

# 学习任务二 开发小游戏

## 子任务2.1 实现游戏界面

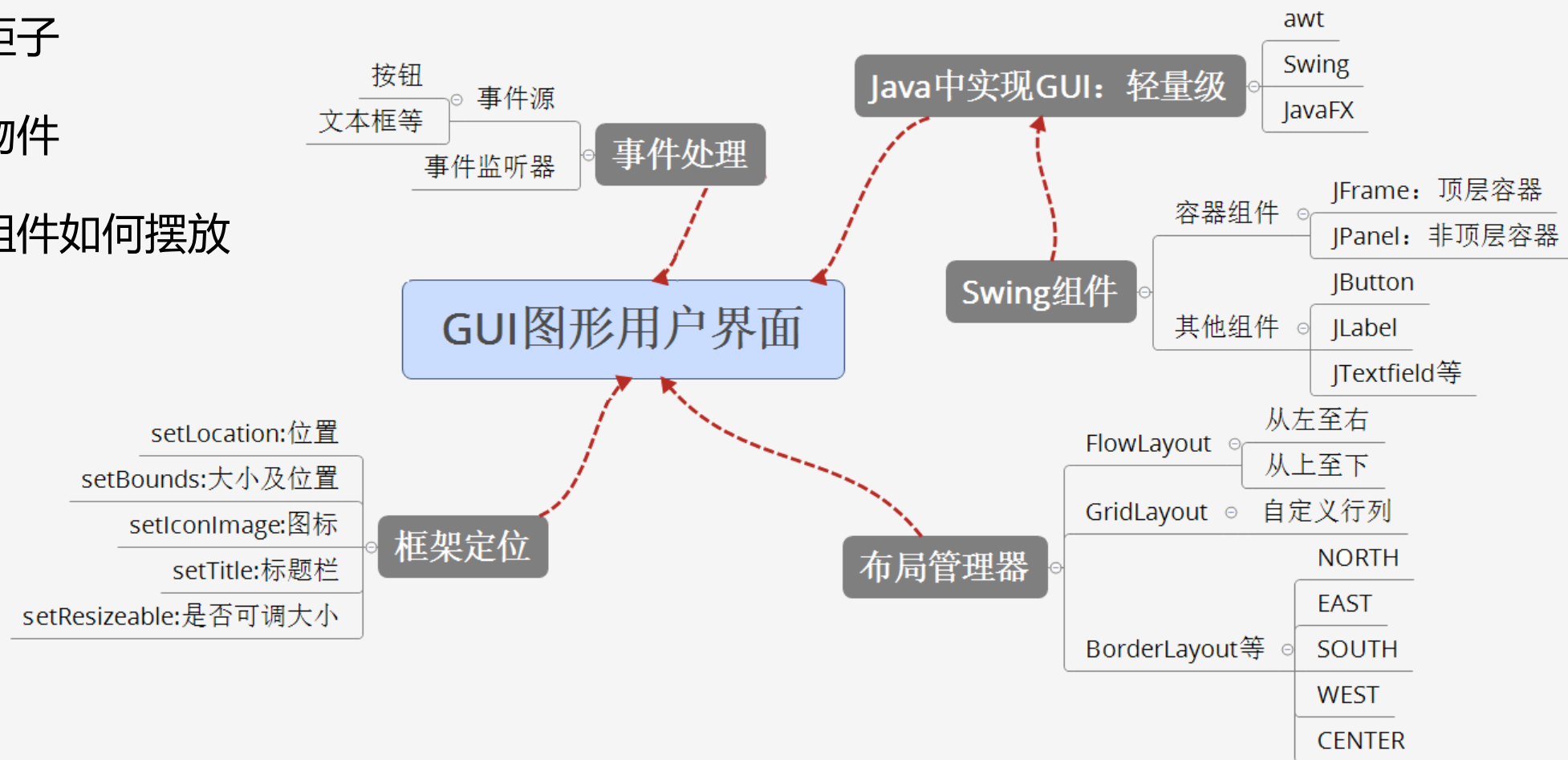
# 任务引入：设计一个打字母小游戏



# 知识准备

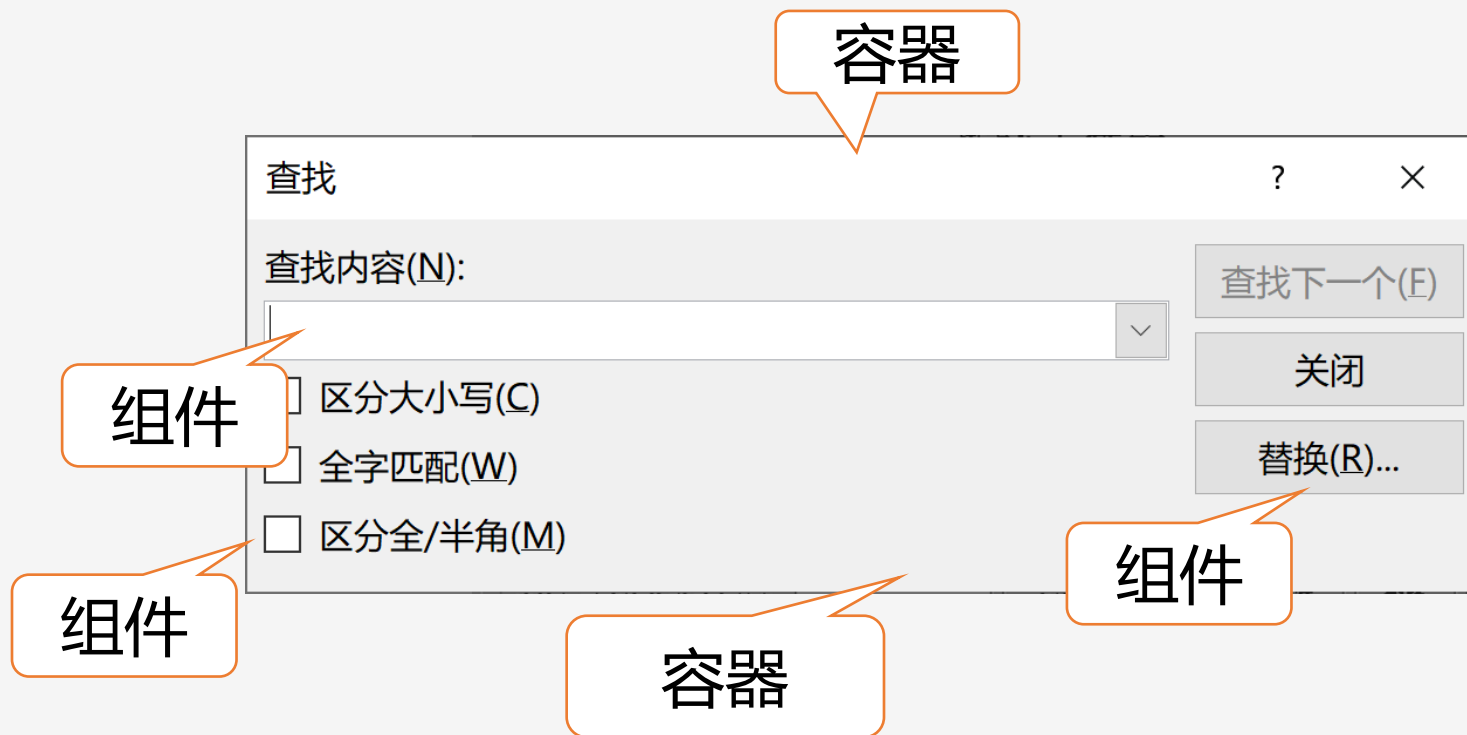
## Java界面设计的基本思路

- 容器---房子，柜子
- 组件---柜子，物件
- 布局管理器---组件如何摆放



# 知识准备——java的GUI基本原理

## ◆ 把组件放进容器里



## 知识准备——认识组件



## 知识准备——理解容器

---

- 可以存放组件的区域，可在容器上进行绘制和着色
- java.awt包中的Container类可直接或间接派生出两个常用容器：框架（Frame类）和面板（Panel类）
  - 框架是一个带有边框的独立的窗口
  - 面板是包含在窗口中的一个不带边框的区域

