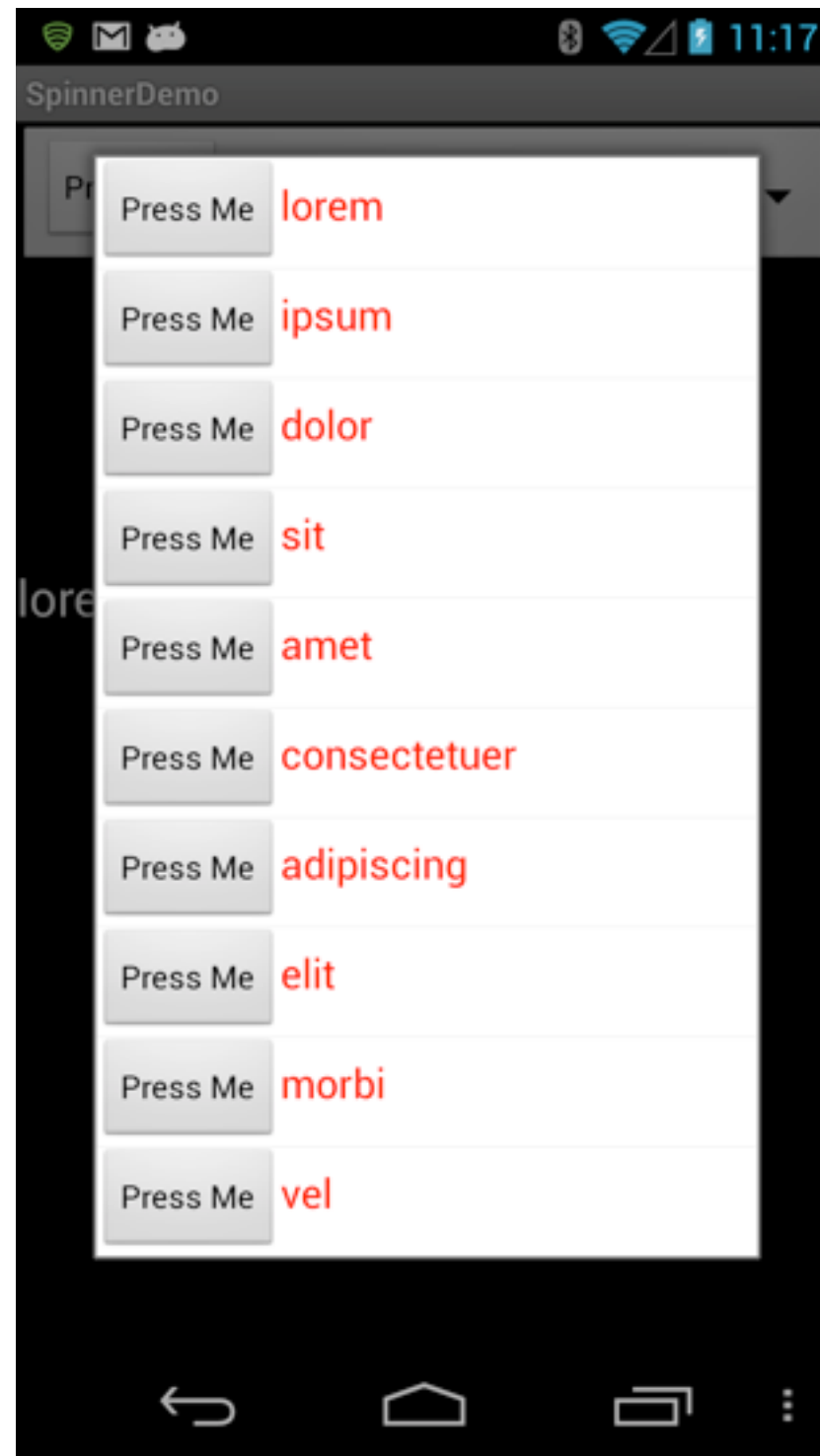
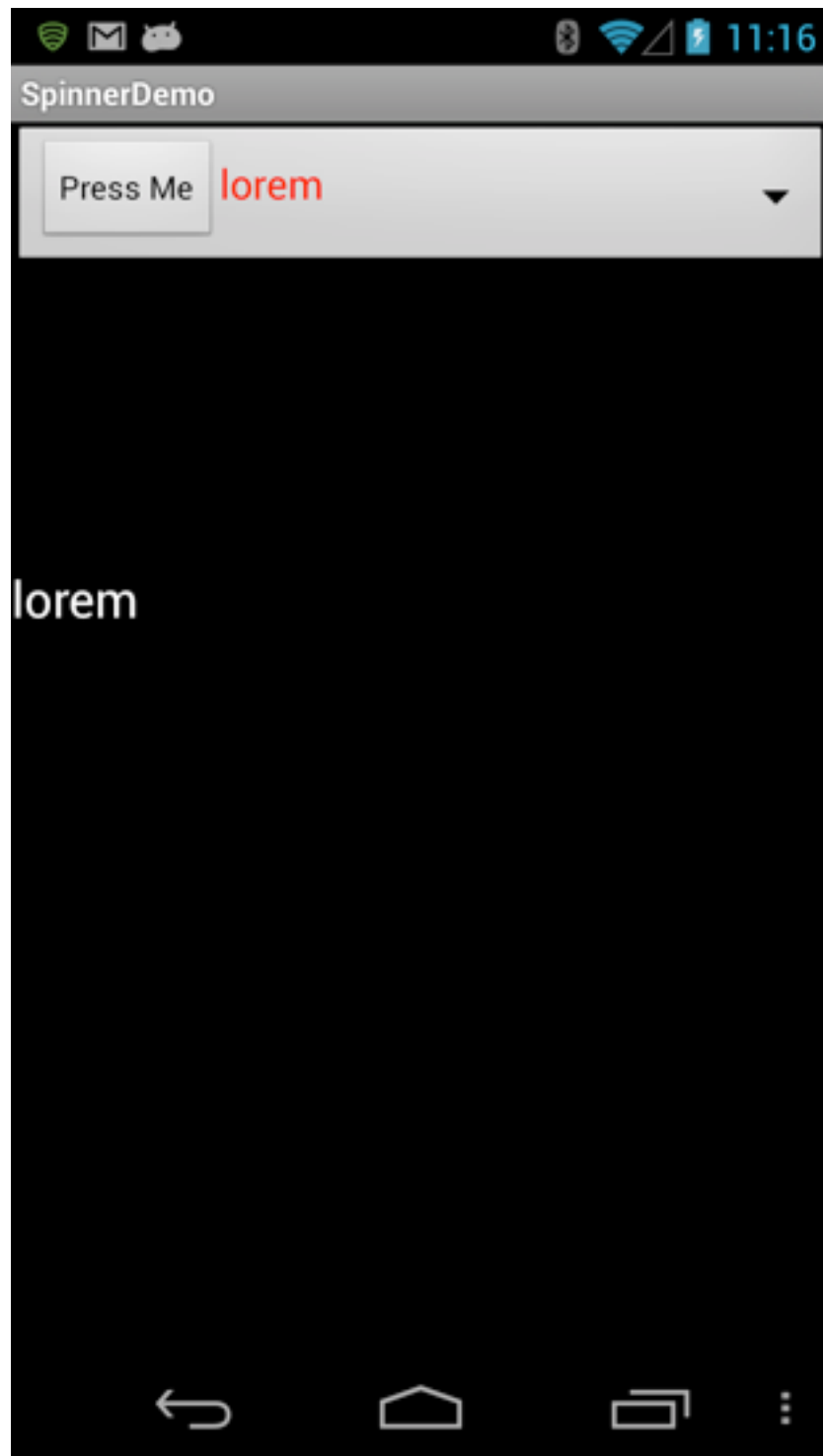
The objects to represent in the ListView.

