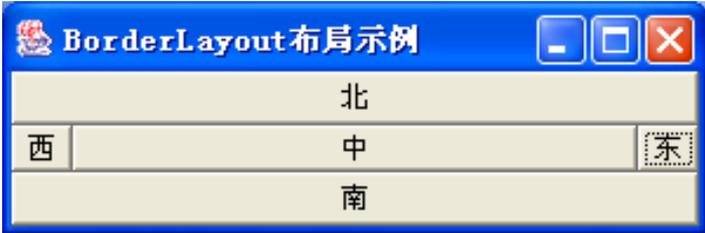


单元 7.1 键盘测试 (一)

单元教学设计

学习阶段		一、实用程序开发训练		学时	
项目 7	键盘测试	学时	6		
单元 7.1	键盘测试 (一)	学时	2		
教学目标	能力目标	知识目标		思政目标	
	能够熟练的使用布局管理器布局界面, 能够使用键盘监听, 完成程序功能。	布局管理器: 1. 了解什么是布局管理器 2. 学会使用 BorderLayout 布局管理器 3. 学会使用 FlowLayout 布局管理器 4. 学会使用 GridLayout 布局管理器	课前知识储备 课上学习练习	引导学生关注工科领域里的学科人文, 通过键盘的发展历史: 1. 让学生了解到小小键盘在布局、工艺精益求精的追求, 这正是大国工匠精神的体现。 2. 通过布局的编号, 字母排序的原因, 引导学生对人机交互、人体工程学的思考。	
课前准备	视频	马士兵讲座《布局管理器 1-FlowLayout》、《布局管理器 2-BorderLayout》、《布局管理器 3-GridLayout》			
	课前练习	1. 用 FlowLayout 在窗口上显示 20 个按钮, 按钮上分别显示数字 1-20, 每个按钮显示不同的颜色。 2. 用 BorderLayout 显示 5 个按钮, 每个按钮上分别显示东西南北中, 各个按钮显示不同的颜色。  3. 创建学生信息登记屏幕, 效果可以如下图所示 (用 GridLayout 实现): 			
课程思政	思政主题	观看科普视频《键盘布局发展史》			
	讨论主题:	1. 说说键盘发展史 2. 从人机交互的角度思考键盘布局发展的过程中所遵循的原则? 3. 那些因素决定了键盘的好坏, 有那些精益求精的工艺? 4. 那些品牌的键盘更优? 他们的优势在哪里?			
	活动方式:	弹幕、QQ 群讨论、形成云图			
教学实施					
项目	使用文本编辑器的应用程序可以帮助用户完成许多工作, 包括写 E-mail, 制定商业计划等, 这些应用程序中, 用户经常需要借助菜单和对话框定义文档格式。在本例中, 将学习如何处理键				

引入

盘事件，该事件是在按下、释放或者按下然后释放键盘上的某个按键时产生的。通过处理键盘事件，应用程序可以完成用户与键盘交互时所执行的各种操作。

本例中我们将做一个打字训练器程序，该程序满足如下需求：

为了练习“盲打”，我们设计一种软件，让学生在不看键盘的情况下，通过屏幕看到所键入的字符。该应用程序能够将学生在实际佳品中所按下的某个按键高亮显示在虚拟键盘上，同时应用程序中还包含一些菜单，实现选择字体、字号、文本颜色以及清除所显示的文本等各种功能。

课程思政

课程资源：

观看视频导入主题：1-键盘布局：按键在键盘上分布方式的图或定义；2-为啥键盘上的字母不按顺序排列？看完解开多年疑惑；这会是未来键盘的模样吗？！

讨论主题：

- 1.键盘布局的原则你知道吗？
- 2.未来的键盘会时什么样子？
- 3.我们用到的键盘都是什么品牌？好键盘好在哪里？

讨论形式：

职教云线上讨论，形成云图



Keyboard Test Utility 可以帮助您检测每一个按键，没敲打一下虚拟键盘就会在所打的键位上高亮颜色，

项目进度安排

单元	功能要求	课前准备	课上学习练习
1	设计程序界面	1. 了解布局管理器 2. 学会使用FlowLayout 布局管理器	1. 熟练使用 BorderLayout 布局管理器 2. 熟练使用 GridLayout 布局管理器 3. 了解 ASCII
2	实现程序功能	1. 了解内部类	1. 学会用内部类实现事件监听器 2. 实现学会使用键盘监听器 3. 学会使用菜单 4. 学会使用

