

Bài tập 12: Ví dụ tổng hợp TextView, EditText, CheckBox, Button và ImageButton trong Android

Bài tập 12 Tôi sẽ làm một ví dụ tổng hợp + kết hợp với ArrayList để bạn củng cố thêm kiến thức về các control đã học. Viết chương trình thanh toán tiền bán sách và thống kê sách đã bán theo từng loại

Thông tin hóa đơn:

Tên Khách Hàng:

Số lượng sách:

☒ Khách hàng là VIP

Thành tiền: **2034000**

Thông tin thống kê:

Tổng số KH:

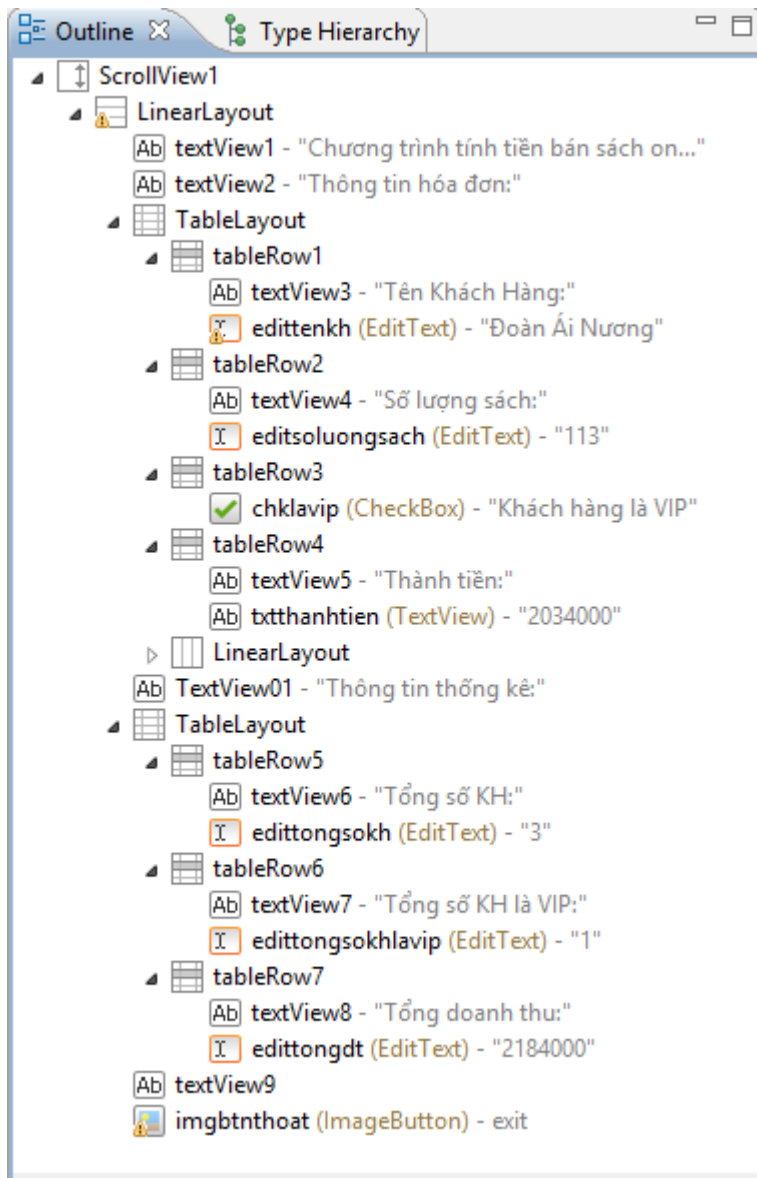
Tổng số KH là VIP:

Tổng doanh thu:

- Mô tả yêu cầu:

- Khi bấm nút Tính Thành Tiền (Tính TT) chương trình sẽ tính thành tiền biết rằng mỗi cuốn sách có đơn giá là 20000, nếu là khách hàng VIP thì giảm 10%
- Khi bấm nút Tiếp, chương trình sẽ lưu thông tin hóa đơn vừa tính Thành Tiền vào danh sách, đồng thời xóa trắng dữ liệu trong hóa đơn và cho focus tới EditText Tên khách hàng
- Khi bấm nút Thống kê, chương trình sẽ hiển thị thông tin vào mục Thông tin thống kê: tổng số KH, tổng số KH VIP và tổng doanh thu
- Khi bấm vào nút thoát (dùng ImageButton): hiển thị AlertDialog hỏi xem người sử dụng có chắc chắn muốn thoát hay không?
- Dùng ScrollView để chương trình có thể làm việc tốt hơn khi sử dụng các thiết bị có màn hình nhỏ.

- Outline giao diện:



Chú ý ImageButton có thuộc tính **android:src="@drawable/exit"**.

Cách thực hiện:

- Tạo một hình có tên *exit.png* rồi kéo vào thư mục *drawable* (bạn có thể tự tạo 1 thư mục mới tên là *drawable* vào trong ứng dụng của bạn).
 - Cách tìm file ảnh: Search trong window hay google sẽ tìm đc ảnh hợp lý.
- Bạn xem cấu trúc tập tin src của ứng dụng này:

- ▶ `DanhSachKhachHang.java`
- ▶ `KhachHang.java`
- ▶ `MainActivity.java`

- Ở trên có **MainActivity.java** (là màn hình chính của ứng dụng mà bạn thấy ở phần giới thiệu)
- class **KhachHang** dùng để lưu thông tin của khách hàng: Tên khách hàng, số lượng mua, thành tiền, là VIP hay không
- class **DanhSachKhachHang** dùng để lưu trữ các khách hàng mua sách, đồng thời cung cấp một số hàm như: tính tổng tiền, tính tổng số khách hàng, tính tổng số khách hàng VIP....