android.R.layout.simple_spinner_item

Android has predefined layouts for UI lists

Can provide your own layout

Custom Row Layouts



res/layout/spinner_row.xml

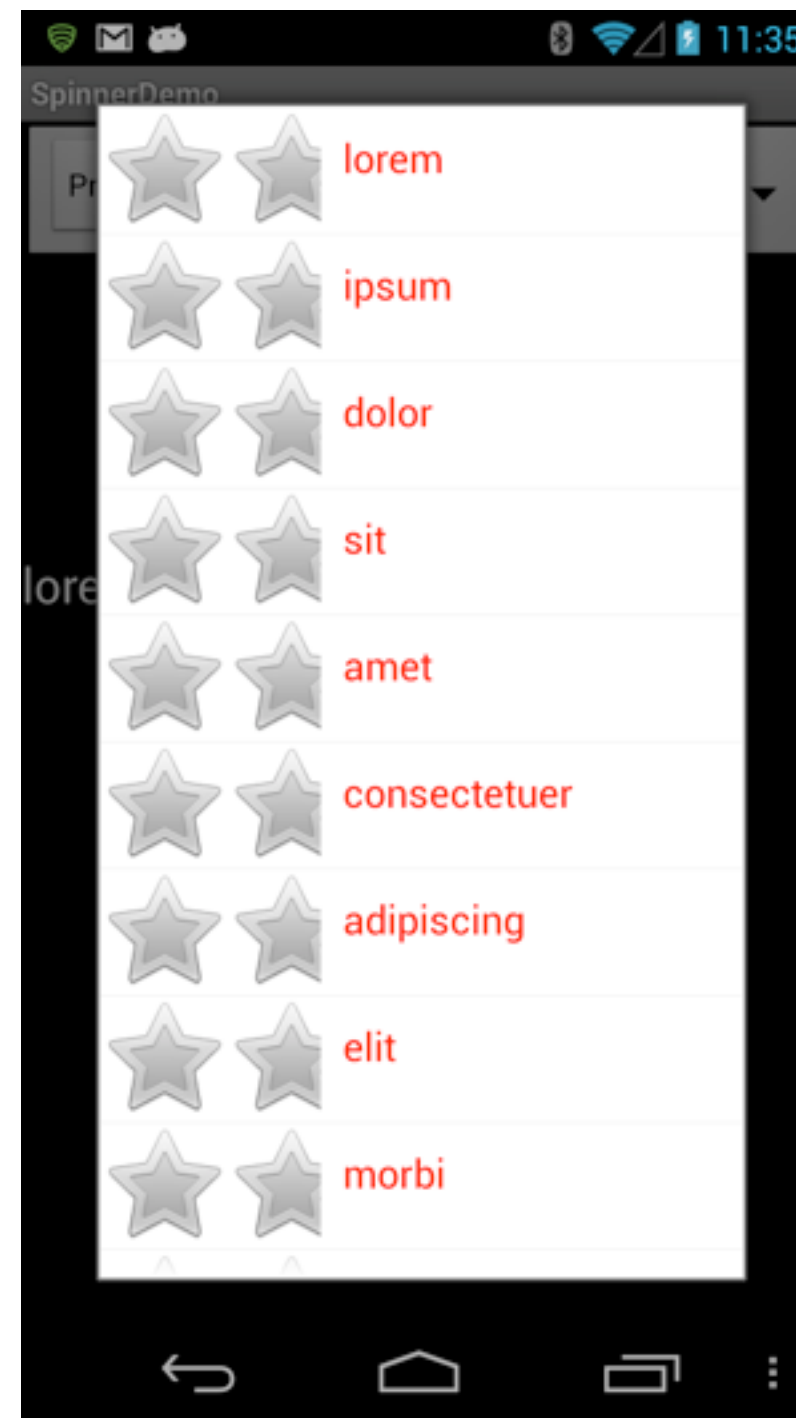
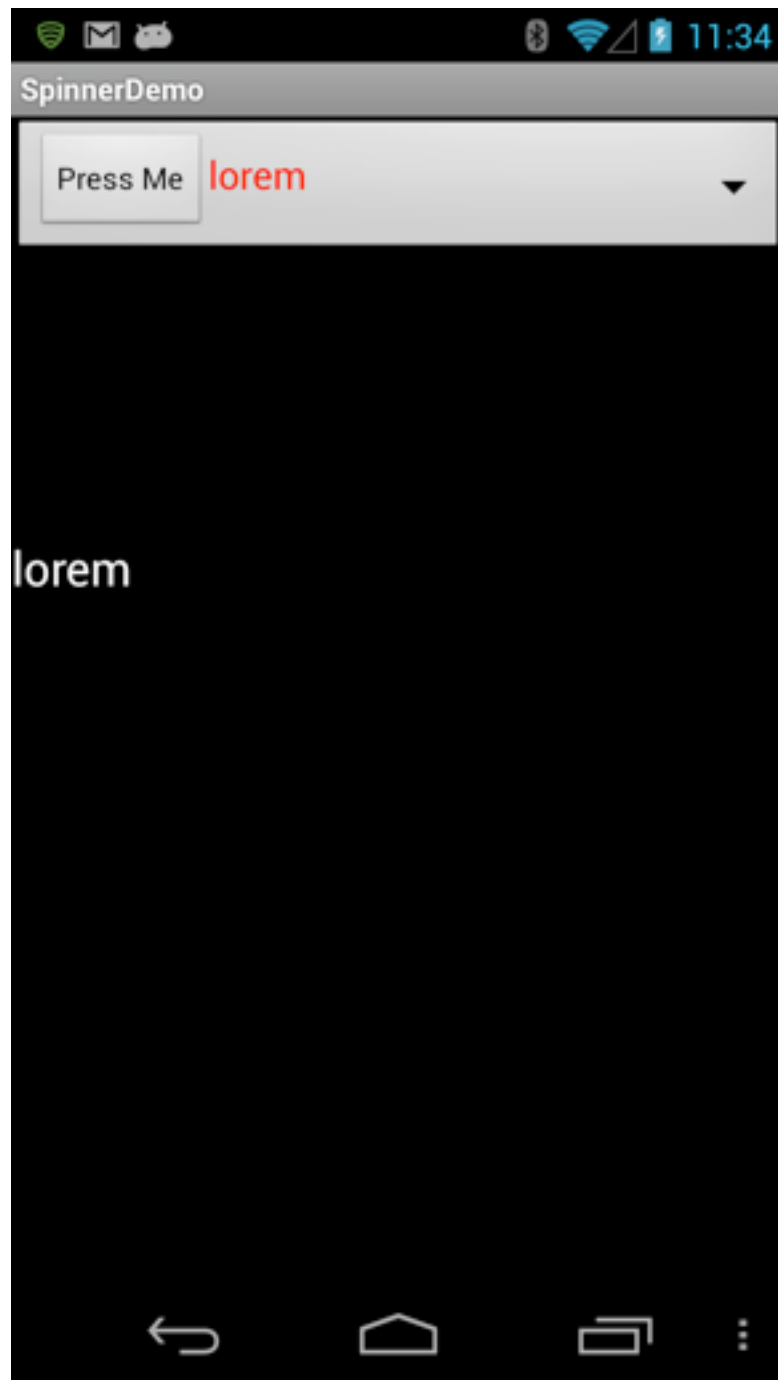
```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal" >
    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Press Me" />

    <TextView
        android:id="@+id/listData"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Text"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textColor="#f00" />
</LinearLayout>
```

