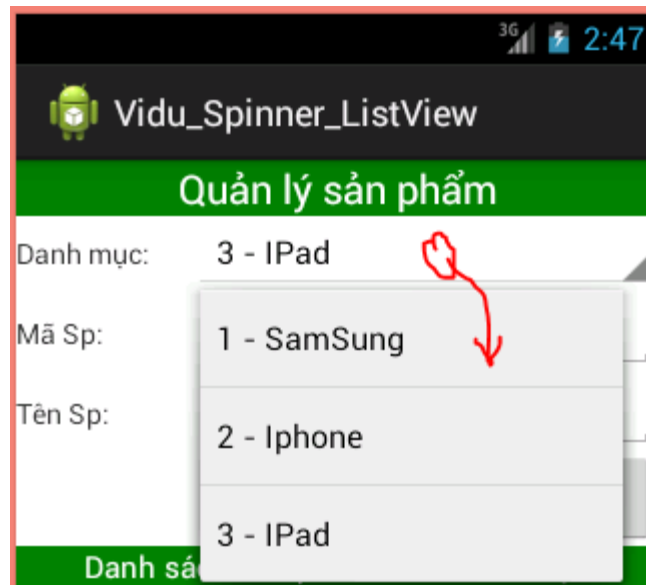


Bài tập 16: Kết hợp Spinner với ListView trong Android

Ở [bài tập 15](#) bạn đã làm quen được với **Spinner**, trong bài tập này là ví dụ về cách kết hợp giữa **Spinner** với **ListView**. Thường thì 2 control này đi với nhau. **Spinner** dùng để lưu trữ danh mục còn **ListView** lưu trữ danh sách của từng danh mục. Sau đây là ví dụ về quản lý sản phẩm.

- Trong minh họa Danh mục các sản phẩm được lưu vào **Spinner**, Khi được chọn sẽ xổ ra danh sách hàng như gợi ý. Danh mục nào được chọn chương trình sẽ tìm kiếm danh sách sản phẩm thuộc danh mục đó.



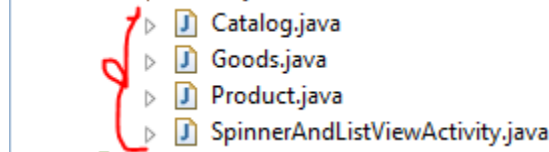
- Nếu chọn 2- Iphone thì nó sẽ load toàn bộ sản phẩm là Iphone vào ListView bên dưới:



- Chương trình cung cấp nút “**Nhập SP**”, khi người sử dụng nhập thông tin cho sản phẩm và nhấn nút này thì chương trình sẽ lưu sản phẩm vào đúng với danh mục được chọn trong **Spinner** đồng thời cập nhật vào **ListView** bên dưới.

- Vậy cần có **ArrayList** + **ArrayAdapter** cho Spinner Và **ArrayList** + **ArrayAdapter** cho ListView, tức là bạn phải có 2 cặp (4 đối tượng trên)

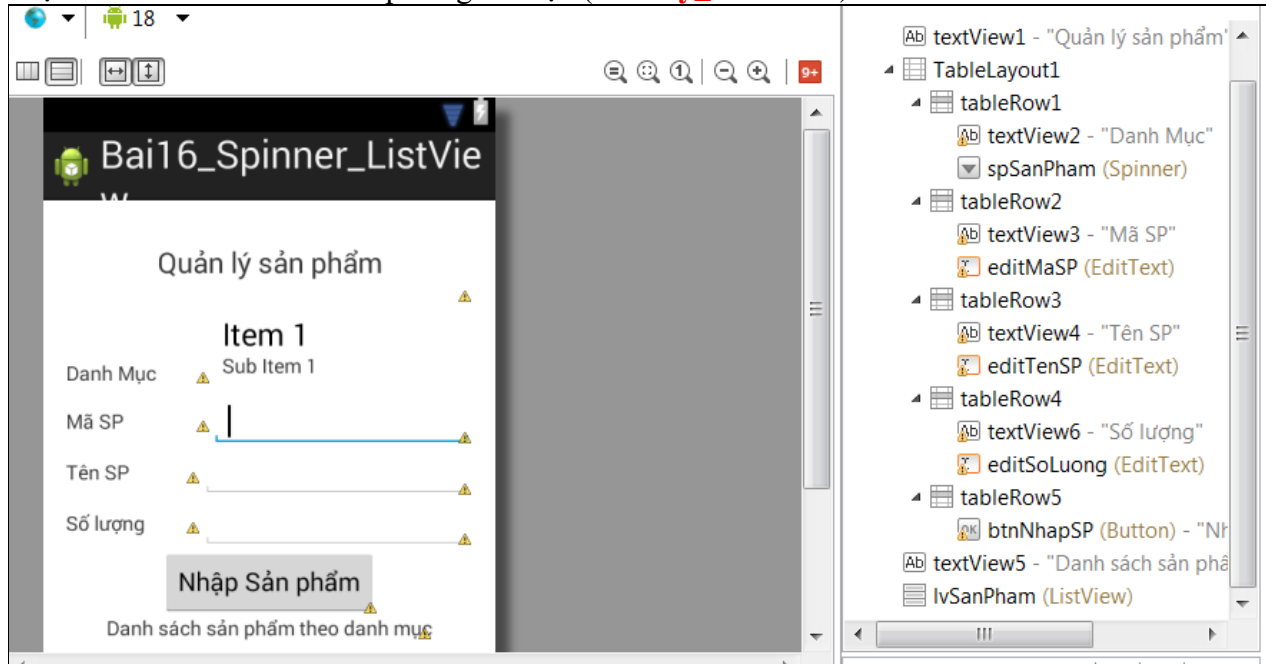
- Bạn xem cấu trúc chương trình:



