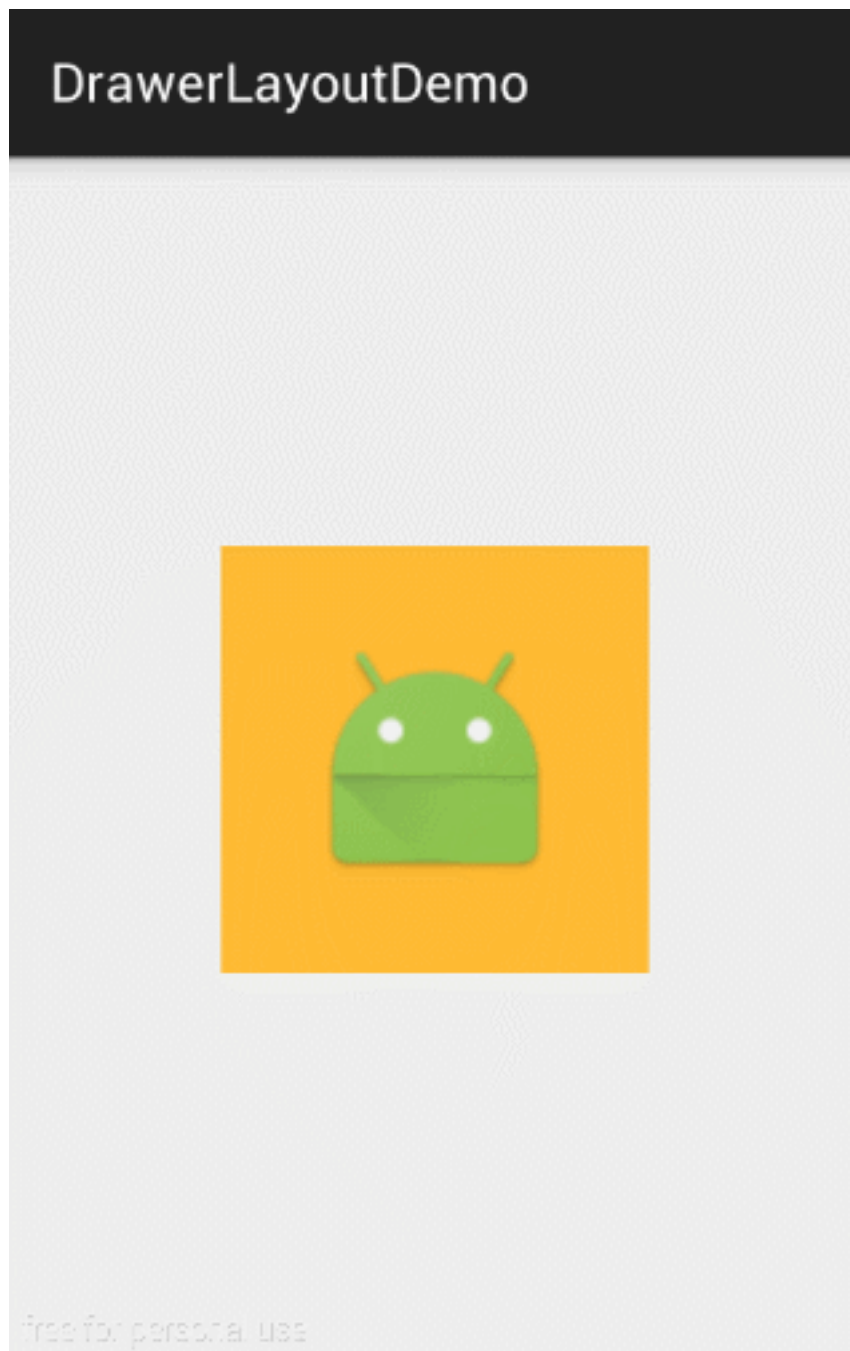


---

## TransitionPlayer

Android library to control Transition animates. A simple way to create a interactive animation.