# 知识准备——理解容器之框架Frame

- 框架是独立于Applet和浏览器的一个窗口。
- 可以通过以下任一构造函数来创建：
  - `Frame()`: 创建一个不含标题的标准窗口
  - `Frame(String Title)`: 创建一个含有标题的窗口，这个标题是由参数 `title` 指定的。
- 当一个Frame窗口被创建以后，需要调用 `setSize()` 方法来设置窗口的大小，并调用 `setVisible()` 来显示窗口。

窗口容器



## 知识准备——理解容器之面板Panel

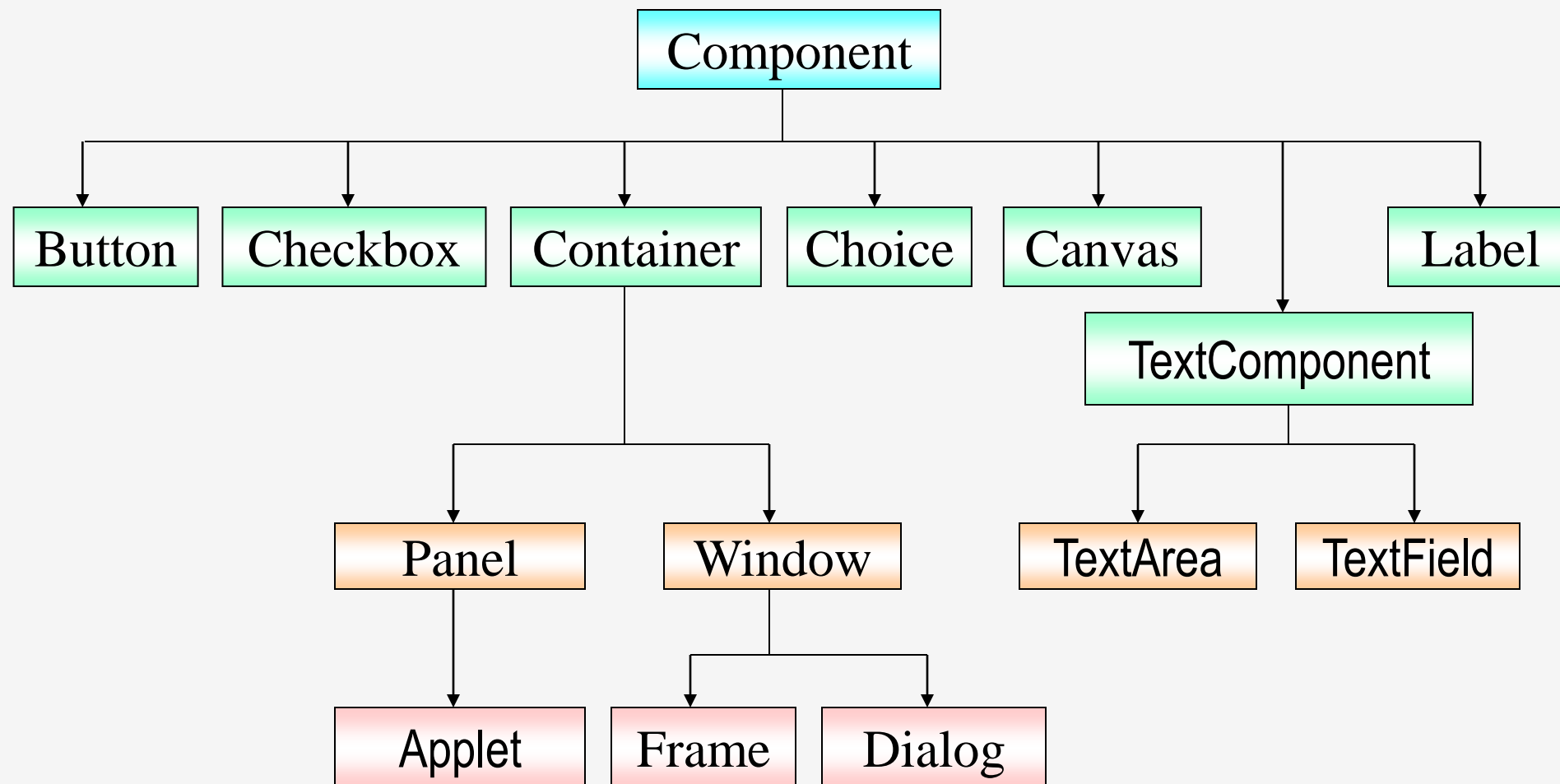
- 面板不是一个单独的窗口，它只是包含在窗口中的一个区域。
- 面板是可以将许多组件组合起来的一种容器。
- 最简单的创建面板的方式就是通过面板的构造函数 `Panel()` 来进行。
- 必须将面板添加到窗体中。