十分钟测试

测试 1：用 GridLayout 实现如下界面。

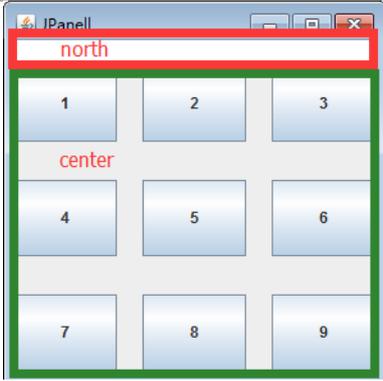


提示：

1. 1-9 九个按钮，用 Jbutton 数组来标示，用 for 循环初始化按钮。
2. 用 3*3 的 GridLayout，用 for 循环将按钮依次放在格子里。
3. 改进：一次循环，是否可以完成初始化+放入格子？

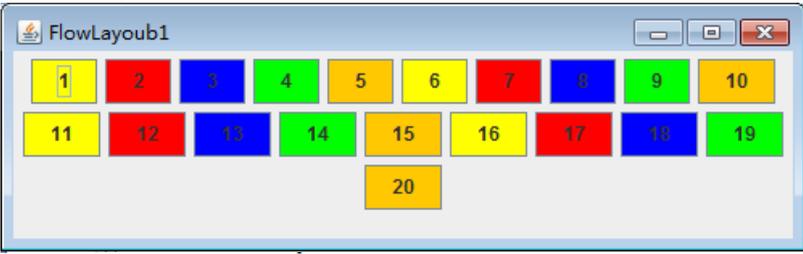
拓展：

1. 表格有哪些属性？对齐方式，上下边距，左右边距，GridLayout 是也能自定义？

能力 训练 任务	任务一	<p>用 BorderLayout 和 GridLayout 共同实现如下窗口：</p> <p>测试 1： （1）在 north 显示一个 textArea （2）在 center 显示按钮。</p> <p>测试 2： （1）在 center 显示 TextArea （2）在 south 显示按钮。</p> <p>思考： 有什么不同？</p>		
		<p>课程思政</p> <p>知识点：抽象类和接口含抽象方法不能创建对象，要利用子类或者实现类去生成对象；接口是一种规范，它只用来声明规则。面向接口编程具有相当大的灵活性。</p> <p>思政元素：进行价值塑造与思想引领，以及人文精神培养。</p> <p>①提醒学生不要空想，一方面要树立远大理想，另一方面要为了实现理想去脚踏实地地努力奋斗，才能有所收获将来为国家做更大的贡献。</p> <p>②对于既定的标准与规则，每一位公民都要遵守。</p> <p>③标准及规范性与灵活性的辩证关系问题，可提升学生的人文精神素养。可引导学生在处理学习、生活等方面的事情时运用唯物辩证法思考问题：在大是大非面前讲规矩、讲纪律，在允许自由裁量范围内讲灵活性。</p>		
	任务二	将 1-9，9 个数字，换成 A-19 个字母。		
	任务三	模拟键盘，具体要求如下： 第一排显示 0-9，共 10 个数字， 第二排显示 A-L，共 12 个字母， 第三排显示 M-Z，共 12 个字母。		
	课后作业	灵活运用布局管理器，模拟一个真实的键盘。		
	课后学习资源			

