

第3章 多个用户界面的 程序设计

2025年4月

3.1 页面切换与传递参数值

3.1.1 传递参数组件Intent

- **Intent**是android系统一种运行时的绑定机制，在应用程序运行时连接两个不同组件。在**android** 的应用程序中不管是页面切换，还是传递数据，或是调用外部程序，都可能要用到**Intent**。**Intent**负责对应用中某次操作的动作、动作涉及数据、附加数据进行描述，**Android**则根据此**Intent**的描述，负责找到对应的组件，将 **Intent**传递给调用的组件，并完成组件的调用。

3.1.2 Activity页面切换

Activity跳转与传递参数值主要通过Intent类协助实现。在一个Activity页面中启动另一个Activity页面的运行，是最简单的Activity页面切换方式。其步骤如下：

(1) 首先创建一个Intent对象，其构造方法为：

```
Intent intent = new Intent(当前Activity.this, 另一Activity.class);
```

(2) 然后调用Activity的startActivity(intent)方法，切换到另一个Activity页面。

【例3-1】 从一个Activity页面启动另一个Activity页面示例。

- 在本项目中，要建立两个页面文件及两个控制文件：
 - 第一个页面的界面布局文件为activity_main.xml，控制文件为MainActivity.java;
 - 第二个页面的界面布局文件为second.xml，控制文件为secondActivity.java。
- 还要修改配置文件AndroidManifest.xml。



3.1.3 应用Intent在Activity页面之间传递数据

1、Bundle类

- Bundle类是一个用于将字符串与某组件对象建立映射关系的组件。Bundle组件与Intent配合使用，可在不同的Activity之间传递数据。Bundle类的常用方法如下：
- putString(String key, String value): 把字符串用“键－值”形式存放到Bundle对象中；
 - remove(String key): 移除指定key的值；
 - getString(String key): 获取指定key的字符。

2、应用Intent在不同的Activity之间传递数据

(1) 在页面Activity A端

- 创建Intent对象和Bundle对象

```
Intent intent = new Intent();
```

```
Bundle bundle = new Bundle();
```

- 为Intent指定切换页面，用Bundle 存放"键 – 值"对数据

```
intent.setClass(MainActivity.this, secondActivity.class);
```

```
bundle.putString("text", txt.getText().toString());
```

- 将Bundle对象传递给Intent

```
intent.putExtras(bundle);
```

(2) 在另一页面Activity B端

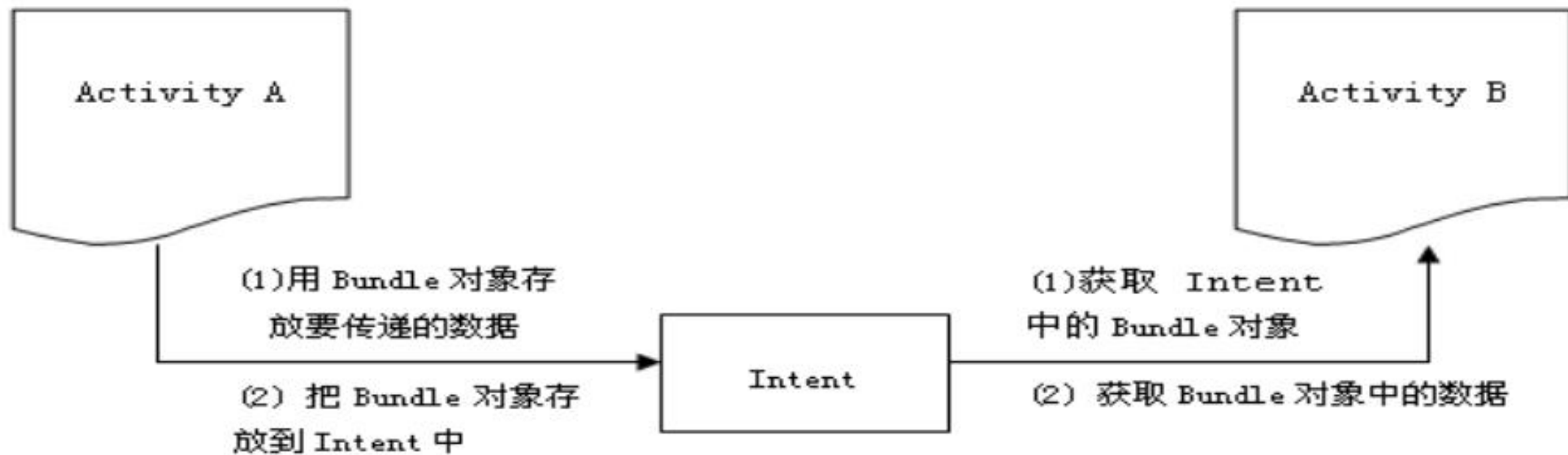
- 从Intent中获取Bundle对象

```
bunde = this getIntent().getExtras();
```

