

Hướng dẫn lập trình cơ bản với Android

Part 9

B5: Time to coding. Tới thư mục src/Example.java và thay đổi nội dung file như sau:

Mã:

```
package at.exam;

import java.util.ArrayList;

import android.app.Activity;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ListView;

public class Example extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        //Tạo mảng để chứa String nội dung công việc và
        //giờ
        final ArrayList<String> arrayWork = new
        ArrayList<String>();
        //Adapter dùng để kết nối mảng với List View
        final ArrayAdapter<String> arrayAdapter = new
        ArrayAdapter<String>(this,
            android.R.layout.simple_list_item_1,
            arrayWork);
```

```
//Các EditText để vào nội dung công việc được
lấy về từ XML
    final EditText workEnter = (EditText)
findViewById(R.id.work_enter);
    final EditText hourEdit = (EditText)
findViewById(R.id.hour_edit);
    final EditText minuteEdit = (EditText)
findViewById(R.id.minute_edit);

//Button khi nhấn sẽ thêm công việc vào
ListView
    final Button button = (Button)
findViewById(R.id.button);

//ListView chứa danh sách công việc
    final ListView list = (ListView)
findViewById(R.id.list);
//Cần set Adapter cho list để biết sẽ lấy nội
dung từ mảng arrayWork
    list.setAdapter(arrayAdapter);

//Định nghĩa Listener xử lý sự kiện nhấn vào
button
    OnClickListener add = new OnClickListener() {
        @Override
        public void onClick(View v) {
            //Nếu 1 trong 3 Edit Text không có nội
dung thì hiện lên thông báo
            if
(workEnter.getText().toString().equals("") ||
hourEdit.getText().toString().equals("") ||
minuteEdit.getText().toString().equals("")) {
                AlertDialog.Builder builder = new
AlertDialog.Builder(Example.this);
                builder.setTitle("Info missing");
                builder.setMessage("Please enter
all information of the work");
```

```
builder.setPositiveButton("Continue", new
DialogInterface.OnClickListener() {
    public void
onClick(DialogInterface dialog, int which) {
    // TODO Auto-generated
method stub
    }
});
builder.show();
}
//Lấy nội dung công việc và thời gian
ra từ Edit Text và đưa vào list
else {
    String str =
workEnter.getText().toString() + " - "
+
hourEdit.getText().toString() + ":"
+
minuteEdit.getText().toString();
arrayWork.add(0, str);

arrayAdapter.notifyDataSetChanged();
workEnter.setText("");
hourEdit.setText("");
minuteEdit.setText("");
}
}

};

//set Listener cho button
button.setOnClickListener(add);
}
}
```

Mình đã chú thích đầy đủ và đoạn code cũng khá dễ hiểu. Tuy nhiên cần lưu ý 2 vấn đề ở đây.

- Khởi tạo đối tượng ArrayAdapter: Các bạn thấy đối số truyền vào là (**this, android.R.layout.simple_list_item_1, arrayWork**). This là đối số của lớp Context

(ở đây chính là activity Example). Bạn sẽ gặp Context trong rất nhiều khởi tạo các lớp và nên hiểu Context có ý nghĩa gì. Mình xin đưa ra giải thích của anh Giáp (thank mr giaplv):

Quote:

Context thuộc ***android.content*** (*android.content.Context*).

Là một Interface (lớp giao tiếp) chứa hầu hết thông tin về môi trường ứng dụng của android, có nghĩa là mọi thao tác, tương tác với hệ điều hành điều phải qua lớp này.

Nó là một lớp abstract (trừu tượng) cung cấp cho những lớp khác các phương thức để tương tác với hệ thống Android.

Nó cho phép truy cập tới các nguồn tài nguyên (resources) đã được định nghĩa và các lớp khác. Ví dụ như nó có thể khởi tạo và chạy các activities, các broadcast và các intents,... Chúng ta coi như Context là một lớp ở mức ứng dụng (Application level- liên quan tới hệ thống).

Tóm lại context giúp chúng ta dễ dàng truy cập và tương tác tới các tài nguyên của hệ thống, các thông tin, các dịch vụ (services), các thông số cấu hình, database, wallpaper, danh bạ, cuộc gọi, kết nối, chế độ rung (vibrator),...

