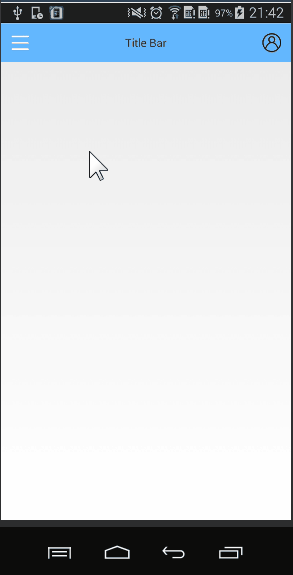
侧拉菜单在[**Android**](http://lib.csdn.net/base/android)应用中非常常见，它的实现方式太多了，今天我们就说说使用Google提供的DrawerLayout来实现侧拉菜单效果，先来看张效果图：



DrawerLayout的实现其实非常简单，只要按照既有的规范来写即可，先来看看布局文件：

**[java]** [view plain](http://blog.csdn.net/u012702547/article/details/49562747) [copy](http://blog.csdn.net/u012702547/article/details/49562747)

[print?](http://blog.csdn.net/u012702547/article/details/49562747)

1. <android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"
2. xmlns:tools="http://schemas.android.com/tools"
3. android:id="@+id/drawerlayout"
4. android:layout\_width="match\_parent"
5. android:layout\_height="match\_parent" >
7. <RelativeLayout
8. android:id="@+id/fragment\_layout"
9. android:layout\_width="match\_parent"
10. android:layout\_height="match\_parent" >
12. <RelativeLayout
13. android:id="@+id/title\_bar"
14. android:layout\_width="match\_parent"
15. android:layout\_height="48dp"
16. android:background="#63B8FF" >
18. <ImageView
19. android:id="@+id/leftmenu"
20. android:layout\_width="48dp"
21. android:layout\_height="48dp"
22. android:padding="12dp"
23. android:src="@drawable/menu" />
25. <TextView
26. android:layout\_width="wrap\_content"
27. android:layout\_height="48dp"
28. android:layout\_centerInParent="true"
29. android:gravity="center"
30. android:text="Title Bar" />
32. <ImageView
33. android:id="@+id/rightmenu"
34. android:layout\_width="48dp"
35. android:layout\_height="48dp"
36. android:layout\_alignParentRight="true"
37. android:padding="12dp"
38. android:src="@drawable/p\_center" />
39. </RelativeLayout>
41. <LinearLayout
42. android:id="@+id/content"
43. android:layout\_width="match\_parent"
44. android:layout\_height="match\_parent"
45. android:layout\_below="@id/title\_bar"
46. android:orientation="vertical" />
47. </RelativeLayout>
49. <RelativeLayout
50. android:id="@+id/left"
51. android:layout\_width="200dp"
52. android:layout\_height="match\_parent"
53. android:layout\_gravity="left"
54. android:background="@android:color/white" >
56. <ListView
57. android:id="@+id/left\_listview"
58. android:layout\_width="match\_parent"
59. android:layout\_height="match\_parent" >
60. </ListView>
61. </RelativeLayout>
63. <RelativeLayout
64. android:id="@+id/right"
65. android:layout\_width="260dp"
66. android:layout\_height="match\_parent"
67. android:layout\_gravity="right"
68. android:background="#BCEE68" >
70. <ImageView
71. android:id="@+id/p\_pic"
72. android:layout\_width="72dp"
73. android:layout\_height="72dp"
74. android:layout\_centerInParent="true"
75. android:src="@drawable/p\_pic" />
77. <TextView
78. android:id="@+id/right\_textview"
79. android:layout\_width="wrap\_content"
80. android:layout\_height="48dp"
81. android:layout\_below="@id/p\_pic"
82. android:layout\_centerHorizontal="true"
83. android:layout\_marginTop="12dp"
84. android:text="个人中心"
85. android:textColor="@android:color/black"
86. android:textSize="14sp" />
87. </RelativeLayout>
89. </android.support.v4.widget.DrawerLayout>

首先，根布局就是DrawerLayout，在根布局之后又主要分为三大块，第一块就是我们主界面的内容，第二块是左边拉出来的布局，第三块是右边拉出来的布局（不需要右边侧拉就不用写，这样的话整个布局就只分为两大块），那么系统怎么知道我们这个布局是主布局还是侧拉菜单的布局？请注意左边侧拉菜单布局android:layout\_gravity="left"这个属性和右边菜单布局的android:layout\_gravity="right"这个属性，哈哈，这下应该明白了吧，系统通过layout\_gravity属性的值来判断这个布局是左边菜单的布局还是右边菜单的布局，如果没有这个属性，那不用说，肯定是主界面的布局。