单元教学进度设计

Step1: 项目导入 (20) 分钟

教学环节	教学内容	教师活动	学生活动
作业点评	<p>用 FlowLayout 在窗口上显示 20 个按钮，按钮上分别显示数字 1-20.</p>  <p><code>JButton[] b=new JButton[20];</code> <code>Color[] color=new Color[5];</code> → 声明按钮数组和颜色数组</p> <p>1.</p>	点评	听讲记录

```

color[0]=Color.yellow;
color[1]=Color.red;
color[2]=Color.blue;
color[3]=Color.green;
color[4]=Color.orange;

for(int i=0;i<b.length;i++){
    b[i]=new JButton(""+(i+1));
    b[i].setBackground(color[i%5]);
    c.add(b[i]);
}

```

初始化颜色数组

初始化按钮数组

将颜色添加到按钮上

提示：灵活使用数组。

提示

思考

知识拓展：

FlowLayout 是最基本的布局管理器，它是 Panel、Applet 等容器默认的布局管理器，也称为流布局。添加到容器上的各个组件按照它们被添加的顺序从左到右依次排列，一行摆满后，就自动转到下一行继续摆放。

1) 构造方法

(1) `FlowLayout()` 创建一个流布局对象。系统默认组件以居中方式对齐，且组件之间的横向与竖向间距为 5。

(2) `FlowLayout(int align)` 创建一个流布局对象。align 指定组件的对齐方式，组件之间的横向与竖向间距为 5。

(3) `FlowLayout(int align, int hgap, int vgap)` 创建一个流布局对象。align 指定组件的对齐方式。组件之间横向与竖向间距分别由 hgap 和 vgap 指定。

注意：对齐方式 align 可以取类常数值。

2) 类常数

用于对齐方式的常数如下：

- (1) `LEFT` 其值为 0，表示每行组件都是左对齐。
- (2) `CENTER` 其值为 1，表示每行组件都是居中对齐。
- (3) `RIGHT` 其值为 2，表示每行组件都是右对齐。
- (4) `LEADING` 其值为 3，表示每行组件都与容器方向的开始边对齐。
- (5) `TRAILING` 其值为 4，表示每行组件都与容器方向的结束边对齐。

3) 常用方法

- (1) `public int getAlignment()` 获得组件的对齐方式。
- (2) `public void setAlignment(int align)` 设置组件的对齐方式。

在前边的例子中，我们已经使用过 FlowLayout 布局。应该注意的是，使用该布局布局的组件，不因容器大小的改变而改变，即组件的大小是不变的。

点评

听讲记录

改进 1：将界面改为左对齐。

方法 (1)： `FlowLayout fl=new FlowLayout(FlowLayout.LEFT);`

方法 (2)： `fl.setAlignment(fl.LEFT);`

编程

听讲记录

改进 2：改变组件之间的间距。

`FlowLayout fl=new FlowLayout(FlowLayout.LEFT,20,30);`

`c.setLAYOUT(fl);` 行间距 列间距

编程

听讲记录

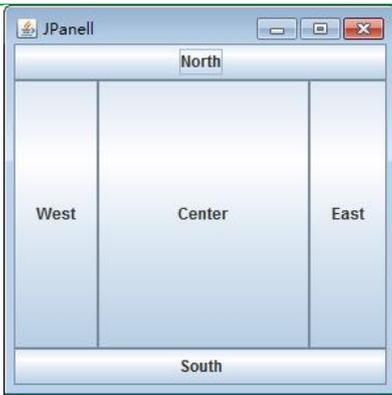
自己测试一下吧。

辅导

编程

作业二 2. 用 BorderLayout 显示 5 个按钮，每个按钮上分别显示东西南北中，各个按钮显示不同的颜色。

思考：如果改变方向的次序，界面会不会发生改变？



```
c.add(b_center,bl.CENTER);
c.add(b_east,bl.EAST);
c.add(b_north,bl.NORTH);
c.add(b_south,bl.SOUTH);
c.add(b_west,bl.WEST);
```