Modified onCreate

```
public void onCreate(Bundle icle) {  
    super.onCreate(icle);  
    setContentView(R.layout.main);  
    selection=(TextView)findViewById(R.id.selection);  
  
    Spinner spin=(Spinner)findViewById(R.id.spinner);  
    spin.setOnItemSelectedListener(this);  
  
    ArrayAdapter<String> adapter=new ArrayAdapter<String>(this,  
                                                         R.layout.spinner_row, R.id.listData, items);  
  
    spin.setAdapter(adapter);  
}
```


More Customization



res/layout/spinner_dropdown_row.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="horizontal" >

    <RatingBar
        android:id="@+id/ratingBar1"
        android:layout_width="100dp"
        android:layout_height="wrap_content" />

    <TextView
        android:id="@+id/listData"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="@dimen/padding"
        android:text="Text"
        android:textAppearance="?android:attr/textAppearanceMedium"
        android:textColor="#f00" />

</LinearLayout>
```

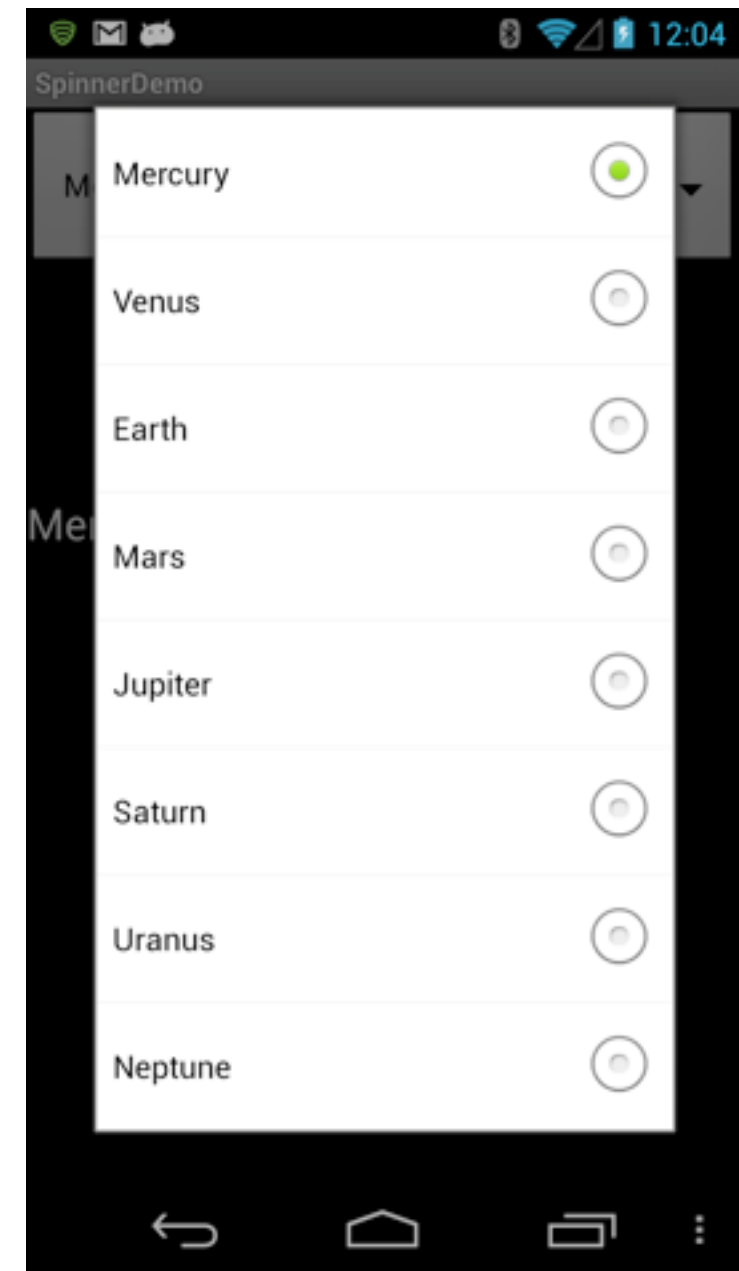
Modified onCreate

```
public void onCreate(Bundle icle) {  
    super.onCreate(icle);  
    setContentView(R.layout.main);  
    selection=(TextView)findViewById(R.id.selection);  
  
    Spinner spin=(Spinner)findViewById(R.id.spinner);  
    spin.setOnItemSelectedListener(this);  
  