### Demo1 SimpleTransition

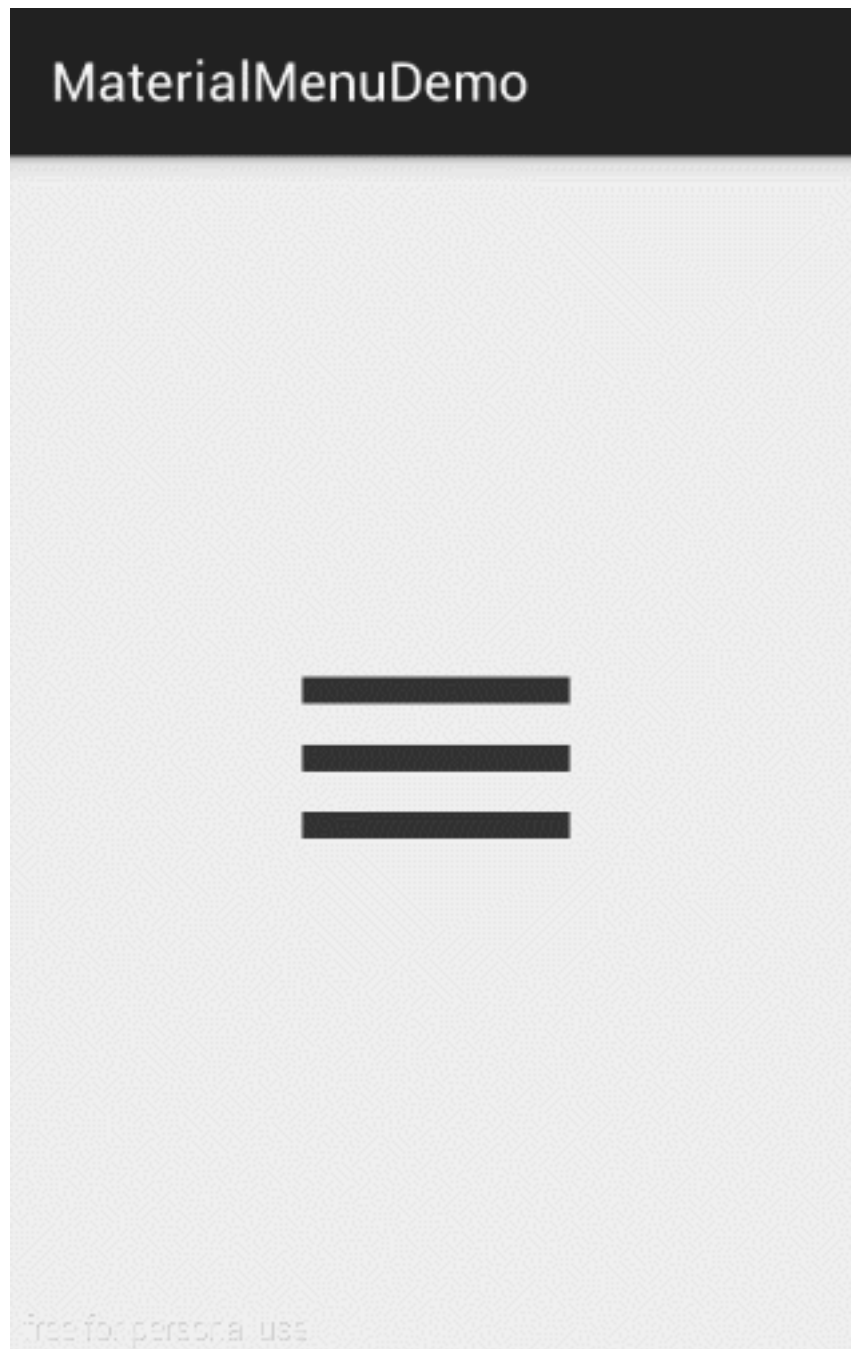


---

```
1      .....
2      TransitionManager.beginDelayedTransition(mSceneRoot,
          transitionPlayer);
3
4      imageView.setRotation(360);
5      imageView.setRotationX(270);
6      imageView.setRotationY(180);
7      imageView.setTranslationX(70 * getResources().getDisplayMetrics().
          density);
8      imageView.setScaleX(0.2f);
9      imageView.setScaleY(0f);
10     imageView.setBackgroundColor(getResources().getColor(android.R.
        color.holo_blue_bright));
11     .....
12     drawerLayout.setDrawerListener(new DrawerLayout.
        SimpleDrawerListener() {
13         public void onDrawerSlide(View drawerView, float slideOffset) {
14             transitionPlayer.setCurrentFraction(slideOffset);
15         }
16     });
```

