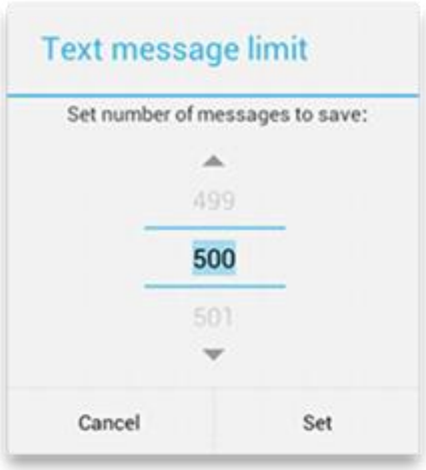
Dialog

a. What is a dialog?

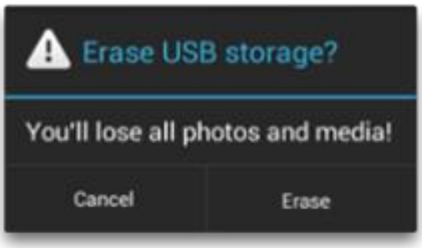
A dialog is a small window that prompts the user to make a decision or enter additional information. A dialog does not fill the screen and is normally used for model events that require users to take an action before they can proceed.

Example:



Text message limit

Set number of messages to save:



Erase USB storage?

You'll lose all photos and media!

b. How to create custom dialog?

To create a dialog create a class that inherits from the DialogFragment class and create an AlertDialog object in the onCreateDialog() method

```
public class LinkDialogFragment extends DialogFragment {
@Override
public Dialog onCreateDialog(Bundle savedInstanceState) {
AlertDialog.Builder builder = new
AlertDialog.Builder(getActivity());
builder.setTitle(getText(R.string.title).toString());
.setNegativeButton(R.string.cancel, new
DialogInterface.OnClickListener(){
public void onClick(DialogInterface dialog, int id) {
listener.onDialogNegativeClick(LinkDialogFragment.this);
}
});
return builder.create();
}}
```

c. How to use existing dialog?

To display the dialog we just created we have to create an instance of the class in our business and invoke the show() method so that the dialog appears.

With getSupportFragmentManager() call method get access to FragmentManager that is responsible for managing the fragments of the application.

```
//LinkDialogFragment object
LinkDialogFragment t linkDialogFragment=new
LinkDialogFragment();
//fragment parameters
Bundle args = new Bundle();
args.putString(“CLOSE", close);
linkDialogFragment.setArguments(args);
linkDialogFragment.show(getSupportFragmentManager(), "");
```