    ArrayAdapter<String> adapter=new ArrayAdapter<String>(this,  
        R.layout.spinner_row, R.id.listData, items);  
  
    adapter.setDropDownViewResource(R.layout.spinner_dropdown_row);  
    spin.setAdapter(adapter);  
}
```

Yet Another Version

Bad idea to embed UI strings in source code

Store them in `res/values/strings.xml`



res/values/strings.xml

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <string name="app_name">SpinnerDemo</string>
    <string-array name="Planets">
        <item>Mercury</item>
        <item>Venus</item>
        <item>Earth</item>
        <item>Mars</item>
        <item>Jupiter</item>
        <item>Saturn</item>
        <item>Uranus</item>
        <item>Neptune</item>
    </string-array>
</resources>
```

OnCreate

```
public class SpinnerDemo extends Activity implements AdapterView.OnItemClickListener {  
    private TextView selection;  
  
    @Override  
    public void onCreate(Bundle icle) {  
        super.onCreate(icle);  
        setContentView(R.layout.main);  
        selection=(TextView)findViewById(R.id.selection);  
  
        Spinner spin=(Spinner)findViewById(R.id.spinner);  
        spin.setOnItemClickListener(this);  
  
        ArrayAdapter<CharSequence> adapter=ArrayAdapter.createFromResource(this,  
            R.array.Planets, android.R.layout.simple_spinner_dropdown_item);  
        spin.setAdapter(adapter);  
    }  
}
```

Getting the selected item

```
public void onItemSelected(AdapterView<?> parent, View v, int position, long id) {  
    String selectedPlanet = parent.getItemAtPosition(position).toString();  
    selection.setText(selectedPlanet);  
}
```

```
public void onNothingSelected(AdapterView<?> parent) {  
    selection.setText("");  
}
```

Using List of Objects



Data Class

```
public class Person {  
    String firstName;  
    String lastName;  
  
    public Person(String firstName, String lastName) {  
        this.firstName = firstName;  
        this.lastName = lastName;  
    }  
  
    public String toString() {  
        return this.firstName + " " + this.lastName;  
    }  
}
```

Defining the List

```
public class SpinnerDemo extends Activity
    implements AdapterView.OnItemClickListener {
    private TextView selection;
    private ArrayList<Person> people = new ArrayList<Person>();

    private void initializePeople() {
        people.add( new Person("Roger", "Whitney"));
        people.add( new Person("Eli", "Whitney"));
    }
}
```

OnCreate

```
public void onCreate(Bundle icle) {  
    super.onCreate(icle);  
    setContentView(R.layout.main);  
    selection=(TextView)findViewById(R.id.selection);  
    initializePeople();
```

```
    Spinner spin=(Spinner)findViewById(R.id.spinner);  
    spin.setOnItemSelectedListener(this);
```

```
        ArrayAdapter<Person> adapter=new ArrayAdapter<Person>(this,  
            android.R.layout.simple_spinner_item, people);
```

```
        adapter.setDropDownViewResource(  
            android.R.layout.simple_spinner_dropdown_item);
```

