**Пример модальных диалогов для Androi**И нет, я не

**package** ua.kiev.mabp;

**import** android.app.Activity;

**import** android.app.AlertDialog;

**import** android.app.Dialog;

**import** android.app.ProgressDialog;

**import** android.content.DialogInterface;

**import** android.os.Bundle;

**import** android.view.View;

**import** android.widget.Button;

**import** android.widget.LinearLayout;

**import** java.util.Random;

**public** **class** **MyActivity** **extends** **Activity** {

// список диалогов

**private** **static** **final** **int** CONFIRM = 1;

**private** **static** **final** **int** PROGRESS = 2;

// счетчик вызовов

**private** **int** counter;

/\*\*

\* Метод вызывается один раз при создании активности

\*

\* **@param** savedInstanceState предыдущее состояние активности

\*/

@Override

**public** **void** **onCreate**(Bundle savedInstanceState) {

**super**.onCreate(savedInstanceState);

// Создаем кнопку

Button button = **new** Button(**this**);

button.setText("click me!");

// вешаем на неё листенер

button.setOnClickListener(**new** View.OnClickListener() {

**public** **void** **onClick**(View v) {

// котрый покажет диалог CONFIRM

showDialog(CONFIRM);

}

});

// сетим кнопку в основной лайаут

setContentView(button, **new** LinearLayout.LayoutParams(

LinearLayout.LayoutParams.WRAP\_CONTENT,

LinearLayout.LayoutParams.WRAP\_CONTENT));

}

/\*\*

\* Метод вызывается один раз на каждый диалог

\*

\* **@param** id номер диалога онисан в начале класса

\* **@return** диалог

\*/

@Override

**protected** Dialog **onCreateDialog**(**int** id) {

**super**.onCreateDialog(id);

**switch** (id) {

**case** CONFIRM:

// новый лайаут для диалога

LinearLayout confirmView = **new** LinearLayout(**this**);

// создаем диалог с параметрами и двумя кнопками

**return** **new** AlertDialog.Builder(MyActivity.**this**)

.setIcon(android.R.drawable.ic\_dialog\_info)

.setView(confirmView)

.setTitle("Title")

.setMessage("Message")

// положительная кнопка с листенером который показывает диалог PROGRESS

.setPositiveButton("Click me!", **new** DialogInterface.OnClickListener() {

**public** **void** **onClick**(DialogInterface dialog, **int** whichButton) {

// запустит новый диалог загрузки

showDialog(PROGRESS);

// в новом потоке

**new** Thread(**new** Runnable() {

**public** **void** **run**() {

// запустит фоновую процедуру

wasteMyTime();

// по завершении которой в основном потоке

MyActivity.**this**.runOnUiThread(**new** Runnable() {

**public** **void** **run**() {

// диплог загрузки закроется

dismissDialog(PROGRESS);

}

});

}

}).start();

}

})

// отрицатильная кнопка просто скроет диалог

.setNegativeButton("Cancel", **null**)

.create();

**case** PROGRESS:

// Диалог загрузки

ProgressDialog dialog = **new** ProgressDialog(**this**);

dialog.setTitle("Please, waite...");

dialog.setMessage("Dummy message about loading...");

// без процентов

dialog.setIndeterminate(**true**);

// можно закрыть кнопкой телефона

dialog.setCancelable(**true**);

**return** dialog;

**default**:

**return** **null**;

}

}

/\*\*

\* Метод вызывается каждый раз перед появлением диалога

\*

\* **@param** id номер появляющегося диалога

\* **@param** dialog сам диалог

\*/

@Override

**protected** **void** **onPrepareDialog**(**int** id, Dialog dialog) {

**super**.onPrepareDialog(id, dialog);

**switch** (id) {

**case** CONFIRM:

((AlertDialog) dialog).setMessage("Этот диалог был вызван " + (++counter) + " раз");

**break**;

**default**:

**break**;

}

}

/\*\*

\* Метод просто тратит время

\*/

**public** **static** **void** **wasteMyTime**() {

Random random = **new** Random();

**for** (**int** i = 0; i < 10000; i++) {

random.nextGaussian();

}

}

}

Вот так это выглядит в новом красивом но не очень удобном (я уже привык к прошлому) эмуляторе.

