

一、项目导入 (20) 分钟

主题导入

教学环节	教学内容	教师活动	学生活动
反馈	反馈作业的完成情况 1.ATM 取款机界面 2.计算器界面 3.门禁系统界面	反馈	
提问	现有两个类 A、B，以下描述中表示 B 继承自 A 的是()。 A) class A extends B B) class B implements A C) class A implements B D) class B extends A 答案：D	提问 分析	讨论 回答
	Java 中的每一个类（多选）： A)只能有一个父类 B)可以有多个父类 C)可以没有父类 D)只能实现一个接口 答案：A C	提问 分析	讨论 回答
	下述关于继承的说法正确的是： A) 要从一个父类派生出一个子类，就要将父类代码全部复制过来，再添上子类特有的代码。 B) 声明继承某个父类就意味着可以使用父类的全部非 private 变量和方法。 C) 被继承的父类代码必须与子类代码在同一个源代码文件（.java 文件）内。 D) 父类的字节码文件（.class 文件）必须与子类的字节码文件放在同一文件夹中。 答案：B	提问 分析	讨论 回答
	用 abstract 修饰的类称为抽象类，它们： A) 只能用以派生新类，不能用以创建对象； B) 只能用以创建对象，不能用以派生新类； C) 既可用以创建对象，也可用以派生新类；	提问 分析	讨论 回答

	D) 既不能用以创建对象，也不可用来派生新类。 答案：A		
	关于接口以下说法正确的是： A) 接口中的变量必须用 public static final 三个修饰词修饰； B) 接口中的方法必须用 public abstract 两个修饰符修饰； C) 一个接口可以继承多个父接口； D) 接口的构造方法名必须为接口名。 答案：C 对接口定义说明如下： 1) 接口的访问限定只有 public 和缺省的。 2) interface 是声明接口的关键字，与 class 类似。 3) 接口的命名必须符合标识符的规定，并且接口名必须与文件名相同。 4) 允许接口的多重继承，通过 “extends 父接口名列表” 可以继承多个接口。 5) 对接口体中定义的常量，系统默认为是 “static final” 修饰的，不需要指定。 6) 对接口体中声明的方法，系统默认为是 “abstract” 的，也不需要指定；对于一些特殊用途的接口，在处理过程中会遇到某些异常，可以在声明方法时加上 “throw 异常列表”，以便捕捉出现在异常列表中的异常。有关异常的概念将在后边的章节讨论。	提问 分析	讨论 回答
	以下关于接口对象的说法中正确的是： A) 接口只能被类实现，不能用来声明对象； B) 接口对象可以用关键词 new 创建； C) 接口对象可以等于任何类的对象； D) 接口对象只能等于实现该接口的类的对象； 答案：D	提问 分析	讨论 回答
	若有如下接口 A 的定义，下列哪些类下确实实现了该接口 <pre>interface A { void method1(int i); void method2(int j); }</pre>	提问 分析	讨论 回答

	<pre> } A) class B implements A{ void method1() {} void method2() {} } B) class B { void method1(int i) {} void method2(int j) {} } C) class B implements A { void method1(int i) {} void method2(int j) {} } D) class B implements A{ public void method1(int x) {} public void method2(int y) {} } 答案：C </pre>		
--	---	--	--

