

# FragmentTabHost

android.support.v4.app.FragmentTabHost 是V4包里的控件

在布局中:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <!--内容区域-->

    <FrameLayout
        android:id="@+id/fl_content"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1">

    </FrameLayout>

    <!--底部菜单区域-->

    <android.support.v4.app.FragmentTabHost
        android:id="@+id/tabhost"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
</LinearLayout>
```

具体的使用方法:

```
FragmentTabHost mTabhost = findViewById(R.id.tabhost);
//mTabhost.setup(this, getSupportFragmentManager()Fragmen管理器, FragmentTabHost对应内容的容器id);
//FragmentTabHost初始化方法
mTabhost.setup(this, getSupportFragmentManager(), R.id.fl_content);

mTabHost.setOnTabChangeListener(this); 设置TabHost的选择监听

//去掉FragmentTabHost的分割线 去除分割线
if (android.os.Build.VERSION.SDK_INT > 10) {
    mTabhost.getTabWidget().setShowDividers(0);
}

List<MainTabInfo> tabInfos = new ArrayList<>();
tabInfos.add(new MainTabInfo("首页", R.drawable.tab_icon_home, HomeFragment.class));
tabInfos.add(new MainTabInfo("购物中心", R.drawable.tab_icon_product_center, ProductCenterFragment.class));
tabInfos.add(new MainTabInfo("购物车", R.drawable.tab_icon_shopping_cart, ShoppingCartFragment.class));
tabInfos.add(new MainTabInfo("我", R.drawable.tab_icon_myself, MySelfFragment.class));
for (int i = 0; i < tabInfos.size(); i++) {
    MainTabInfo mainTabInfo = tabInfos.get(i);
```

```

String titles = mainTabInfo.title;
int drawResID = mainTabInfo.drawResID;
Class clz = mainTabInfo.clz;

//添加tab标签页
TabHost.TabSpec tabSpec = mTabHost.newTabSpec("tatile");

//找孩子
View indicatorView = View.inflate(this, R.layout.inflate_main_indicatorview, null);
TextView title = (TextView) indicatorView.findViewById(R.id.tab_title);

//赋值
title.setText("首页");

//设置drawableTop 设置了图片选择器
title.setCompoundDrawablesWithIntrinsicBounds(0, R.drawable.tab_icon_new, 0, 0);

//设置Tab标题 这个地方可以接收的参数（一、字符串。二、字符串和Drawable。三、View）
tabSpec.setIndicator(indicatorView);

//设置参数
Bundle args1 = new Bundle();
args1.putString("args", "我是参数");
mTabHost.addTab(tabSpec, HomeFragment.class, args1);

mTabHost.setTag(i);
}

```

重写一个选择监听的方法： 与ViewPager关联的核心方法

如果要关联类还需要实现**implements TabHost.OnTabChangeListener**

```

@Override
public void onTabChanged(String tabId) {
    int position = mTabHost.getCurrentTab();
    mViewPager.setCurrentItem(position);
}

```

关于**inflate\_main\_indicatorview** 这个布局

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content">

    <TextView
        android:id="@+id/tab_title"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:layout_gravity="center"
        android:drawableTop="@drawable/tab_icon_new" 使用了图片选择器
        android:gravity="center"
        android:text="资讯"
    >

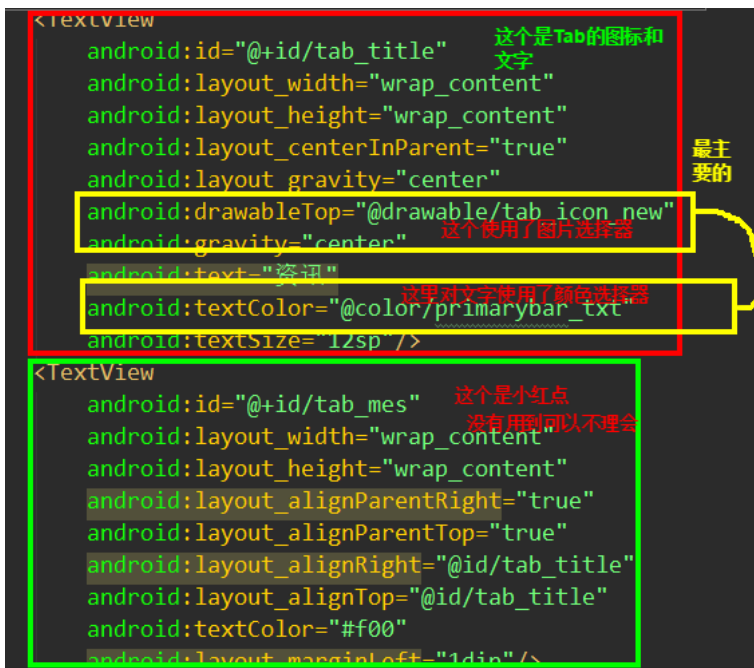
```

```
android:textColor="@color/primarybar_txt"    使用了字体颜色选择器
android:textSize="12sp"/>
```

```
<TextView
    android:id="@+id/tab_mes"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentRight="true"
    android:layout_alignParentTop="true"
    android:layout_alignRight="@id/tab_title"
    android:layout_alignTop="@id/tab_title"
    android:textColor="#f00"
    android:layout_marginLeft="1dip"/>
```