Dưới đây là chi tiết cho từng class:

class KhachHang:

```
package VUTHIDUONG.com;
public class KhachHang {
    private String tenKh;
    private int slmua;
    private boolean isVip;
    public static final int GIA=20000;
```

```

public KhachHang() {}
public KhachHang(String tenKh,int slmua,boolean isVip)
{
    this.tenKh=tenKh;
    this.slmua=slmua;
    this.isVip=isVip;
}
public String getTenKh() {return tenKh;}
public void setTenKh(String tenKh) {this.tenKh = tenKh;}
public int getSlmua() {return slmua;}
public void setSlmua(int slmua) {this.slmua = slmua;}
public boolean isVip() {return isVip;}
public void setVip(boolean isVip) {this.isVip = isVip;}
public double tinhThanhTien()
{
    return (!isVip?slmua*GIA:slmua*GIA*0.9);
}
}

```

Class DanhSachKhachHang:

```

package VUTHIDUONG.com;
import java.util.ArrayList;
public class DanhSachKhachHang {

    ArrayList<KhachHang>listKH=new ArrayList<KhachHang>();
    public void addKhachHang(KhachHang kh)
    {
        listKH.add(kh);
    }
    public double tongDoanhThu()
    {
        double tien=0.0;
        for(KhachHang kh:listKH) tien+=kh.tinhThanhTien();
        return tien;
    }
    public int tongKhachHang()
    {
        return listKH.size();
    }
    public int tongKhachHangVip()
    {
        int s=0;
        for(KhachHang kh:listKH)
        {
            if(kh.isVip()) s++;
        }
        return s;
    }
}

```

class MainActivity:

- Dùng để triệu gọi 2 class trên và thực thi các nghiệp vụ:

```

package VUTHIDUONG.com;

public class MainActivity extends Activity {
    Button btnTT,btnTiep,btnTK;
    ImageButton btnThoat;
    EditText editTen,editSl,editTongKh,editTongKhVip,editTongTT;
    TextView txtTT;
}

```

```

CheckBox chkVip;
DanhSachKhachHang danhSach=new DanhSachKhachHang();
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    getControls();
    addEvents();
}
private void getControls()
{
    //thực hiện lấy các điều khiển trên giao diện (SV tự làm)
}
private void addEvents()
{
    //bắt sự kiện thông qua 1 đối tượng của lớp tự tạo
    btnTT.setOnClickListener(new ProcessMyEvent());
    btnTiep.setOnClickListener(new ProcessMyEvent());
    btnTK.setOnClickListener(new ProcessMyEvent());
    btnThoat.setOnClickListener(new ProcessMyEvent());
}
private void doTinhTien()
{
    KhachHang kh=new KhachHang();
    kh.setTenKh(editTen.getText()+"");
    kh.setSlmua(Integer.parseInt(editSl.getText()+""));
    kh.setVip(chkVip.isChecked());
    txtTT.setText(kh.tinhThanhTien()+"");
    danhSach.addKhachHang(kh);
}
Private void doTiep()
{
    editTen.setText(???);
    editSl.setText(???);
    txtTT.setText(??);
    editTen.requestFocus();
}

private void doThongKe()
{
    editTongKh.setText(danhSach.tongKhachHang()+"");
    editTongKhVip.setText(danhSach.tongKhachHangVip()+"");
    ediTongTT.setText(danhSach.tongDoanhThu()+"");
}

private void doThoat()
{
    AlertDialog.Builder builder=new AlertDialog.Builder(this);
    builder.setTitle("hỏi thoát chương trình");
    builder.setMessage("Muốn thoát chương trình này hả?");
    builder.setNegativeButton("Không", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            dialog.cancel();
        }
    });
}

```

```

        }
    });
    builder.setPositiveButton("Có", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            finish();
        }
    });
    builder.create().show();
}
private class ProcessMyEvent implements OnClickListener
{
    @Override
    public void onClick(View arg0) {
        switch(arg0.getId())
        {
            case R.id.btntinhhtt: doTinhTien(); break;
            case R.id.btntiep: doTiep(); break;
            case R.id.btnthongke: doThongKe(); break;
            case R.id.imgbtnthoat: doThoat(); break;
        }
    }
}
}

```

- Trong MainActivity có tách các nghiệp vụ ra thành các hàm riêng biệt để bạn dễ dàng xử lý. Đây cũng là cách viết mã nguồn sạch sẽ, khi có lỗi xảy ra cũng giúp các bạn dễ Fixed bug, sai hàm nào thì sửa hàm đó mà nó không bị ảnh hưởng tới các nghiệp vụ khác.
- Bài tập tiếp theo bạn sẽ được học về các control nâng cao trong Android, và ta sẽ thực hành thật kỹ từng control cụ thể, đầu tiên là ListView (control rất thường xuyên được sử dụng trong một chương trình Android nào đó).