点评

听讲记录

知识拓展与总结

BorderLayout 布局管理器是一种简单的布局管理器，它将容器划分为东、西、南、北、中 5 个区域。当容器使用该布局时，每次添加组件都要指明把它放在哪个区域。它是 Frame 容器默认的布局管理器。

1) 构造方法

(1) `BorderLayout()` 创建一个 BorderLayout 布局管理器。

(2) `BorderLayout(int hgap, int vgap)` 创建一个 BorderLayout 布局管理器，hgap 和 vgap 分别指定组件之间的水平和垂直距离。

2) 类常数

以下类常数用于指定组件在容器中的摆放位置。

- (1) `EAST` 其值为"East"，摆放在右边（东）。
- (2) `WEST` 其值为"West"，摆放在左边（西）。
- (3) `SOUTH` 其值为"South"，摆放在底部（南）。
- (4) `NORTH` 其值为"North"，摆放在顶部（北）。
- (5) `CENTER` 其值为"Center"，摆放在中部。

3) 常用方法

- (1) `public int getHgap()` 获得组件之间的水平距离。
- (2) `public void setHgap(int hgap)` 设置组件之间的水平距离为 hgap。
- (3) `public int getVgap()` 获得组件之间的垂直距离。
- (4) `public void setVgap(int vgap)` 设置组件之间的垂直距离为 vgap。

改进 1: 改变组件之间的间距。

```
BorderLayout bl=new BorderLayout(20,30);
```

20: 水平距离, 30: 垂直距离

创建学生信息登记屏幕，效果可以如下图所示（用 GridLayout 实现）：



```
public class TestGridLayout1 extends JFrame{
    String [] mark={"身份证号","姓名","别名","性别","出生年月","出生地",
    "学号","成绩","备注"};
    JLabel [] lab; //声明标签数组显示标识
    JTextField [] text; //声明文本框数组输入各项信息
    Button bt1, bt2; //声明两个按钮
    public TestGridLayout1()
```

作业三

```

{
    setLayout(new GridLayout(0, 6)); //设置网格布局, 每行6列, 行数不
    定

    lab=new JLabel[mark.length]; //定义标签数组的大小
    text=new JTextField[mark.length]; //定义文本框数组的大小

    for(int i=0;i<mark.length;i++)
    {
        lab[i]=new JLabel(mark[i]); //创建每个标签对象
        text[i]=new JTextField();
        add(lab[i]); //将每个标签加入容器
        add(text[i]); //将每个文本框加入容器
    }

    bt1=new Button("重置"); //创建按钮对象bt1
    bt2=new Button("提交"); //创建按钮对象bt2
    add(new Label()); //为了按钮的摆放位置, 添加一个空标签
    add(bt1); //将按钮bt1加入容器
    add(new Label()); // 添加一个空标签
    add(bt2); //将按钮bt2加入容器
    this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    this.setTitle("GridLayout布局示例");
    this.setSize(400, 150); //设置容器的大小
    this.setVisible(true);
}

public static void main(String[] args){
    TestGridLayout t=new TestGridLayout();
}
}

```

注意:

1. GridLayout, 行或列的值设置为 0 的时候, 标示行数不定或者列数不定, 但不能同时为 0。
2. 灵活应用数组可以事半功倍点击解决问题。
自己联系一下吧。

十分钟
测试

测试 1: 用 GridLayout 实现如下界面。



提示:

1. 1-9 九个按钮, 用 JButton 数组来标示, 用 for 循环初始化按钮。
2. 用 3*3 的 GridLayout, 用 for 循环将按钮依次放在格子里。
3. 改进: 一次循环, 是否可以完成初始化+放入格子?

拓展:

1. 表格有哪些属性? 对齐方式, 上下边距, 左右边距, GridLayout 是也能自定义?