- 从Bundle对象中按"键 - 值"对的键名获取对应数据值

```
String str = bunde.getString("text");
```


图3.3 应用Intent在Activity页面之间传递数据



【例3-2】从第一个Activity页面传递数据到第二个Activity页面示例。

- (代码见教材)



3.2 菜单Menu

- 一个菜单 (**Menu**) 由多个菜单选项组成, 选择一个菜单项就可以引发一个动作事件。
- 在**Android**系统中, 菜单可以分为三类: 选项菜单(**Option Menu**), 上下文菜单(**Context Menu**)以及子菜单(**Sub Menu**)。

3.2.1 选项菜单(Option Menu)

- 选项菜单需要通过按下设备的Menu键来显示。当按下设备上的Menu键后，在屏幕底部弹出一个菜单，这个菜单称作选项菜单(OptionsMenu)。

1、Activity中创建菜单的方法

Activity实现选项菜单的回调方法

方 法	说 明
onCreateOptionsMenu(Menu menu)	用于初始化菜单，menu为Menu对象实例。
onPrepareOptionsMenu(Menu menu)	改变菜单状态，在菜单显示前调用。
onOptionsItemSelected(Menu menu)	菜单被关闭时调用。
onOptionsItemSelected(MenuItem item)	菜单项被点击时调用，即菜单项的监听方法。

2、菜单Menu

- 设计选项菜单需要用到Menu、MenuItem接口。一个Menu对象代表一个菜单，Menu对象中可以添加菜单项MenuItem对象，也可以添加子菜单SubMenu。
- 菜单Menu使用add(int groupId, int itemId, int order, CharSequence title) 方法添加一个菜单项。
- add()方法中的四个参数，依次是：
 - (1) 组别；
 - (2) Id，这个很重要，Android根据这个Id来确定不同的菜单；
 - (3) 顺序，哪个菜单项在前面由这个参数的大小决定；
 - (4) 文本，菜单项的显示文本。

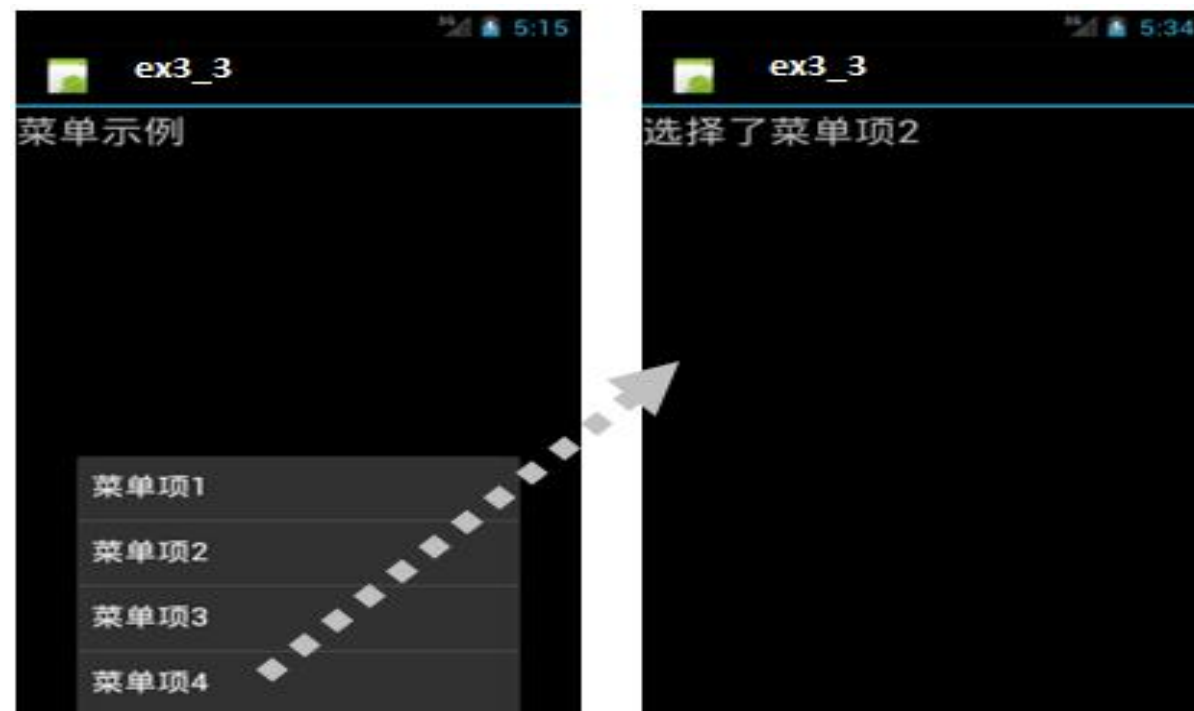
3、创建选项菜单的步骤

创建选项菜单的步骤如下：

- (1) 重写Activity的onCreateOptionsMenu(Menu menu)方法，当菜单第一次被打开时调用；
- (2) 调用Menu的add()方法添加菜单项(MenuItem)；
- (3) ， 重写Activity的onOptionsItemSelected(MenuItem item)方法，当菜单项(MenuItem)被选择时来响应事件。