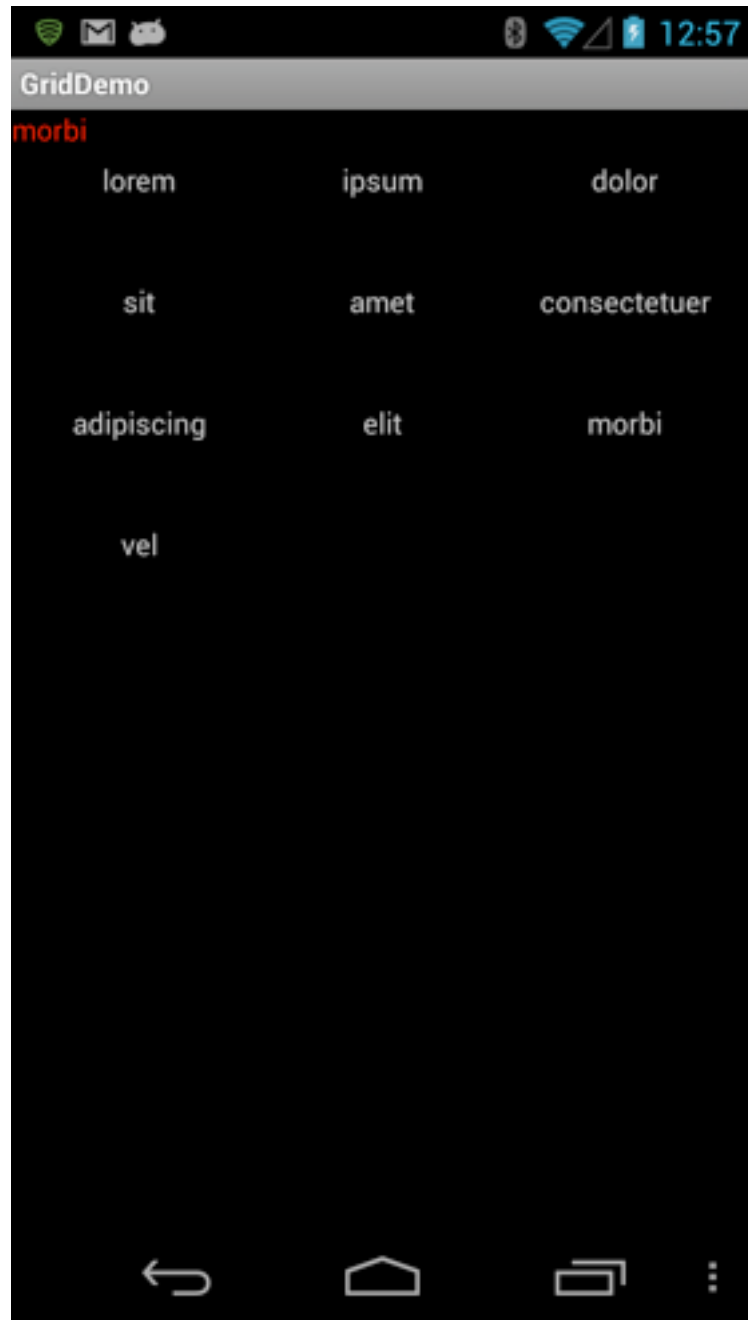
```
        spin.setAdapter(adapter);  
    }
```

Getting the selected Item

```
public void onItemSelected(AdapterView<?> parent, View v, int position, long id) {  
    selection.setText(people.get(position).toString());  
}
```

```
public void onNothingSelected(AdapterView<?> parent) {  
    selection.setText("");  
}
```

Grid View



Grid Activity

```
public class GridDemo extends Activity
    implements AdapterView.OnItemClickListener {
    private TextView selection;
    private static final String[] items={"lorem", "ipsum", "dolor",
                                         "sit", "amet","consectetuer", "adipiscing", "elit", "morbi", "vel"};

    @Override
    public void onCreate(Bundle icle) {
        super.onCreate(icle);
        setContentView(R.layout.main);
        selection=(TextView)findViewById(R.id.selection);

        GridView grid=(GridView) findViewById(R.id.grid);
        grid.setAdapter(new ArrayAdapter<String>(this,
                                                R.layout.cell,
                                                items));

        grid.setOnItemClickListener(this);
    }
}
```

Handling the click

```
public void onItemClick(AdapterView<?> parent, View v,  
    int position, long id) {  
    selection.setText(items[position]);  
}
```

main layout

```
<?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:id="@+id/selection"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:textColor="#f00"/>
    <GridView
        android:id="@+id/grid"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:verticalSpacing="40dip"
        android:horizontalSpacing="5dip"
        android:numColumns="auto_fit"
        android:columnWidth="100dip"
        android:stretchMode="columnWidth"
        android:gravity="center"/>
</LinearLayout>
```


View for each cell

res/layout/cell.xml

```
<?xml version="1.0" encoding="utf-8"?>
<TextView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="14dip"
/>
```

Adapter Options

All the same adapter options give in Spinner examples work here