GridLayout 布局管理器将容器划分成 m 行 n 列的网格, 添加到容器中的组件按行列顺序被依次放置到每个网格中。网格的大小是一样的, 因此, 被放在网格

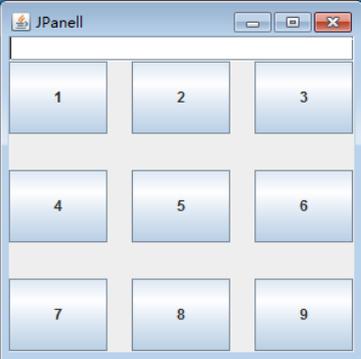
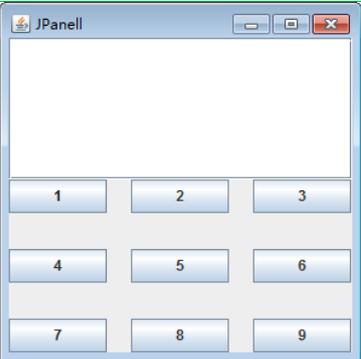
	<p>中组件的大小也是一样的。</p> <p>1) 构造方法</p> <p>(1) <code>GridLayout()</code> 创建一个网格布局对象。所有的组件都被放在一行上且组件之间无间距。</p> <p>(2) <code>GridLayout(int rows, int cols)</code> 创建具有 rows 行 cols 列的网格布局对象。其中 rows 和 cols 中可以有一个为零（但不能两者同时为零），这表示可以将任何数目的组件对象置于行或列中。</p> <p>(3) <code>GridLayout(int rows, int cols, int hgap, int vgap)</code> 和第二个构造方法类似，但添加了组件之间的横向间距 hgap 和竖向间距 vgap。</p> <p>2) 常用方法</p> <p>(1) <code>public int getColumns()</code> 获得布局的列数。</p> <p>(2) <code>public int getRows()</code> 获得布局的行数。</p> <p>(3) <code>public void setColumns(int cols)</code> 设置布局列数为 cols。</p> <p>(4) <code>public void setRows(int rows)</code> 设置布局行数为 rows。</p> <p>下面给出一个使用 GridLayout 布局管理器的示例。</p>			
--	--	--	--	--

新课导入

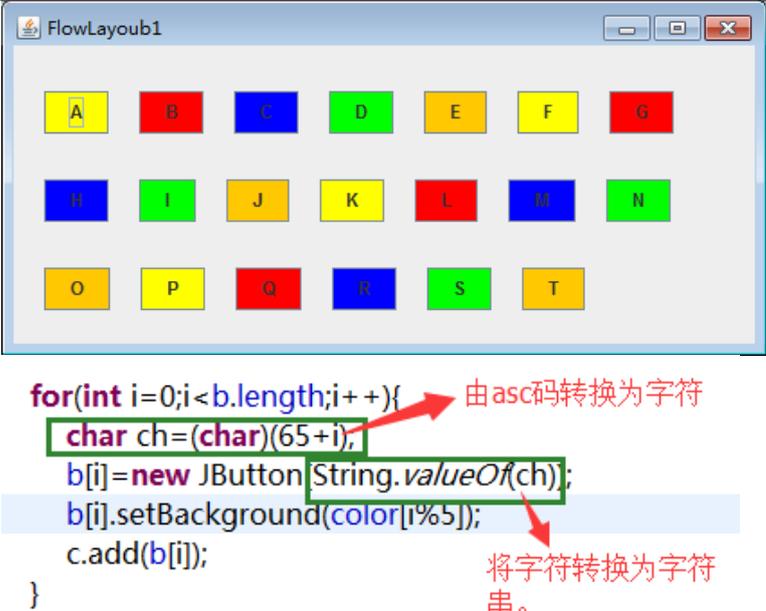
教学环节	教学内容	教师活动	学生活动
	<p>使用文本编辑器的应用程序可以帮助用户完成许多工作，包括写 E-mail，制定商业计划等，这些应用程序中，用户经常需要借助菜单和对话框定义文档格式。在本例中，将学习如何处理键盘事件，该事件是在按下、释放或者按下然后释放键盘上的某个按键时产生的。通过处理键盘事件，应用程序可以完成用户与键盘交互时所执行的各种操作。</p> <p>本例中我们将做一个打字训练器程序，该程序满足如下需求：</p> <p>为了练习“盲打”，我们设计一种软件，让学生在不看键盘的情况下，通过屏幕看到所键入的字符。该应用程序能够将学生在实际佳品中所按下的某个按键高亮显示在虚拟键盘上，同时应用程序中还包含一些菜单，实现选择字体、字号、文本颜色以及清除所显示的文本等各种功能。</p>	提出问题	讨论
	分析	分析	思考
单元	功能要求	课前准备	课上学习练习
1	点击按键，相应按钮高亮显示。	布局管理器。	1. java 的内部类和键盘监听器。 2. JPanel 的使用。
2	添加菜单，选择字体、字号、文本颜色以及清除所显示的文本。	Java 中的 Object 类。	1. 菜单 2. 颜色选择器 3. ASCII 码
	<p>课程思政</p> <p>课程资源：</p> <p>观看视频导入主题：1-键盘布局：按键在键盘上分布方式的图或定义；2-为啥键盘上的字母不按顺序排列？看完解开多年疑惑；这会是未来键盘的模样吗？！</p> <p>讨论主题：</p> <p>1.键盘布局的原则你知道吗？</p> <p>2.未来的键盘会时什么样子？</p> <p>3.我们用到的键盘都是什么品牌？好键盘好在哪里？</p> <p>讨论形式：</p> <p style="text-align: center;">职教云线上讨论，形成云图</p>	主持引导	参与讨论

