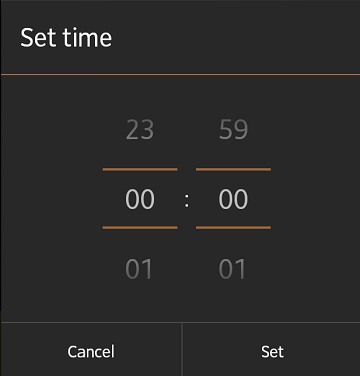
Let’s talk about Dialogs

You need to deal with the dialog before you can continue.



Here’s another dialog, the TimePickerDialog.

It’s all about class: The Dialog class

All dialogs are part of the Dialog class.

We’re going to be using the AlertDialog sub-class.

You’ll use it for most of your dialogs. It determines what the dialog looks like.

Put it inside a container: Enter the DialogFragment

It’s best to put the dialog inside a fragment. We use the DialogFragment for this.

It takes care of the dialog’s lifecycle, such as when the screen orientation changes or the user presses the back button. No more crashes!

The DialogFragment was added in Android 3.0 (Honeycomb, API level 11).

We’re going to use the support library version because we want to target devices from Android 1.6 (Donut, API level 4) and higher.

Let’s build a simple alert dialog like the “choose a drink dialog” displayed above.

The AlertDialog

Here are a few things we need to be aware of:

1. We create the dialog in a dialog fragment
2. We show the dialog fragment in an activity.
3. We use an interface to communicate between the fragment and the activity, to tell the activity when the buttons are pressed or list items selected

So here’s the dialog fragment

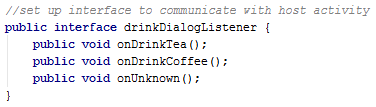
First off, make sure that you are importing the correct class:

Android DialogFragment

The interface

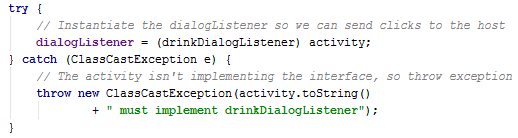
You need an interface to tell the hosting activity which dialog button was pressed or which list item was selected.

Here’s our dialog fragment’s interface. Note the three method calls. These methods are in the host activity.



The host activity must implement this interface to be able to receive the action button clicks

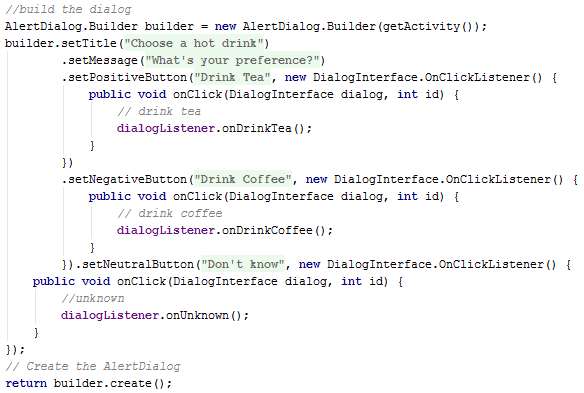
We use a try/catch statement inside the fragment’s onAttach() method to ensure that the activity is implementing the interface:



An exception is thrown if the activity does not implement the interface

Create the dialog

Here’s the code for creating the dialog:



We use AlertDialog.Builder to build the dialog and set the title, message and buttons. There’s a limit of three buttons:

* Positive button – usually used for accepting (we use it for drinking tea!)
* Negative button – usually used for cancelling the dialog (we use it for drinking coffee!)
* Neutral button – usually used for neither accepting nor cancelling the dialog (we use it for the unknown!)

Each of the buttons has a listener, listening for the clicks. These are communicated back to the hosting activity via the interface.

You can even send data back to the activity via the interface. I’ll show you how later.

Now for the host activity

Make sure that the activity imports the support FragmentActivity class:

Android support library FragmentActivity

And that it also implements the fragment’s interface:

Dialog activity implements interface

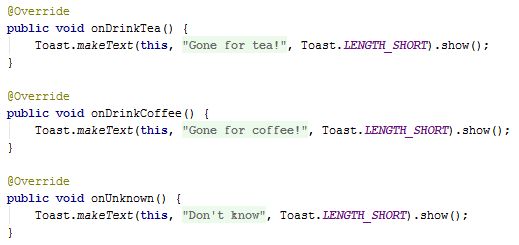
This is the code that shows the dialog. It creates a new dialog fragment:

Show Android AlertDialog

Calling show() displays the dialog and adds the fragment, identified by the tag “DrinkDialogFragment”, to the fragment manager (which we get by calling getSupportFragmentManager())

The dialog pops up, the user presses a button and an appropriate [Toast message](http://101apps.co.za/articles/making-toast.html) is displayed.

Here are the methods called by the fragment’s interface when the action buttons are clicked:



Dismissing the dialog

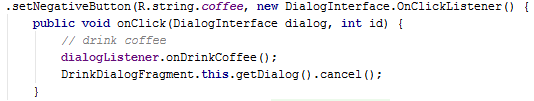
* Touching any of the buttons automatically dismisses the dialog.
* Touching any dialog list item also dismisses the dialog except if it’s a multi-choice list with radio buttons and check boxes. These need a button.
* You can also dismiss the dialog in your code by calling dismiss() on the dialog. We’ll cover this later.

Cancel the dialog

Cancelling a dialog means you’ve left it without completing the task. You can cancel a dialog by:

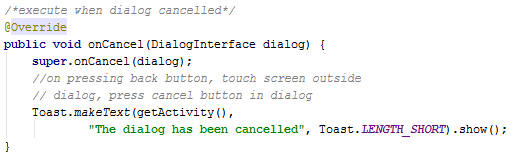
* Pressing the negative (or cancel) button - but you need to include some code (see below)
* Touching the screen outside of the dialog
* Pressing the back button

I’ve edited the negative button to look like this:



Now whenever the user chooses to drink coffee, the dialog is cancelled!

Calling cancel() executes the onCancel() method (which has to be included in the fragment). Here’s the code:



And there you have an alert dialog!

I hope you found this tutorial helpful.