ButterKnife 一个优雅的注解工具

ButterKnife 注解

为什么要使用ButterKnife

- 避免重复写findViewById()来绑定View,使用代码块简洁明了
- 简单的点击，长按事件，不需要设置setOnclickListener()

如何使用ButterKnife

- Activity中的使用

```
class ExampleActivity extends Activity {
  @Bind(R.id.title) TextView title;
  @Bind(R.id.subtitle) TextView subtitle;
  @Bind(R.id.footer) TextView footer;

  @Override public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.simple_activity);
    ButterKnife.bind(this);
    // TODO Use fields...
  }
}
```

- Fragment中的使用

```
public class FancyFragment extends Fragment {
  @Bind(R.id.button1) Button button1;
  @Bind(R.id.button2) Button button2;

  @Override public View onCreateView(LayoutInflater inflater,
ViewGroup container, Bundle savedInstanceState) {
    View view = inflater.inflate(R.layout.fancy_fragment,
container, false);
    ButterKnife.bind(this, view);
    // TODO Use fields...
    return view;
  }
    @Override public void onDestroyView() {
    super.onDestroyView();
    ButterKnife.unbind(this);
  }
}
}
```

- Adapter中的使用

```
public class MyAdapter extends BaseAdapter {
  @Override public View getView(int position, View view, ViewGroup parent) {
    ViewHolder holder;
    if (view != null) {
      holder = (ViewHolder) view.getTag();
    } else {
      view = inflater.inflate(R.layout.whatever, parent, false);
      holder = new ViewHolder(view);
      view.setTag(holder);
    }

    holder.name.setText("John Doe");
    // etc...

    return view;
  }

  static class ViewHolder {
    @Bind(R.id.title) TextView name;
    @Bind(R.id.job_title) TextView jobTitle;

    public ViewHolder(View view) {
      ButterKnife.bind(this, view);
    }
  }
}
```

- 自定义View中的使用

```
public class FancyButton extends Button {
  @OnClick
  public void onClick() {
    // TODO do something!
  }
}
```

- 资源绑定RESOURCE BINDING

```
class ExampleActivity extends Activity {
  @BindString(R.string.title) String title;
  @BindDrawable(R.drawable.graphic) Drawable graphic;
  @BindColor(R.color.red) int red; // int or ColorStateList field
  @BindDimen(R.dimen.spacer) Float spacer; // int (for pixel size) or float (for exact value) field
  // ...
}
```

- 其他用法

```
/*
*VIEW LISTS
*You can group multiple views into a List or array.
*/
@Bind({ R.id.first_name, R.id.middle_name, R.id.last_name })
List< EditText > nameViews;
```

```
/*LISTENER BINDING*/

/*Listeners can also automatically be configured onto methods.*/

@OnClick(R.id.submit)
public void submit(View view) {
  // TODO submit data to server...
}
/*All arguments to the listener method are optional.*/

@OnClick(R.id.submit)
public void submit() {
  // TODO submit data to server...
}
/*Define a specific type and it will automatically be cast.*/

@OnClick(R.id.submit)
public void sayHi(Button button) {
  button.setText("Hello!");
}
/*Specify multiple IDs in a single binding for common event
handling.*/

@OnClick({ R.id.door1, R.id.door2, R.id.door3 })
public void pickDoor(DoorView door) {
  if (door.hasPrizeBehind()) {
    Toast.makeText(this, "You win!", LENGTH_SHORT).show();
  } else {
    Toast.makeText(this, "Try again", LENGTH_SHORT).show();
  }
}
```

实现原理

主要用到了Java中的注解。这样可以使代码结构化，更加简洁，不会导致代码杂乱难以阅读，同时注解还可以帮我们并避免年写赘余的部署杂乱文件
ButterKnife（apt）不同于其他注解框架（利用反射机制实现）,它就类似于一个简化代码的工具类。

Demo

请点击 ButterKnife
链接https://github.com/Morcal/sampleCode/tree/master/samplecode/ButterKnifeDemo