Step2: 任务实施

任务 1: (40 分钟)

教学环节	教学内容	教师活动	学生活动											
任务引入	在测试窗口的基础上，用 BorderLayout，添加一个 JTextArea。	布置任务	思考											
任务部署	 <p>中间容器 JPanel 1. 中间容器中可以放组件。 2. 中间容器中依然可以放中间容器。 如何理解？ 宣传栏中的海报。。。。</p>	讲解	思考听讲											
	<table border="1"> <thead> <tr> <th>常用构造函数及方法</th> <th>用途</th> </tr> </thead> <tbody> <tr> <td>JPanel ()</td> <td>创建一个 JPanel 中间容器。</td> </tr> <tr> <td>JPanel(LayoutManager layout)</td> <td>创建一个 JPanel 中间容器，具有指定的布局管理。</td> </tr> <tr> <td>void add(Component comp)</td> <td>将组件添加到 JPanel 面板上。</td> </tr> <tr> <td>void setBackground(Color c)</td> <td>设置 JPanel 的背景色。</td> </tr> <tr> <td>void setLayout(LayoutManager mgr)</td> <td>设置 JPanel 的布局管理。</td> </tr> </tbody> </table>	常用构造函数及方法	用途	JPanel ()	创建一个 JPanel 中间容器。	JPanel(LayoutManager layout)	创建一个 JPanel 中间容器，具有指定的布局管理。	void add(Component comp)	将组件添加到 JPanel 面板上。	void setBackground(Color c)	设置 JPanel 的背景色。	void setLayout(LayoutManager mgr)	设置 JPanel 的布局管理。	
常用构造函数及方法	用途													
JPanel ()	创建一个 JPanel 中间容器。													
JPanel(LayoutManager layout)	创建一个 JPanel 中间容器，具有指定的布局管理。													
void add(Component comp)	将组件添加到 JPanel 面板上。													
void setBackground(Color c)	设置 JPanel 的背景色。													
void setLayout(LayoutManager mgr)	设置 JPanel 的布局管理。													
任务实施 1	(1) 在 north 显示一个 textArea (2) 在 center 显示按钮。	演示	学习编程											
	学生做实验。	辅导	编程											
任务部署	 <p>(1) 在 center 显示 TextArea (2) 在 south 显示按钮。</p>	讲解	听讲											
任务实施 2	<p>由学生自己做实验。</p> <p>课程思政</p> <p>知识点：抽象类和接口含抽象方法不能创建对象，要利用子类或者实现类去生成对象；接口是一种规范，它只用来声明规则。面向接口编程具有相当大的灵活性。</p> <p>思政元素：进行价值塑造与思想引领，以及人文精神培养。</p> <p>①提醒学生不要空想，一方面要树立远大理想，另一方面要为了实现理想去脚踏实地地努力奋斗，才能有所收获将来为国家做更大的贡献。</p> <p>②对于既定的标准与规则，每一位公民都要遵守。</p> <p>③标准及规范性与灵活性的辩证关系问题，可提升学生的人文精神素养。可引导学生在处理学习、生活等方面的事情时运用唯物辩证法思考问题：在大是大非面前讲规矩、讲纪律，在允许自由裁量范围内讲灵活性。</p>	辅导	编程											