</RelativeLayout>

相关说明:



图片选择器说明:

在drawable目录下的tab\_icon\_new.xml

```
<?xml version="1.0" encoding="utf-8"?>
<selector
    xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:state_pressed="true" android:drawable="@drawable/widget_bar_news_over" />
    <item android:state_checked="true" android:drawable="@drawable/widget_bar_news_over" />
    <item android:state_selected="true" android:drawable="@drawable/widget_bar_news_over" />
    <item android:drawable="@drawable/widget_bar_news_nor" />
</selector>
```

颜色选择器说明: 字体颜色选择器

在res下的新建目录color在color目录下的primarybar\_txt.xml

```
<?xml version="1.0" encoding="utf-8"?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
```

```

        <item android:state_pressed="true" android:color="#0b9a27"/>
        <item android:state_selected="true" android:color="#0b9a27"/>
        <item android:color="#666"/>

</selector>

```

通过资源数组循环添加TAB标签按钮一：适用于只有一个Fragment

```

String[] titleArr = {"综合", "动弹", "", "发现", "我"};
int[] topResIdArr = {R.drawable.tab_icon_new, R.drawable.tab_icon_tweet, R.drawable.tab_icon_tweet,
R.drawable.tab_icon_explore, R.drawable.tab_icon_me};

for (int i = 0; i < titleArr.length; i++) {
    String titles = titleArr[i];
    int drawResID = drawResArr[i];
    //添加tab标签页
    TabHost.TabSpec tabSpec = mTabHost.newTabSpec(titles);
    //找孩子
    View idicatorView = View.inflate(this, R.layout.inflate_main_indicatorview, null);
    TextView title = (TextView) idicatorView.findViewById(R.id.tab_title);
    //赋值
    title.setText(titles);
    //设置drawableTop
    title.setCompoundDrawablesWithIntrinsicBounds(0, drawResID, 0, 0);
    tabSpec.setIndicator(idicatorView);    //设置Tab标题
    //设置参数
    Bundle args1 = new Bundle();
    args1.putString("args", "我是参数");
    mTabHost.addTab(tabSpec, HomeFragment.class, args1);
}

```

通过集合资源实现 方法二：

创建资源Bean：

```

public class MainTabInfo {
    public String title;
    public int drawResID;
    public Class clz;
    public MainTabInfo(String title, int drawResID, Class zls) {
        this.title = title;
        this.drawResID = drawResID;
        this.clz = zls;
    }
}

```

创建资源集合：

```

List<MainTabInfo> tabInfos = new ArrayList<>();
tabInfos.add(new MainTabInfo("首页", R.drawable.tab_icon_home, HomeFragment.class));

```

```

tabInfos.add(new MainTabInfo("购物中心", R.drawable.tab_icon_product_center, ProductCenterFragment
tabInfos.add(new MainTabInfo("购物车", R.drawable.tab_icon_shopping_cart, ShoppingCartFragment.class));
tabInfos.add(new MainTabInfo("我", R.drawable.tab_icon_myself, MySelfFragment.class));

for (int i = 0; i < tabInfos.size(); i++) {
    MainTabInfo mainTabInfo = tabInfos.get(i);
    String titles = mainTabInfo.title;
    int drawResID = mainTabInfo.drawResID;
    Class clz = mainTabInfo.clz;
    //添加tab标签页
    TabHost.TabSpec tabSpec = mTabHost.newTabSpec(titles);
    //找孩子
    View indicatorView = View.inflate(this, R.layout.inflate_main_indicatorview, null);
    TextView title = (TextView) indicatorView.findViewById(R.id.tab_title);
    //赋值
    title.setText(titles);
    //设置drawableTop
    title.setCompoundDrawablesWithIntrinsicBounds(0, drawResID, 0, 0);
    tabSpec.setIndicator(indicatorView);    //设置Tab标题
    //设置参数
    Bundle args1 = new Bundle();
    args1.putString("args", "我是参数");
    mTabHost.addTab(tabSpec, clz, args1);
}

