**Generics**

What problem does it solve. Before generics, how did we do it? How do we do it. Are there issues with generics, Runtime? how do we solve it.

**Activities and layouts**

What do I want to say here, what’s the thesis statement — What a simple application is made of, how does it start, what does it need. How do I do it. Then a demo. Another demo, this time create new Activity and create a new layout file.

**Event Handling**

This is Android Recipe book (pragmatic)

Most user interaction with an Android device is captured by the system and sent to a cor- responding callback method. For example, if the physical Back button is pressed, the onBackPressed() method is called.These events can be handled by extending the class and overriding the methods, called *event handlers*.

User interaction with View or ViewGroup objects can also support *event listeners*.These are methods that wait for the registered event and then trigger the system to send the event information to the corresponding callback method. For example, the setOnClickListener() event listener can be registered for a button and when it is pressed, the onClick() method is called.

Event listeners are the preferred method when available because they avoid the class extension overhead. Furthermore, an Activity implementing an event listener gets a call- back for *all* the layout objects it contains, allowing for more concise code. Both event lis- teners and event handlers are demonstrated in this chapter within the context of handling physical key press events and screen touch events.

Constraint layout

<http://wiresareobsolete.com/2016/07/constraintlayout-part-1/>

Layout editor

https://developer.android.com/studio/write/layout-editor