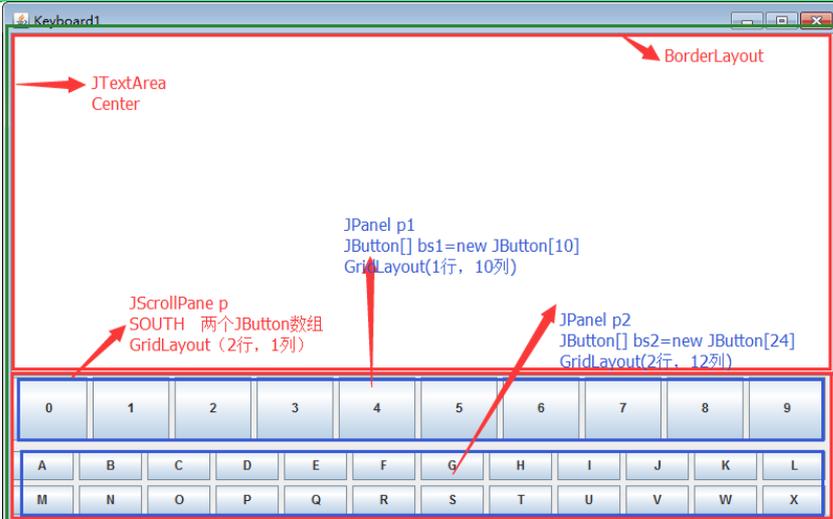
任务 2: (45 分钟)

教学环节	教学内容	教师活动	学生活动
任务引入	将 1-20 共 20 个数字，换成 A-T 共 20 个字母。	引入	思考
任务部署	<p>了解一下 ASCII (American Standard Code for Information Interchange) 美国标准信息交换码 为什么使用 ASCII 码?</p> <p>计算机中，所有的数据在存储和运算时都要使用二进制数表示（因为计算机用高电平和低电平分别表示 1 和 0），例如，像 a、b、c、d 这样的 52 个字母（包括大写）、以及 0、1 等数字还有一些常用的符号（例如*、#、@等）在计算机中存储时也要使用二进制数来表示，而具体用哪些二进制数字表示哪个符号，当然每个人都可以约定自己的一套（这就叫编码），而大家如果要想互相通信而不造成混乱，那么大家就必须使用相同的编码规则，于是美国有关的标准化组织就出台了 ASCII 编码，统一规定了上述常用符号用哪些二进制数来表示。</p> <p>ASCII 是应用最广泛的用于文本文件的字符格式，学计算机的和上网的人几乎没有人不知道它的。在一个 ASCII 文件中，每个数字，字母或特殊字符都对应于一个 7 位二进制数，这个 7 位二进制数还是以以一个字节，也就是 8 个字节表示，大家可以算一下 7 位二进制数可以表示多个字符，数字不会很大的。</p> <p>UNIX 和基于 DOS 的操作系统文本文件都是 ASCII 码形式的，而 NT 操作系统使用西欧字符集作为它文本文件的内容，而在 IBM 的 System 390 服务器中，使用的是另一种编码 EBCDIC 码，它的使个码字对应于一个 8 位字符。最后，希望大家知道的是 ASCII 码是由美国国家标准局(ANSI)制定的。</p> <p>0-9 (48-57) A-Z (65-90) a-z (97-122)</p>	讲解总结	思考
任务实施	 <pre> for(int i=0;i<b.length;i++){ char ch=(char)(65+i); b[i]=new JButton(String.valueOf(ch)); b[i].setBackground(color[i%5]); c.add(b[i]); } </pre> <p>由asc码转换为字符</p> <p>将字符转换为字符串。</p>	编程	思考 计算