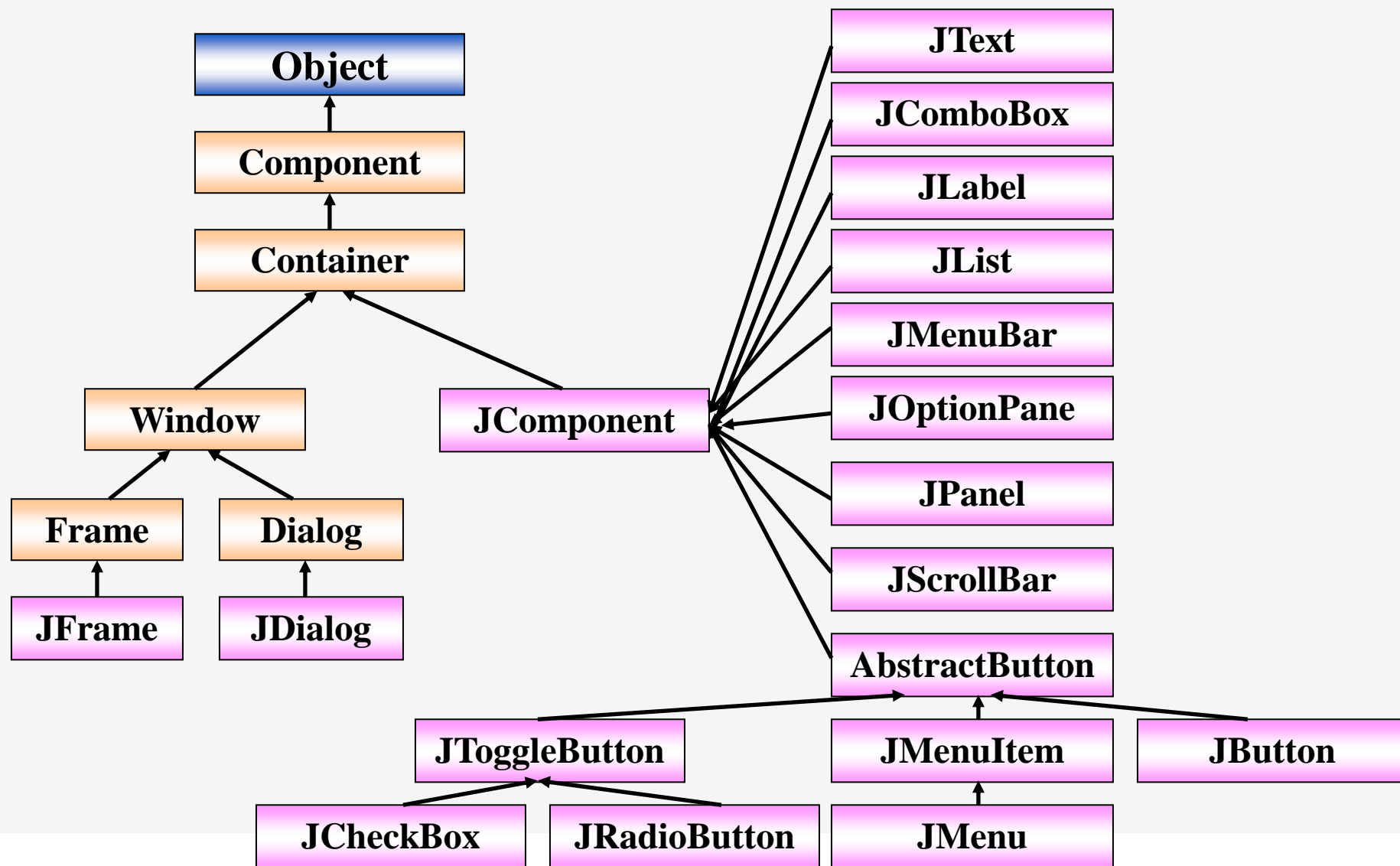


# 知识准备——GUI的框架

## Awt框架



## 知识准备——GUI的框架



# 课堂演示——使用容器示例

```
import java.awt.*;
```

使用GUI组件  
必须导入该包

```
class PanelTest extends Panel {  
    public static void main(String args[]) {  
        PanelTest p= new PanelTest();  
        Frame f=new Frame("正在测试面板！");  
        f.add(p);  
        f.setSize(300,200);  
        f.setVisible(true);  
    }  
}
```

组件和容器的概念是相对的，容器本身也可以当成组件加到其他容器中

## 知识准备——理解布局管理器

---

- 用户界面上的组件可以按不同方式排列
- Java使用布局管理器来管理这些排列方式
- 每当需要重新调整屏幕大小或重新绘制屏幕上任一项目时，就要用到布局管理器
- AWT提供一组用来进行布局管理的类，称为布局管理器或布局，所有布局都实现 `LayoutManager` 接口

组件可以add到容器，加进去后如何摆放这些组建呢？

## 知识准备——了解布局类型

---

- **Java中的布局类型包括以下几种：**
  - **FlowLayout（流式布局）**
  - **BorderLayout（边界布局）**
  - **GridLayout（网格布局）**
  - **GridBagLayout（网格包布局）**
  - **CardLayout（卡片布局）**

## 知识准备——如何设置布局

---

- 每个容器（Container对象）都有一个与它相关的缺省的布局管理器。
- 可以通过setLayout( )方法为容器设置新的布局。

# FlowLayout

---

- FlowLayout的构造函数有：
  - `FlowLayout()`:生成一个默认的流式布局
  - `FlowLayout(int alignment)`:可以设定每一行组件的对齐方式
  - `FlowLayout(int alignment,int horz,int vert)`:可以设定组件间的水平和垂直距离



# 课堂演示与练习

## 演示流式布局实现过程



选择容器



确定布局和组件



为容器确定布局



为容器添加组件



提问：流式布局有什么特点？



# GridLayout

---

- GridLayout的构造函数如下所示：
  - `GridLayout()`:生成一个单列的网格布局
  - `GridLayout(int row,int col)`:生成一个设定行数和列数的网格布局