- Ở trên bạn thấy có 3 class: **Goods**, **Product**, **Catalog** :

1. **Goods** : mã sp, tên sp.
2. **Product** sẽ kế thừa từ **Goods** và có thêm số lượng,
3. **Catalog** – là danh sách các sản phẩm: có mã danh mục, tên danh mục và có danh sách sản phẩm

-Bạn xem cấu trúc XML cho phần giao diện (**activity_main.xml**):



- Tương tự như các bài tập trước, chúng ta phải đặt id cho các control. Bài tập này bạn tự nhìn vào để xem Tôi đặt Id như thế nào, Tôi không nhắc lại nữa.

- Chúng ta lần lượt xem nội dung coding của các class trong phần xử lý nghiệp vụ:

- 1) Class Goods:

```
public class Good {
    public String maSP;
    public String tenSP;
    //hàm tạo
    public Good(String maSP, String tenSP) {
        super();
        this.maSP = maSP;
        this.tenSP = tenSP;
    }
    public Good(){}
    //các hàm getter/setter
    public String getMaSP() {
        return maSP;
    }
    public void setMaSP(String maSP) {
        this.maSP = maSP;
    }
    public String getTenSP() {
        return tenSP;
    }
    public void setTenSP(String tenSP) {
```

```

        this.tenSP = tenSP;
    }
    //ghi đề toString
    @Override
    public String toString() {
        return "ma SP=" + maSP + "\t tenSP=" + tenSP ;
    }
    //chèn equal & hashCode
    @Override
    public int hashCode() {
        final int prime = 31;
        int result = 1;
        result = prime * result + ((maSP == null) ? 0 : maSP.hashCode());
        return result;
    }
    @Override
    public boolean equals(Object obj) {
        if (this == obj)
            return true;
        if (obj == null)
            return false;
        if (getClass() != obj.getClass())
            return false;
        Good other = (Good) obj;
        if (maSP == null) {
            if (other.maSP != null)
                return false;
        } else if (!maSP.equals(other.maSP))
            return false;
        return true;
    }
}

```

- 2 – class Product:

//lưu thông tin sản phẩm.

```

public class Product extends Good{
    private int soLuong;
    public Product() {
        super();
        soLuong=0;
    }
    public Product(String maSP, String tenSP, int soLuong) {
        super(maSP, tenSP);
        this.soLuong = soLuong;
    }
    @Override
    public String toString() {
        return super.toString() + " \t số lượng="+soLuong;
    }
    public int getSoLuong() {
        return soLuong;
    }
    public void setSoLuong(int soLuong) {
        this.soLuong = soLuong;
    }
}

```

3 – class Catalog:

```

import java.util.ArrayList;
//lớp chứa danh sách sản phẩm
public class Catalog {
    private String maDM,tenDM;