好了，布局文件写好之后，我们的侧拉效果其实就已经可以实现了，但是你发现左边拉出来什么东西都没有，那是因为还没有数据，所以我们要在MainActivity中初始化左边布局的ListView，给ListView赋值这个想必大家都会，我就直接贴代码了：

**[java]** [view plain](http://blog.csdn.net/u012702547/article/details/49562747) [copy](http://blog.csdn.net/u012702547/article/details/49562747)

[print?](http://blog.csdn.net/u012702547/article/details/49562747)

1. listView = (ListView) findViewById(R.id.left\_listview);
2. initData();
3. adapter = **new** ContentAdapter(**this**, list);
4. listView.setAdapter(adapter);

初始化数据的方法：

**[java]** [view plain](http://blog.csdn.net/u012702547/article/details/49562747) [copy](http://blog.csdn.net/u012702547/article/details/49562747)

[print?](http://blog.csdn.net/u012702547/article/details/49562747)

1. **private** **void** initData() {
2. list = **new** ArrayList<ContentModel>();
4. list.add(**new** ContentModel(R.drawable.doctoradvice2, "新闻", 1));
5. list.add(**new** ContentModel(R.drawable.infusion\_selected, "订阅", 2));
6. list.add(**new** ContentModel(R.drawable.mypatient\_selected, "图片", 3));
7. list.add(**new** ContentModel(R.drawable.mywork\_selected, "视频", 4));
8. list.add(**new** ContentModel(R.drawable.nursingcareplan2, "跟帖", 5));
9. list.add(**new** ContentModel(R.drawable.personal\_selected, "投票", 6));
10. }

ContentModel类：

**[java]** [view plain](http://blog.csdn.net/u012702547/article/details/49562747) [copy](http://blog.csdn.net/u012702547/article/details/49562747)

[print?](http://blog.csdn.net/u012702547/article/details/49562747)

1. **public** **class** ContentModel {
3. **private** **int** imageView;
4. **private** String text;
5. **private** **int** id;

8. **public** **int** getId() {
9. **return** id;
10. }
12. **public** **void** setId(**int** id) {
13. **this**.id = id;
14. }
16. **public** ContentModel() {
17. }
19. **public** ContentModel(**int** imageView, String text, **int** id) {
20. **this**.imageView = imageView;
21. **this**.text = text;
22. **this**.id = id;
23. }
25. **public** **int** getImageView() {
26. **return** imageView;
27. }
29. **public** **void** setImageView(**int** imageView) {
30. **this**.imageView = imageView;
31. }
33. **public** String getText() {
34. **return** text;
35. }
37. **public** **void** setText(String text) {
38. **this**.text = text;
39. }
41. }

数据适配器：

**[java]** [view plain](http://blog.csdn.net/u012702547/article/details/49562747) [copy](http://blog.csdn.net/u012702547/article/details/49562747)

[print?](http://blog.csdn.net/u012702547/article/details/49562747)

1. **public** **class** ContentAdapter **extends** BaseAdapter {
3. **private** Context context;
4. **private** List<ContentModel> list;
6. **public** ContentAdapter(Context context, List<ContentModel> list) {
7. **super**();
8. **this**.context = context;
9. **this**.list = list;
10. }
12. @Override
13. **public** **int** getCount() {
14. **if** (list != **null**) {
15. **return** list.size();
16. }
17. **return** 0;
18. }
20. @Override
21. **public** Object getItem(**int** position) {
22. **if** (list != **null**) {
23. **return** list.get(position);
24. }
25. **return** **null**;
26. }
28. @Override
29. **public** **long** getItemId(**int** position) {
30. **return** list.get(position).getId();
31. }
33. @Override
34. **public** View getView(**int** position, View convertView, ViewGroup parent) {
35. ViewHold hold;
36. **if** (convertView == **null**) {
37. hold = **new** ViewHold();
38. convertView = LayoutInflater.from(context).inflate(
39. R.layout.content\_item, **null**);
40. convertView.setTag(hold);
41. } **else** {
42. hold = (ViewHold) convertView.getTag();
43. }
45. hold.imageView = (ImageView) convertView
46. .findViewById(R.id.item\_imageview);
47. hold.textView = (TextView) convertView.findViewById(R.id.item\_textview);
49. hold.imageView.setImageResource(list.get(position).getImageView());
50. hold.textView.setText(list.get(position).getText());
51. **return** convertView;
52. }
54. **class** ViewHold {
55. **public** ImageView imageView;
56. **public** TextView textView;
57. }
59. }

