

## Exam-2018

1. Gson serialize/deserialize Json
  2. Picasso To download and display pictures
  3. Swipecards display a list of cards like Tinder, allows you to swipe left or right.
  5. Change the header of the app by a Toolbar
- 
1. Palette: Personalize the colour of your detail using the picture of the movie.
  2. TabLayout: to display above the first screen a list of day, and sort each movie per day based on the tab selected.

```
1      mHandler = new Handler()
2      {
3          public void handleMessage(Message msg)
4          {
5              switch(msg.what)
6              {
7                  case MSG_DOWN_FAIL:
8                      //mTipTv.setText("download fail");
9                      break;
10                 case MSG_DOWN_SUCCESS:
11                     SetColor((Bitmap)msg.obj);
12                     break;
13                 default:
14                     break;
15             }
16         };
17     };
18     new Thread(){
19         public void run(){
20             Message msg = new Message();
21             msg.what = MSG_DOWN_SUCCESS;
22             msg.obj = bitmap;
23             mHandler.sendMessage(msg);
24         }.start();
25     }
```

4. Resource require a context to get from your code.

# Service

- Background process (same as the app)
- Doesn't display anything
- Can work while your app is in foreground or in background
- Service has been mostly replaced by JobScheduler (background tasks)
- Still used for foreground service, for bluetooth app or music
- Needs to have a notification since Android 8.0

## What is a Content Provider

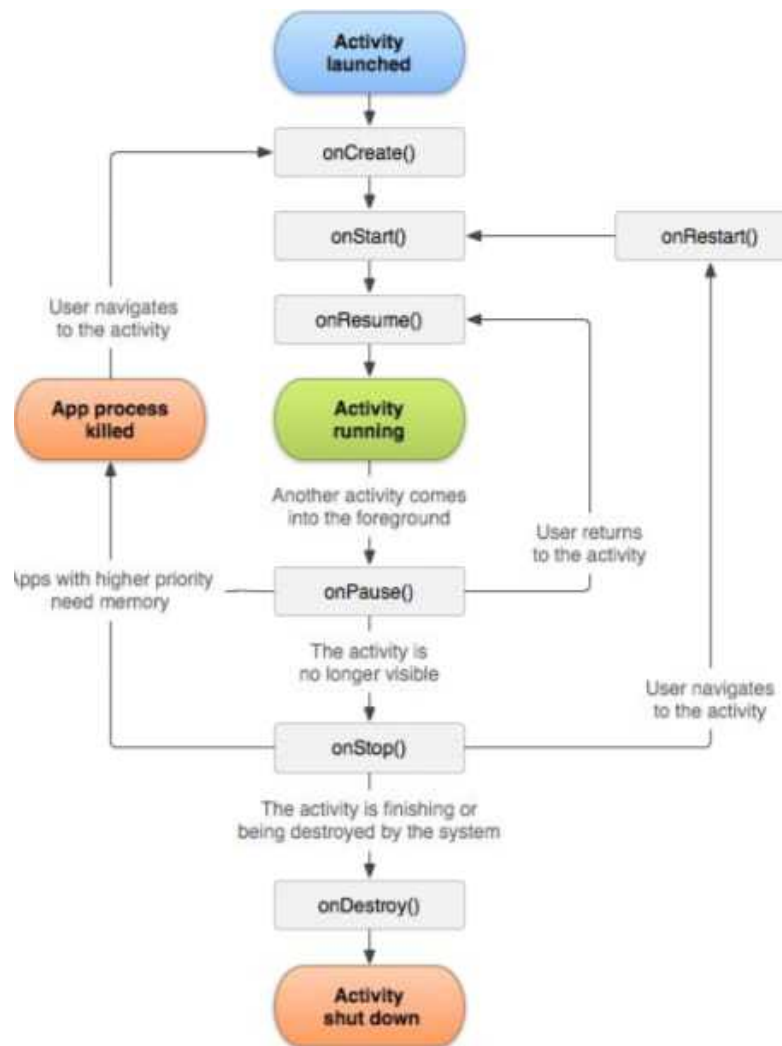
- A way of sharing data between apps
- Often linked to an SQLite storage
- The old way to store and access data in your own app

### Normal permissions

- Granted at install time
- INTERNET
- BLUETOOTH
- VIBRATE
- ACCESS\_NETWORK\_STATE
- ...

### Dangerous permissions

- Granted with explicit user agreement
- READ\_CALENDAR / WRITE\_CALENDAR
- READ\_CALL\_LOG / WRITE\_CALL\_LOG
- RECORD\_AUDIO
- CAMERA
- SEND\_SMS / READ\_SMS
- WRITE\_EXTERNAL\_STORAGE



A ViewGroup organizing its child View.

Every activity/fragment will use a layout file to describe its content.

It is stored in the layout folder of your resources, and wrote in XML

Types