新课导入

教学环节	教学内容	教师活动	学生活动								
	某制药公司希望配备设备的实验室外安装一个门禁系统，只有得到授权并取得保护密码的人员才被允许进入实验室。以下便提供了一些有效密码及所代表的雇员小组： <table><tr><th>保护密码</th><th>雇员小组</th></tr><tr><td>1645</td><td>技术人员</td></tr><tr><td>8345</td><td>管理人员</td></tr><tr><td>9998, 1006, 1007, 1008</td><td>科学家</td></tr></table> 当输入保护密码后，应使该保护密码对任何可能站在门禁系统前的人员都是不可见的。而所输入的每一个保护密码，其请求要么是被允许，要么是被拒绝。所有试图进入实验室的请求都将显示在位于小键盘下的一个屏幕中，如果请求被允许，则相应的日期、时间及其所属的小组名称（科学家、管理员等）均会显示在屏幕中；如果请求被拒绝，则相应的时间、日期以及一条错误信息也会显示在屏幕中。	保护密码	雇员小组	1645	技术人员	8345	管理人员	9998, 1006, 1007, 1008	科学家	提出问题	讨论 思考
保护密码	雇员小组										
1645	技术人员										
8345	管理人员										
9998, 1006, 1007, 1008	科学家										
	在上一节课的学习中，我们已经实现了门禁系统的界面，同学们根据自己的理解，设计了如下界面（PPT 展示），有了界面，我们需要给	分析	讨论								

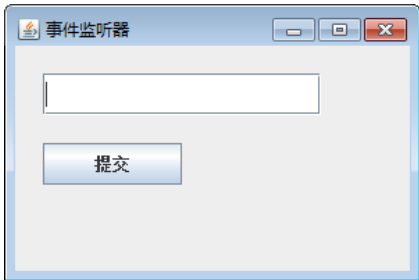
按钮加上动作，那么如何实现呢？		观看
什么是事件监听器？		
我们单击按钮，程序自动进行逻辑处理，并显示结果，这个过程就是 java 的事件处理机制完成的。首先我们要理解几个概念： 所谓事件，即用户对组件的一个操作，比如用鼠标点击按钮；所谓事件源即发生事件的组件，比如“计算”按钮； 所谓事件处理器即负责处理事件的方法。 Java 程序对事件的处理放在一个类对象中，这个类对象就是事件监听器，我们必须将一个事件监听器同某个事件源相关联，这样，某个时间源上发生了某种事件后，关联的事件监听器对象中的有关代码才能被执行。我们把这个关联的过程成为向事件源注册事件监听器对象。事件处理器首先与组件相关联，当组建收到外部作用时，把此对象传给事件监听器，事件处理器就会被启动并执行相关的代码。 Java.awt.event 包中定义了若干事件类，如 MouseEvent、WindowEvent、ActionEvent 等。我们在这个程序中使用了 ActionEvent 类，如果用户的一个动作导致了某个组件本身最基本的动作发生了，这就是 ActionEvent 事件，如鼠标单击按钮、按下键盘等。 以我们的程序为例，b.addActionListener(this) 完成了注册事件监听器的功能。即组件 b（事件源）与监听器 this 进行了关联。那么，this 又是什么呢？我们在学习 java 基础的时候知道 this 用来指代对象本身，就好比我说“我吃饭了”中的“我”一样，用来指代了自身，那么程序中的 this 就是指代了我们运行后看到的窗口。那么，这个窗口就是监听器吗？答案是肯定的，…… implements ActionListener，即我们的窗口类实例化了 ActionListener 接口类，也就是说它不但是一个窗口，而且是一个监听器。这个接口类有一个 actionPerformed(ActionEvent e) 抽象方法，我们只要继承 ActionListener 接口，就必须实现这个抽象方法。这个方法有一个参数，正是我们的 ActionEvent 事件 e。也就是说只要有事件产生，我们的程序将默认的调用 actionPerformed(ActionEvent e) 方法，然后判断事件源是什么 (e.getSource()==b)，在根据不同的事件源进行不同的处理。一个事件监听器可以关联在若干个组件上。	讲解	听讲

