六大界面布局方式包括： 线性布局(LinearLayout)、框架布局(FrameLayout)、表格布局(TableLayout)、相对布局(RelativeLayout)、绝对布局(AbsoluteLayout)和网格布局(GridLayout) 。

Android四大组件分别为activity、service、content provider、broadcast receiver。

ndroid中，可供选择的存储方式有SharedPreferences(ʃeəd 'prefərənsɪs])、文件存储、SQLite(sk'laɪt)数据库方式、内容提供器（Content provider）和网络

RelativeLayout布局中  
android:layout\_centerHorizontal="true" --将控件置于水平方向的中心位置  
其他属性有：  
android:layout\_above="@id/xxx" --将控件置于给定ID控件之上  
android:layout\_below="@id/xxx" --将控件置于给定ID控件之下  
android:layout\_toLeftOf="@id/xxx" --将控件的右边缘和给定ID控件的左边缘对齐  
android:layout\_toRightOf="@id/xxx" --将控件的左边缘和给定ID控件的右边缘对齐  
android:layout\_alignLeft="@id/xxx" --将控件的左边缘和给定ID控件的左边缘对齐  
android:layout\_alignTop="@id/xxx" --将控件的上边缘和给定ID控件的上边缘对齐  
android:layout\_alignRight="@id/xxx" --将控件的右边缘和给定ID控件的右边缘对齐  
android:layout\_alignBottom="@id/xxx" --将控件的底边缘和给定ID控件的底边缘对齐  
android:layout\_alignParentLeft="true" --将控件的左边缘和父控件的左边缘对齐  
android:layout\_alignParentTop="true" --将控件的上边缘和父控件的上边缘对齐  
android:layout\_alignParentRight="true" --将控件的右边缘和父控件的右边缘对齐  
android:layout\_alignParentBottom="true" --将控件的底边缘和父控件的底边缘对齐  
android:layout\_centerInParent="true" --将控件置于父控件的中心位置  
android:layout\_centerVertical="true" --将控件置于垂直方向的中心位置

Intent促进了组件之间的交互，这对于开发者非常重要，而且它还能做为消息的载体，去指导组件做出相应的行为，也就是说Intent可以携带数据，传递给Activity/Service/BroadcastReceiver。

* 启动Activity。Activity可以简单的理解为手机屏幕中的一个页面，你可以通过将Intent传入startActivity方法来启动一个Activity的实例，也就是一个页面，同时，Intent也可以携带数据，传递给新的Activity。如果想要获取新建的Activity执行结果，可以通过onActivityResult()方法去启动Activity。
* 启动Service。Service是一个不呈现交互画面的后台执行操作组件，可以通过将Intent穿入startService()方法来启动一个Service来启动服务。
* 传递广播BroadCast。广播是任何应用都可以接收到的消息，通过将Intent传递给 sendBroadcast()、sendOrderedBroadcast()或 sendStickyBroadcast()方法，可以将广播传递接收方。

IntentFilter：

IntentFilter对象负责过滤掉组件无法响应和处理的Intent，只将自己关心的Intent接收进来进行处理。 IntentFilter实行“白名单”管理，即只列出组件乐意接受的Intent，但IntentFilter只会过滤隐式Intent，显式的Intent会直接传送到目标组件。 Android组件可以有一个或多个IntentFilter，每个IntentFilter之间相互独立，只需要其中一个验证通过则可。除了用于过滤广播的IntentFilter可以在代码中创建外，其他的IntentFilter必须在AndroidManifest.xml文件中进行声明。

Jpanel 面板

JFrame 框架

JButton

JRadioButton 单选控件

JLable 标签类组件

 JTextField 文本框

FlowLayout 流布局管理器 指定的对齐方式以及指定的水平和垂直间隙

[GridLayout(网格布局)](http://blog.csdn.net/buaaroid/article/details/27064059)

JPasswordField 一个只能输入数字的密码框

setVisible（true） 窗口可视

把各个JPanel加入JFrame **this**.add(jp1)

setTitle 设置自定义标题

setLayout 设置界面（布局）

setBounds 设置控件的上下左右位置

方法setsetResizable（boolean resizable）

参数为boolean类型，resizeable值为true时，表示在生成的窗体可以自由改变大小；

resizeable值为false时，表示生成的窗体大小是由程序员决定的，用户不可以自由改变该窗体的大小。

**this**.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE) 设置默认关闭方式

1.  **this**.dispose();  //关闭当前界面

2.           **new** UI();   //打开新界面

actionPerformed：

这个是ActionListener这个接口里面的方法

用于接收操作事件的侦听器接口。对处理操作事件感兴趣的类可以实现此接口，而使用该类创建的对象可使用组件的 addActionListener 方法向该组件注册。在发生操作事件时，调用该对象的 actionPerformed 方法。

Class.forName("com.mysql.jdbc.Driver"); String url = "jdbc:mysql://localhost:3306:test"; Connection conn = DriverManager.getConnection(url, "root", "root"); } } String url = "jdbc:mysql://localhost:3306:test";

 MySQL的JDBC URL编写方式：jdbc:mysql://主机名称：连接端口/数据库的名称?参数=值

提示：如果要对数据库中的某个表进行操作，需要像这样子做：String sql = "SELECT\*  FROM [数据库名].[dbo].[表名] where xxx "; 例如String sql = "SELECT\*  FROM [metro].[dbo].[4] wherexxx" 。注意，中括号是必要的，不能去掉。

5、点击右键，选择run as——>Java Application, 控制台出现下图则连接成功！