***sở dĩ hầu hết các lớp có liên quan tới UI (layout, button, textview, imageview, listview,...) đều phải super tới Context vì bản thân nó đảm nhiệm việc truy cập resource (R.id, R.layout,...). Nếu chúng ta không tham chiếu tới Context class thì đương nhiên không thể dùng tới các resources mà chúng ta đã tạo ra.

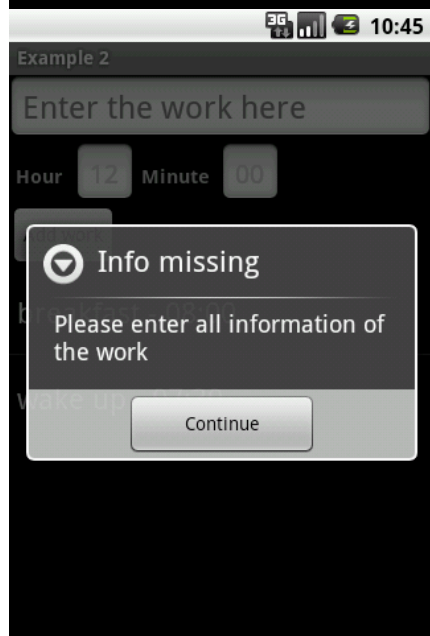
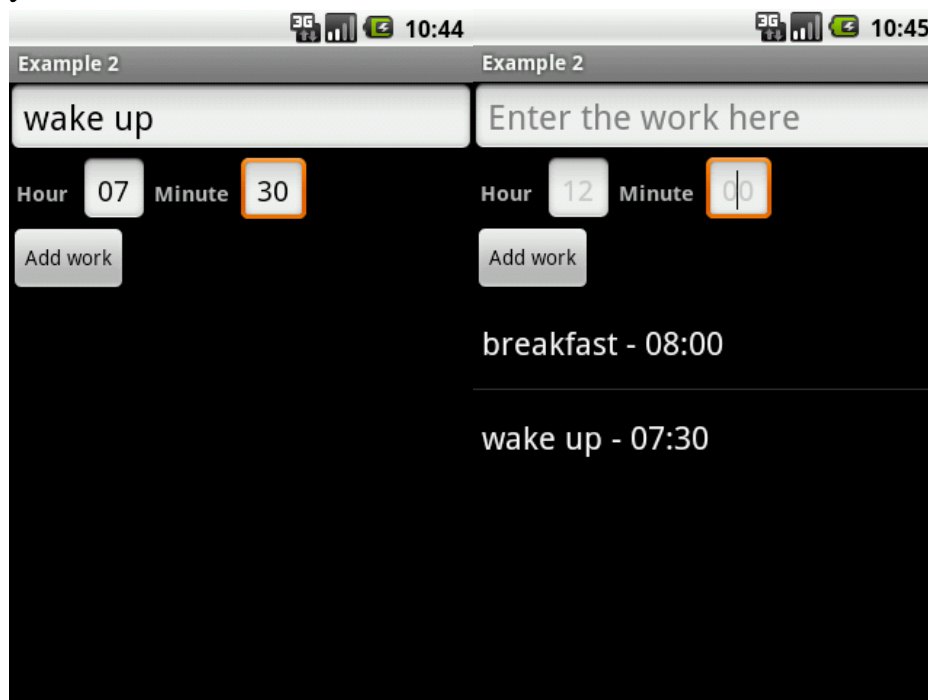
Tiếp theo là android.R.layout.simple_list_item_1, đối này định nghĩa cách thể hiện item (ở đây là String) trong List View. Các bạn hãy ghi nhớ android.R.* là các tài nguyên (resource) có sẵn của Android cho phép bạn truy cập và sử dụng. Sau này khi hướng dẫn tạo custom View cho List View mình sẽ đề cập lại vấn đề này.

Cuối cùng arrayWork chính là mảng cần được bind của adapter.

- AlertDialog là lớp cho phép đưa ra 1 hộp thoại, thường dùng để đưa ra thông tin hoặc cảnh báo đơn giản. Trong code mình tạo 1 builder, tạo tiêu đề (title) cho nó, đưa ra thông báo (message) và cuối cùng là tạo 1 positive button (nhưng không định nghĩa xử lý khi nhấn nút này, vì vậy nếu bạn nhấn nút thì dialog sẽ chỉ đơn giản thực hiện việc đóng lại).

B6: Tiến hành chạy thử chương trình. Run as -> Android Application. Enjoy

yourself 🌱



Hướng dẫn lập trình cơ bản với Android