```

```

    private ArrayList<Product> dsSanPham =null;
    public Catalog(String ma,String ten) {
        // TODO Auto-generated constructor stub
        maDM=ma;tenDM=ten;
        dsSanPham = new ArrayList<Product>();
    }
    /*
     * Kiểm tra sản phẩm đã tồn tại trong danh mục chưa
     * @para p - Product
     * @return true nếu tồn tại
     */
    public boolean kiemTraSanPham(Product p)
    {
        for(Product p1:dsSanPham)
        {
            if (p1.getMaSP().trim().equalsIgnoreCase(p.getMaSP().trim()))
                return true;
        }
        return false;
    }
    /*
     * Thêm 1 sản phẩm vào danh mục
     * thêm thành công true
     */
    public boolean addSP(Product p)
    {
        boolean kiemTra=kiemTraSanPham(p);
        if (!kiemTra)
        {
            dsSanPham.add(p);
            return true;
        }
        else return false;
    }
    public ArrayList<Product> getDsSanPham() {
        return dsSanPham;
    }
    @Override
    public String toString() {
        return maDM + "-" + tenDM ;
    }
}

```

4 – class MainActivity – xử lý nghiệp vụ trong Activity:

```

public class MainActivity extends Activity {
    Spinner spinDanhMuc;
    EditText editMaSP,editTenSP,editSoLuong;
    Button btnNhap;
    ListView lvSanPham;
    //khai báo cặp đối tượng dùng cho Spinner
    ArrayList<Catalog> arraySpinner=new ArrayList<Catalog>();
    ArrayAdapter<Catalog> adapterSpinner =null;

    //khai báo cặp đối tượng dùng cho listView
    ArrayList<Product> arrayListView=new ArrayList<Product>();
    ArrayAdapter<Product> adapterListView=null;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        //lấy các control gắn với Activity
        getWidgetsControl();
    }
}

```

```

        //Giả dữ liệu mặc định
        fakeDataCatalog();
        addEventsFormWidGets();
    }
    private void getWidgetsControl()
    {
        //sv tự lấy các điều khiển trên form gắn với code cho hợp lý
        //cấu hình cho Spinner
        adapterSpinner=new ArrayAdapter<Catalog>(
            this,
            android.R.layout.simple_spinner_item,
            arraySpinner
        );

        adapterSpinner.setDropDownViewResource(
            android.R.layout.simple_spinner_dropdown_item);
        spinDanhMuc.setAdapter(adapterSpinner);
        //cấu hình cho listView
        adapterListView=new ArrayAdapter<Product>(
            this,
            android.R.layout.simple_list_item_1,
            arrayListView
        );
        lvSanPham.setAdapter(adapterListView);
    }
    //hàm giả dữ liệu tạo 4 danh mục mã định cho Spinner
    private void fakeDataCatalog()
    {
        Catalog cat1=new Catalog("1","SamSung");

        Catalog cat2=new Catalog("2","Nokia");
        Catalog cat3=new Catalog("3","IPAD");
        Catalog cat4=new Catalog("4","HTC");

        arraySpinner.add(cat1);
        arraySpinner.add(cat2);
        arraySpinner.add(cat3);
        arraySpinner.add(cat4);

        adapterSpinner.notifyDataSetChanged();
    }
    /*
     * Hàm gán sự kiện cho Button và spinner
     */
    private void addEventsFormWidGets()
    {
        btnNhap.setOnClickListener(new OnClickListener() {

            @Override
            public void onClick(View v) {
                // TODO Auto-generated method stub
                addProductForCatalog();
            }
        });
        spinDanhMuc.setOnItemClickListener(new OnItemSelectedListener() {

            @Override
            public void onItemSelected(AdapterView<?> arg0, View arg1,
                int arg2, long arg3) {
                //mỗi lần chọn danh mục trong Spinner thì cập nhật listView
                loadListProductByCalalog(arraySpinner.get(arg2));
            }
        });
    }

```

```

        @Override
        public void onNothingSelected(AdapterView<?> arg0) {
        }
    });
}
private void addProductForCatalog()
{
    Product p=new Product();
    p.setMaSP(editMaSP.getText()+"");
    p.setTenSP(editTenSP.getText()+"");
    p.setSoLuong(Integer.parseInt(editSoLuong.getText()+""));
    Catalog c=(Catalog)spinDanhMuc.getSelectedItem();
    c.addSP(p);
    //cập nhật lại listview
    loadListProductByCalalog(c);
}
/*
 * lọc lại danh sách sản phẩm theo dạng mục và update lại listview
 * @see android.app.Activity#onCreateOptionsMenu(android.view.Menu)
 */
private void loadListProductByCalalog(Catalog c)
{
    //xóa danh sách cũ
    arrayListView.clear();
    //lấy lại danh sách mới từ catalog chonh trong Spinner
    arrayListView.addAll(c.getDsSanPham());
    //cập nhật lại listView
    adapterListView.notifyDataSetChanged();
}
}

```