每个Item的布局文件：

**[java]** [view plain](http://blog.csdn.net/u012702547/article/details/49562747) [copy](http://blog.csdn.net/u012702547/article/details/49562747)

[print?](http://blog.csdn.net/u012702547/article/details/49562747)

1. <?xml version="1.0" encoding="utf-8"?>
2. <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3. android:layout\_width="match\_parent"
4. android:layout\_height="match\_parent"
5. android:orientation="horizontal"
6. android:paddingTop="20dp"
7. android:paddingBottom="20dp"
8. android:gravity="center">
10. <ImageView
11. android:id="@+id/item\_imageview"
12. android:layout\_width="wrap\_content"
13. android:layout\_height="wrap\_content"
14. android:src="@drawable/ic\_launcher"/>
16. <TextView
17. android:id="@+id/item\_textview"
18. android:layout\_width="wrap\_content"
19. android:layout\_height="wrap\_content"
20. android:text="aa"
21. android:textSize="20dp"/>
23. </LinearLayout>

好了，给左边的侧拉菜单中的ListView赋值之后，在把它拉出来，这下就全都有数据了，但是我除了想通过滑动让侧拉菜单出来之外，我还希望在App的标题栏上有一个按钮，点击之后左边的侧拉菜单也会出来，这要怎么实现？看代码：

先将原有的标题栏隐藏：

**[java]** [view plain](http://blog.csdn.net/u012702547/article/details/49562747) [copy](http://blog.csdn.net/u012702547/article/details/49562747)

[print?](http://blog.csdn.net/u012702547/article/details/49562747)

1. getActionBar().hide();

然后：

**[java]** [view plain](http://blog.csdn.net/u012702547/article/details/49562747) [copy](http://blog.csdn.net/u012702547/article/details/49562747)

[print?](http://blog.csdn.net/u012702547/article/details/49562747)

1. leftMenu = (ImageView) findViewById(R.id.leftmenu);
2. leftMenu.setOnClickListener(**new** OnClickListener() {
4. @Override
5. **public** **void** onClick(View v) {
6. drawerLayout.openDrawer(Gravity.LEFT);
7. }
8. });

其实很简单，直接调用DrawerLayout的openDrawer方法，参数传Gravity.LEFT表示让左边的侧拉菜单出来，参数如果传Gravity.RIGHT，则表示让右边的侧拉菜单出来。好了，当左边的侧拉菜单出来之后，我希望点击菜单的每一个item，主界面都会有所反应，即当我点击“新闻”，主界面显示新闻内容，当我点击”订阅“，主界面显示订阅的内容，这个也很好实现，首先，点击事件不用说，就是ListView的setOnItemClickListener，点击之后，我们的主界面会显示相应的Fragment，即，如果点击了新闻，则注解卖弄显示新闻的Fragment，如果点击了订阅，则主界面显示订阅的Fragment，看代码：

**[java]** [view plain](http://blog.csdn.net/u012702547/article/details/49562747) [copy](http://blog.csdn.net/u012702547/article/details/49562747)

[print?](http://blog.csdn.net/u012702547/article/details/49562747)

1. listView.setOnItemClickListener(**new** OnItemClickListener() {
3. @Override
4. **public** **void** onItemClick(AdapterView<?> parent, View view,
5. **int** position, **long** id) {
6. FragmentTransaction bt = fm.beginTransaction();
7. **switch** ((**int**) id) {
8. **case** 1:
9. bt.replace(R.id.content, **new** NewsFragment());
10. **break**;
11. **case** 2:
12. bt.replace(R.id.content, **new** SubscriptionFragment());
13. **break**;
15. **default**:
16. **break**;
17. }
18. bt.commit();
19. drawerLayout.closeDrawer(Gravity.LEFT);
20. }
21. });

每次点击之后，就用相应的Fragment替换主界面的LinearLayout，当然，替换完成之后要记得关闭左边的侧拉菜单，传入的参数为Gravity.LEFT表示关闭左边的侧拉菜单，如果传入的菜单为Gravity.RIGHT表示关闭右边的侧拉菜单。这里两个Fragment都很简单，我就不贴源码了，大家一会直接下载Demo看