

viewPager 通过滑动跳转主界面

1801210657 王鲁月

本文在实现 ViewPager 引导页面制作完成的基础上, 加入滑动跳转主界面的功能
即在滑动到最后一页时继续向右滑动, 即可跳转至主界面

一、在引导 Activity 组建中加入页面滑动监听接口

找到创建的引导 Activity 组建, 在类名后面加上处理滑动事件的 implements
ViewPager.OnPageChangeListener 接口:

```
public class GuideActivity extends Activity implements ViewPager.OnPageChangeListener {  
    private ViewPager viewPager;  
    private List<View> views;  
    private ImageView[] dots;  
    private int[] ids = {R.id.iv1, R.id.iv2};
```

光标选中 ViewPager.OnPageChangeListener, 按 ALT+Enter 自动生成三个缺省的方法:

```
@Override  
public void onPageScrolled(int i, float v, int i1) {  
  
}  
  
@Override  
public void onPageSelected(int i) {  
  
}  
  
@Override  
public void onPageScrollStateChanged(int i) {  
  
}
```

二、重写 onPageScrollStateChanged 方法

onPageScrollStateChanged 方法是在页面状态改变的时候调用，其中参数 i 有三种状态：

SCROLL_STATE_DRAGGING (int 值为 1)：

表示用户手指“按在屏幕上并且开始拖动”的状态

SCROLL_STATE_IDLE (int 值为 0)：

滑动动画做完的状态。

SCROLL_STATE_SETTLING (int 值为 2)：

在“手指离开屏幕”的状态。

一个完整的滑动动作，三种状态的出发顺序为 (1, 2, 0)

因此，只需设置一个标志位判断 ViewPager 是否发生了滑动操作，且该操作页面为最后一个页面，即可实现跳转主界面功能。

首先，需定义一个标志位 flag 监测页面滑动：

```
private ViewPager pageView;  
private List<View> views;  
private GuideAdapter guideAdapter;  
private ImageView[] dots;  
private int[] ids = {R.id.iv1, R.id.iv2};  
private Button btn;  
private boolean flag;
```

当页面发生拖动时，将 flag 赋值为 false，记录发生页面滑动事件

```

@Override
public void onPageScrollStateChanged(int i) {
    switch (i) {
        case ViewPager.SCROLL_STATE_DRAGGING:
            flag = false;
            break;
        case ViewPager.SCROLL_STATE_SETTLING:
            flag = true;
            break;
        case ViewPager.SCROLL_STATE_IDLE:
            flag = true;
            break;
    }
}

```

在 case ViewPager.SCROLL_STATE_IDLE:状态下，编写跳转主界面条件（当前页面为最后一页，且发生滑动）

```

case ViewPager.SCROLL_STATE_IDLE:
    if (pageView.getCurrentItem() == pageView.getAdapter().getCount() - 1 && !flag) {
        Intent ss = new Intent( packageContext: GuideActivity.this, MainActivity.class);
        startActivity(ss);
        finish();
    }
    flag = true;
    break;

```

三、GuideActivity 类源码

包含动态圆点显示，滑动跳转，按键跳转

```
public class GuideActivity extends Activity implements ViewPager.OnPageChangeListener {

    private ViewPager pageView;

    private List<View> views;

    private GuideAdapter guideAdapter;

    private ImageView[] dots;

    private int[] ids = {R.id.iv1, R.id.iv2};

    private Button btn;

    private boolean flag;

    void initDots() {

        dots = new ImageView[views.size()];

        for (int i = 0; i < views.size(); i++) {

            dots[i] = (ImageView) findViewById(ids[i]);

        }

    }

    private void initViews() {

        LayoutInflater inflater = LayoutInflater.from(this);

        views = new ArrayList<View>();

        views.add(LayoutInflater.from(this).inflate(R.layout.page1_guide, null));

        views.add(LayoutInflater.from(this).inflate(R.layout.page2_guide, null));

        guideAdapter = new GuideAdapter(this, views);

        pageView = (ViewPager) findViewById(R.id.guidePage);

        pageView.setAdapter(guideAdapter);

        pageView.setOnPageChangeListener(this);

    }

}
```

```

@Override

protected void onCreate(Bundle savedInstanceState) {

    //TODO 自动生成方法存根

    super.onCreate(savedInstanceState);

    setContentView(R.layout.guide_layout);

    initView();

    initDots();

    btn = (Button) views.get(1).findViewById(R.id.btn);

    btn.setOnClickListener(new View.OnClickListener() {

        @Override

        public void onClick(View view) {

            Intent i = new Intent(GuideActivity.this, MainActivity.class);

            startActivity(i);

            finish();

        }

    });
}

```

```

@Override

public void onPageScrolled(int i, float v, int i1) {

}

```

```

@Override

public void onPageSelected(int i) {

    for (int a = 0; a < ids.length; a++) {

        if (a == i)

            dots[a].setImageResource(R.drawable.page_indicator_focused);

        else

```

```

        dots[a].setImageResource(R.drawable.page_indicator_unfocused);
    }
}

@Override

public void onPageScrollStateChanged(int i) {
    switch (i) {
        case ViewPager.SCROLL_STATE_DRAGGING:
            flag = false;
            break;
        case ViewPager.SCROLL_STATE_SETTLING:
            flag = true;
            break;
        case ViewPager.SCROLL_STATE_IDLE:
            if (pageView.getCurrentItem() == pageView.getAdapter().getCount() - 1
&& !flag) {
                Intent ss = new Intent(GuideActivity.this, MainActivity.class);
                startActivity(ss);
                finish();
            }
            flag = true;
            break;
    }
}
}

```

//ViewPagerhua

@Override

```

public void onPageScrollStateChanged(int i) {
    switch (i) {

```

```

case ViewPager.SCROLL_STATE_DRAGGING:

    flag = false;

    break;

case ViewPager.SCROLL_STATE_SETTLING:

    flag = true;

    break;

case ViewPager.SCROLL_STATE_IDLE:

    if (pageView.getCurrentItem() == pageView.getAdapter().getCount() - 1

&& !flag) {

        Intent ss = new Intent(GuideActivity.this, MainActivity.class);

        startActivity(ss);

        finish();

    }

    flag = true;

    break;

    }

}

}

```