二、课前测试与知识梳理（20 分钟）

三、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	添加事件监听器常见的有 4 种方法，当然，我们只要掌握一种就可以了。	布置任务	思考
任务部署 1	在一个窗口上，有一个按钮，一个文本框，点击按钮后，在文本框中输出“你点击了按钮。”	布置任务	
任务实施 1	5分钟时间，自己定义有页面。 	编程辅导	编程
任务部署 2	自身类作为事件监听器（最常用） (1) 该类实现 <u>ActionListener</u> 接口。 (2) 在组件上添加事件监听器：b.addActionListener(this); (3) 实现 actionPerformed 方法。 <pre>public void actionPerformed(ActionEvent e) { t.setText("您点击了按钮! "); }</pre>	编程	编程模仿
任务实施 2	<pre>public class Practice3_ActionListener extends JFrame implements ActionListener{ private JTextField t=new JTextField(); private JButton b=new JButton("提交"); public Practice3_ActionListener(){ Container c=this.getContentPane(); c.setLayout(null);</pre>		

```
t.setBounds(20, 20, 200, 30);

b.setBounds(20, 70, 100, 30);

b.addActionListener(this);

c.add(t);
c.add(b);

this.setTitle("事件监听器");

this.setSize(300, 200);

this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

this.setVisible(true);

}

public static void main(String[] args){

    Practice3_ActionListener p=new
Practice3_ActionListener();

}

@Override

public void actionPerformed(ActionEvent e) {

    t.setText("您点击了按钮! ");
```

	<pre> } } 分析： 事件监听的过程。 </pre>		
--	---	--	--

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	如果窗口上有若干个按钮，如何判断事件是由那个按钮发出的？  三种方式： 1. 判断事件源 2. 判断事件上的标示 3. 判断事件源上的文本	引入	思考
任务部署 1	1. 判断事件源 <pre> public void actionPerformed(ActionEvent e) { if(e.getSource()==b1){ t.setText("您点击了按钮1"); }else if(e.getSource()==b2){ t.setText("您点击了按钮2"); } } </pre>	讲解 总结	思考
任务部署 2	2. 判断事件源标示 第一步：在事件源上添加标示 <pre> b1.setActionCommand("1"); </pre>	讲解 引导	思考 计算

	<pre>b2.setActionCommand("2"); 第二步：判断事件源获得的标示 public void actionPerformed(ActionEvent e) { if(e.getActionCommand().equals("1")){ t.setText("您点击了按钮1"); }else if(e.getActionCommand().equals("2")){ t.setText("您点击了按钮2"); } }</pre>		
任务部署 3	3.判断事件源上的文本 <pre>public void actionPerformed(ActionEvent e) { JButton b=(JButton)e.getSource(); if(b.getText().equals("Button1")){ t.setText("您点击了按钮1"); }else if(b.getText().equals("Button2")){ t.setText("您点击了按钮2"); } }</pre> 知识复习： 什么是多态？ 多态：父类型的引用可以指向子类型的对象。 比如 Parent p = new Child(); 当使用多态方式调用方法时，首先检查父类中是否有该方法，如果没有，则编译错误；如果有，再去调用子类的该同名方法。 如果想要调用子类中有而父类中没有的方法，需要进行强制类型转换，如上面的例子中，将 p 转换为子类 Child 类型的引用。 因为当用父类的引用指向子类的对象，用父类引用调用方	演示	学习思考

	法时，找不到父类中不存在的方法。这时候需要进行向下的类型转换，将父类引用转换为子类引用。		
练习	有学生编程，练习事件监听的 3 种方式。	辅导	编程

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	在门禁系统上添加事件监听器。 讨论： 1.用哪种方式更好？ 2.讨论程序应该实现什么样的功能？ 3.用 if……else 语句还是用 switch 语句？	引入	思考
任务实施		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	if 语句的语法和应用 switch 语句的语法和应用	总结	听讲 记录
布置作业	练一练：实现监听器的功能，ATM 取款机窗口、计算器窗口，详见作业说明。 本节复习：课堂录像、项目说明文档 作业要求：程序上传至云盘，放在 *组-姓名 文件夹下。		
课后学习资源	下节预告 项目 2.1.我爱大明星 课前储备： 微课《构造方法》		