```

通过枚举来创建资源实现 方法三:

```

enum MAINTAB{
    //对枚举赋值
    HOME("首页", R.drawable.tab_icon_home, HomeFragment.class),
    CENTER("购物中心", R.drawable.tab_icon_product_center, ProductCenterFragment.class),
    SHOPING("购物车", R.drawable.tab_icon_shopping_cart, ShoppingCartFragment.class),
    MYSELF("我", R.drawable.tab_icon_myself, MySelfFragment.class);
    public String title;
    public int drawResID;
    public Class clz;
    //创建构造方法
    MAINTAB(String title, int drawResID, Class zls) {
        this.title = title;
        this.drawResID = drawResID;
        this.clz = zls;
    }
}

for (int i = 0; i < MAINTAB.values().length; i++) {
    MAINTAB maintab = MAINTAB.values()[i];
    String titles = maintab.title;
    int drawResID = maintab.drawResID;
}

```

```

Class clz = maintab.clz;
//添加tab标签页
TabHost.TabSpec tabSpec = mTabHost.newTabSpec(titles);
//找孩子
View indicatorView = View.inflate(this,R.layout.inflate_main_indicatorview, null);
TextView title = (TextView) indicatorView.findViewById(R.id.tab_title);
//赋值
title.setText(titles);
//设置drawableTop
title.setCompoundDrawablesWithIntrinsicBounds(0, drawResID, 0, 0);
tabSpec.setIndicator(indicatorView); //设置Tab标题
//设置参数
Bundle args1 = new Bundle();
args1.putString("args", "我是参数");
mTabHost.addTab(tabSpec, clz, args1);
}

```

```

private void initFragmentTabHost() {
    mTabHost.setup(this, getSupportFragmentManager(), R.id.contentframe);
    //方法二：
    List<MainTabInfo> tabInfos = new ArrayList<>();
    tabInfos.add(new MainTabInfo("首页",R.drawable.tab_icon_home,HomeFragment.class));
    tabInfos.add(new MainTabInfo("购物中心",R.drawable.tab_icon_product_center,Product
    tabInfos.add(new MainTabInfo("购物车",R.drawable.tab_icon_shopping_cart,ShoppingCa
    tabInfos.add(new MainTabInfo("我",R.drawable.tab_icon_myself,MySelfFragment.class));
    for (int i = 0; i < tabInfos.size(); i++) {
        MainTabInfo mainTabInfo = tabInfos.get(i);
        String titles = mainTabInfo.title;
        int drawResID = mainTabInfo.drawResID;
        Class clz = mainTabInfo.clz;
        //添加tab标签页
        TabHost.TabSpec tabSpec = mTabHost.newTabSpec(titles);
        //找孩子
        View indicatorView = View.inflate(this,R.layout.inflate_main_indicatorview, null);
        TextView title = (TextView) indicatorView.findViewById(R.id.tab_title);
        //赋值
        title.setText(titles);
        //设置drawableTop
        title.setCompoundDrawablesWithIntrinsicBounds(0, drawResID, 0, 0);
        tabSpec.setIndicator(indicatorView); //设置Tab标题
    }
}

```

方法三 主要也是改这一块的东西

FragmentTabHost获取子控件并设置某个子控件的状态等事件的方法：

**方法一：**通过FragmentTabHost对象获取某一个子Tab对象

```
View childTabViewAt = mTabHost.getTabWidget().getChildTabViewAt(2);
```

**方法二：**通过FragmentTabHost对象获取某一个子Tab对象

```
View childAt = mTabHost.getTabWidget().getChildAt(2);
ImageView viewById = (ImageView) childAt.findViewById(R.id.tab_mes);
```

默认点击Tab的事件是进入对应的Fragment，重写Tab的点击事件后就无法进入Fragment  
此时要设置一个让`mTabHost.setCurrentTab(2)`；选择进入某一个Tab的Fragment

```
View childTabViewAt = mTabHost.getTabWidget().getChildTabViewAt(2);  
final ImageView viewById = (ImageView) childTabViewAt.findViewById(R.id.tab_mes);  
childTabViewAt.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        if (viewById.getVisibility() == View.VISIBLE) {  
            viewById.setVisibility(View.GONE);  
        }  
        //点击事件的处理  
        mTabHost.setCurrentTab(2);  
    }  
});
```

[MainActivity.java](#)

[activity\\_main.xml](#)

[inflate main indicatorview.xml](#)