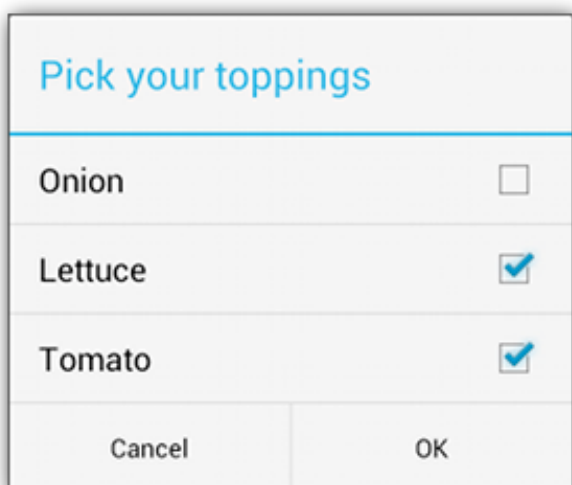
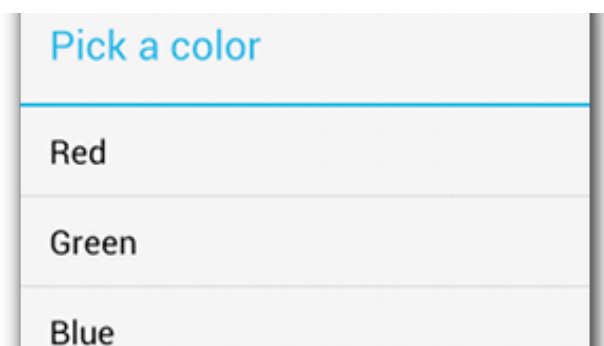


其他对话框

知识解析

除了常见的对话框外，还可以将对话框设置成特殊用途的对话框，例如：可以让对话框变成单选或者多选的对话框，方便用户选择。



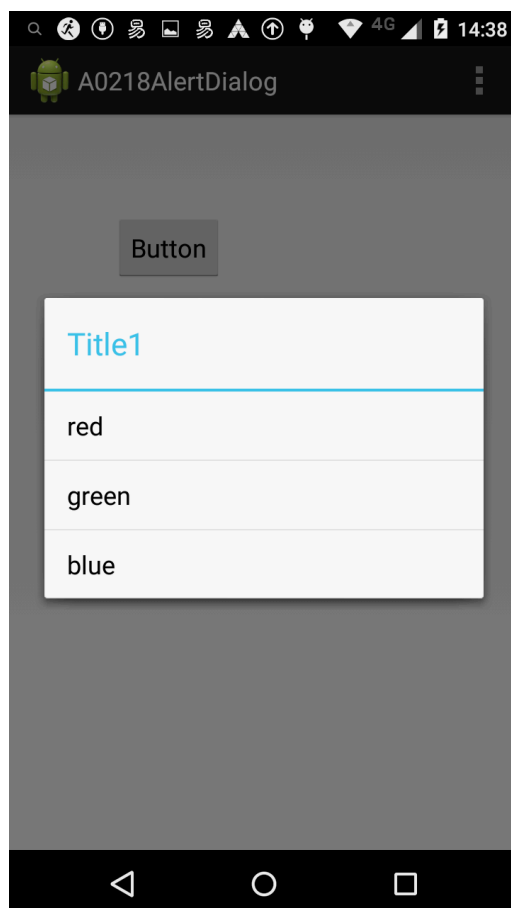
以上 2 个对话框，分别可以让用户进行单选或者多选，此时可以使用 Alert.Builder 上

的方法，轻易达成这个效果：

`setItems(int itemId, DialogInterface.OnClickListener listener)/ setItems(CharSequence[] items, DialogInterface.OnClickListener listener)`：设置单选选项。第一个参数设置选项，可以是一个字符串数组或者是资源文件中定义的数组资源，第二个参数用于监听当用户点击选项后的事件。

`setMultiChoiceItems(CharSequence[] items, boolean[] checkedItems, DialogInterface.OnMultiChoiceClickListener listener)/setMultiChoiceItems(int itemId, boolean[] checkedItems, DialogInterface.OnMultiChoiceClickListener listener)`：这 2 个方法用预设值多选选项。其中第二个参数用于指定各选项默认的选中（checked）状态。

功能演示



实战操作

```
public class MainActivity extends Activity {  
    private int i = 1;  
  
    @Override  
    protected Dialog onCreateDialog(int id) {  
        Dialog dialog = null;  
        switch(id) {  
            case ALERT_DIALOG:  
                AlertDialog.Builder builder = new  
AlertDialog.Builder(this);  
                builder.setTitle("选择颜色");  
                builder.setCancelable(false);  
                builder.setItems(R.array.colors, new  
DialogInterface.OnClickListener() {  
@Override  
                    public void onClick(DialogInterface dialog, int which) {  
                        Log.i("item", "item:"+which);  
                    }  
                });  
                dialog = builder.create();  
                break;  
            case COMMON_DIALOG:  
                dialog = null;  
                break;  
        }  
        return dialog;  
    }  
  
    @Override  
    protected void onPrepareDialog(int id, Dialog dialog) {  
        dialog.setTitle("Title"+i++);  
    }  
}
```

```
private final int ALERT_DIALOG = 1;

private final int COMMON_DIALOG = 2;

@Override

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_main);

    Button btn = (Button) this.findViewById(R.id.button1);

    btn.setOnClickListener(new OnClickListener() {

        @Override

        public void onClick(View v) {

            showDialog(ALERT_DIALOG);});});}
```

职业素质

Activities 提供了一种方便管理的创建、保存、回复的对话框机制，例如 onCreateDialog(int),onPrepareDialog(int,Dialog),showDialog(int),dismissDialog(int)等方法，如果使用这些方法的话，Activity 将通过 getOwnerActivity()方法返回该 Activity 管理的对话框 (dialog).