任务 3: (45 分钟)

教学环节	教学内容	教师活动	学生活动
任务引入	<p>模拟键盘，具体要求如下： 第一排显示 0-9，共 10 个数字， 第二排显示 A-L，共 12 个字母，</p>	引入	思考

入 第三排显示 M-Z，共 12 个字母。



引导

思考
讨论

任务部署

```
JTextArea ta=new JTextArea();  
JScrollPane sp=new JScrollPane(ta);
```

```
JPanel p1=new JPanel();  
JPanel p2=new JPanel();  
JPanel p=new JPanel();
```

将组建定义为
成员变量

```
JButton[] bs1=new JButton[10];  
JButton[] bs2=new JButton[24];
```

```
Container c=this.getContentPane();  
BorderLayout bl=new BorderLayout();  
c.setLayout(bl);  
GridLayout gl1=new GridLayout(1,10,5,5);  
p1.setLayout(gl1);  
GridLayout gl2=new GridLayout(2,12,5,5);  
p2.setLayout(gl2);  
GridLayout gl=new GridLayout(2,1,10,10);  
p.setLayout(gl);
```

设置各个容器的
布局管理器

```
for(int i=0;i<bs1.length;i++){  
    char ch=(char)(48+i);  
    bs1[i]=new JButton(String.valueOf(ch));  
    p1.add(bs1[i]);  
}
```

0-9 10个按钮添加到中间容器P1

```
for(int i=0;i<bs2.length;i++){  
    char ch=(char)(65+i);  
    bs2[i]=new JButton(String.valueOf(ch));  
    p2.add(bs2[i]);  
}
```

将A-Z共24个按钮，添加到中间容器P2

	<pre>p.add(p1); p.add(p2); c.add(sp,bl.CENTER); c.add(p,bl.SOUTH);</pre>  <p>将各个中间容器，分别添加。</p>		
任务实施	以小组为单位，互相讨论，完成 keyBoard 界面。	辅导	编程

四、总结与课后安排

教学环节	教学内容	教师活动	学生活动
教学小结	<p>1. BorderLayout 布局管理器 BorderLayout 布局管理器是一种简单的布局管理器，它将容器划分为东、西、南、北、中 5 个区域。当容器使用该布局时，每次添加组件都要指明把它放在哪个区域。它是 Frame 容器默认的布局管理器。</p> <p>2. FlowLayout 布局管理器 FlowLayout 是最基本的布局管理器，它是 Panel、Applet 等容器默认的布局管理器，也称为流布局。添加到容器上的各个组件按照它们被添加的顺序从左到右依次排列，一行摆满后，就自动转到下一行继续摆放。</p> <p>3. GridLayout 布局管理器 GridLayout 布局管理器将容器划分成 m 行 n 列的网格，添加到容器中的组件按行列顺序被依次放置到每个网格中。网格的大小是一样的，因此，被放在网格中组件的大小也是一样的。</p> <p>4. AscII</p>	总结	听讲记录
布置作业	<p>1. 同步课外项目：用布局管理器，模拟一个真实的键盘。</p> <p>2. 本节复习：课堂录像、项目说明文档</p> <p>3. 作业要求：程序上传至云盘，放在 *组-姓名 文件夹下。</p>		
课后学习资源	<p>下节预告 项目 2.3.2 键盘测试程序（二） 课前储备： 微课《Java 中的内部类》</p>		
课程思政	<p>1.键盘布局在发展过程中遵循的人体工程学原理</p> <p>2.键盘的工艺追求。</p>		