  - `GridLayout(int row,int col,int horz,int vert)`: 可以设置组件之间的水平和垂直间隔



# 课堂演示与练习



选择容器



确定布局和组件



为容器确定布局



为容器添加组件



提问：表格式布局有什么特点？

# BorderLayout

---

- 下面是BorderLayout所定义的构造函数：
  - `BorderLayout()`: 生成默认的边界布局
  - `BorderLayout(int horz, int vert)`: 可以设定组件间的水平和垂直距离

**窗口、框架和对话框等的缺省布局**  
只有五个位置放组件：东南西北中

# 课堂演示与练习



选择容器  
↓  
确定布局和组件  
↓  
为容器确定布局  
↓  
为容器添加组件

## 登陆窗体

请输入用户名

请输入密码

提交

清除

提问：边界式布局有什么特点？

# 课堂练习——GUI综合应用

The image shows a GUI window for contact management. It has a title bar and a main content area. The main content area is divided into two columns. The left column contains labels for contact information: 姓名 (Name), 手机 (Mobile), 办公电话 (Office Phone), 家庭电话 (Home Phone), QQ, Email, and 其他 (Other). The right column contains the corresponding values: 张三 (Zhang San), 123456, 123456, 123456, 123456, zhangsan@163.com, and 空 (Empty). Below the main content area is a toolbar with three buttons: 添加 (Add), 保存 (Save), and 删除 (Delete). The window is annotated with three labels: pmain points to the main content area, pmiddle points to the left column of labels, and pdown points to the toolbar.

姓名	张三
手机	123456
办公电话	123456
家庭电话	123456
QQ	123456
Email	zhangsan@163.com
其他	空

添加 保存 删除

联系人查询

请输入要查询的联系人姓名:

查询

# 任务实施——实现游戏界面

这个游戏界面应该如何布局（先不考虑字母哪里来的）