【例3-3】选项菜单应用示例。

- (代码见教材)



3.2.2 上下文菜单(Context Menu)

- Android系统 的上下文菜单类似于 PC 上的右键菜单。当为一个视图注册了上下文菜单之后，长按（2 秒左右）这个视图对象就会弹出一个浮动菜单，即上下文菜单。

创建一个上下文菜单的步骤如下：

1. 重写Activity 的 onCreateContenxtMenu() 方法，调用 Menu 的 add 方法添加菜单项 (MenuItem) 。
2. 重写 Activity 的 onContextItemSelected() 方法，响应上下文菜单菜单项的单击事件。
3. 调用Activity 的 registerForContextMenu() 方法，为视图注册上下文菜单。

【例3-4】上下文菜单应用示例。

- (代码见教材)



3.3 对话框

对话框是一个有边框、有标题栏的独立存在的容器，在应用程序中经常使用对话框组件来进行人机交互。Android系统提供了四种常用对话框：

- AlertDialog: 消息对话框；
- ProgressDialog: 进度条对话框；
- DatePickerDialog: 日期选择对话框；
- TimePickerDialog: 时间选择对话框。

3.3.1 消息对话框AlertDialog

方 法	说 明
<code>AlertDialog.Builder(Context)</code>	对话框Builder对象的构造方法
<code>create()</code> ;	创建AlertDialog对象
<code>setTitle();</code>	设置对话框标题
<code>setIcon();</code>	设置对话框图标
<code>setMessage();</code>	设置对话框的提示信息
<code>setItems();</code>	设置对话框要显示的一个list
<code>setPositiveButton();</code>	在对话框中添加"yes"按钮
<code>setNegativeButton();</code>	在对话框中添加"no"按钮
<code>show();</code>	显示对话框
<code>dismiss();</code>	关闭对话框

设计AlertDialog对话框的步骤如下：

(1) 用AlertDialog.Builder类创建对话框Builder对象

```
Builder dialog=new AlertDialog.Builder(Context);
```

(2) 设置对话框的标题、图标、提示信息内容、按钮等

```
dialog.setTitle("普通对话框");
```

```
dialog.setIcon(R.drawable.icon1);
```

```
dialog.setMessage("一个简单的提示对话框")    ;
```

```
dialog.setPositiveButton("确定", new okClick())    ;
```


(3) 创建并显示AlertDialog对话框对象

```
dialog.create();
```

```
dialog.show();
```

如果在对话框内部设置了按钮，还需要对其设置事件监听OnClickListener。

【例3-5】消息对话框应用示例。

- 在本例中设计了两种形式的对话框程序，一个是发出提示信息的普通对话框，另一个是用户登录对话框。
- 在用户登录对话框中，设计了用户登录的布局文件long.xml，供用户输入相关验证信息。



(a) 普通对话框



(b) 用户登录对话框

(代码见教材)

3.3.2 几种其他常用对话框

- 1、进度条对话框ProgressDialog
- 2、日期选择对话框和时间选择对话框

1、进度条对话框ProgressDialog

方 法	说 明
getMax()	获取对话框进度的最大值
getProgress()	获取对话框当前进度值
onStart()	开始调用对话框
setMax(int max)	设置对话框进度的最大值
setMessage(CharSequence message)	设置对话框的文本内容
setProgress(int value)	设置对话框当前进度
show(Context context, CharSequence title, CharSequence message)	设置对话框的显示内容和方式
ProgressDialog(Context context)	对话框的构造方法

2、日期选择对话框和时间选择对话框

方 法	说 明
updateDate(int year, int monthOfYear, int dayOfMonth)	设置DatePickerDialog对象的当前日期
onDateChanged(DatePicker view, int year, int month, int day)	修改DatePickerDialog对象的日期
updateTime(int hourOfDay, int minuteOfHour)	设置TimePickerDialog对象的时间
onTimeChanged(TimePicker view, int hourOfDay, int minute)	修改TimePickerDialog对象的时间

【例3-6】进度及日期、时间对话框示例。

- (代码见教材)



习题三

- 1、设计一个具有两个页面的程序，第一个页面显示一张封面的图片，第二个页面显示“欢迎进入本系统”，这两个页面之间能相互切换。
- 2、设计一个具有3个选项的菜单程序，当单击每个选项时，分别跳转到3个不同的页面。
- 3、设计一个具有计算器功能的对话框程序。

Q&A

谢谢大家！