---

## Demo2 MaterialMenu



### Code:

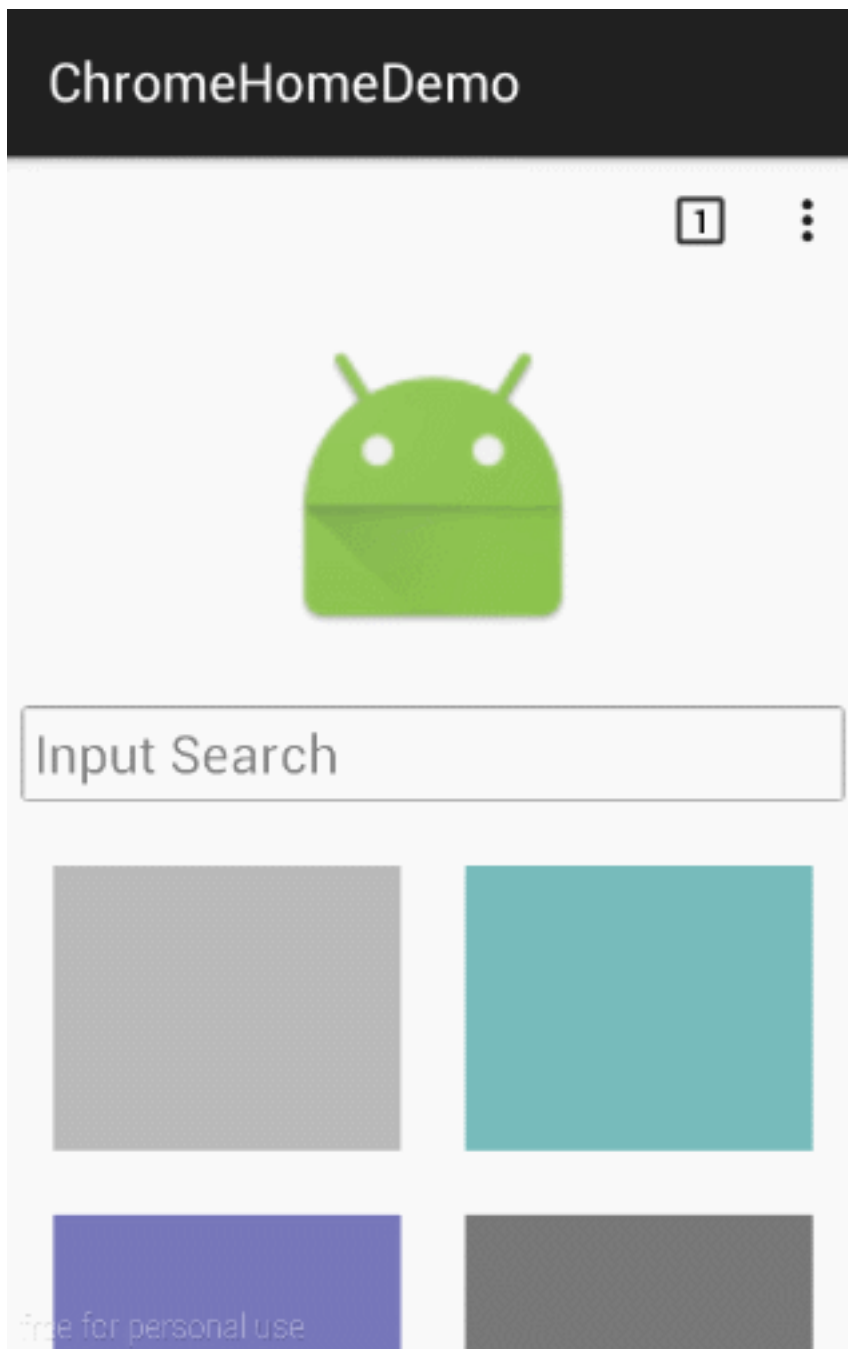
```
1      .....
2      int[] layouts = new int[]{R.layout.material_menu_option, R.layout.
      material_menu_back, R.layout.material_menu_y};
3      Scene scene = Scene.getSceneForLayout(mSceneRoot, layout[random.
      nextInt(layouts.length)], context);
4      TransitionManager.go(scene, transitionPlayer);
```

---

```
5      .....
6      drawerLayout.setDrawerListener(new DrawerLayout.
          SimpleDrawerListener() {
7          public void onDrawerSlide(View drawerView, float slideOffset) {
8              transitionPlayer.setCurrentFraction(slideOffset);
9          }
10     });
```

---

### Demo3 ChromeAppIndexPage



### Code: ChromeHome-

Demo.java

### License

MIT License.