**第6课 Java Applet**

|  |  |  |
| --- | --- | --- |
| **课 题** | Java Applet | |
| **课 时** | 5课时（225 min）。 | |
| **教学目标** | **知识技能目标：**  1．理解 Applet 的基本概念。  2．掌握 Applet 的执行过程。  **思政育人目标：**  让学生通过学习Java Applet，了解Applet 就是为此服务的，在理解、掌握该技术的前提下，熟练应用该技术，为科教兴国战略出一份力。 | |
| **教学重难点** | **教学重点：**在网页中运行 Java 程序  **教学难点：**在网页中绘图 | |
| **教学方法** | 讲授法、问答法、讨论法 | |
| **教学用具** | 电脑、投影仪、多媒体课件、教材 | |
| **教学设计** | 第1节课：考勤（2min）--知识讲解（40min）--作业布置（3min）  第2节课：知识讲解（40min）--课堂小结（3min）--作业布置（2min）  第3节课：知识讲解（40min）--课堂小结（3min）--作业布置（2min）  第4节课：知识讲解（40min）--课堂小结（3min）--作业布置（2min）  第5节课：知识讲解（40min）--课堂小结（3min）--作业布置（2min） | |
| **教学过程** | **主 要 教 学 内 容 及 步 骤** | **设计意图** |
| **考勤**  **（2min）** | ■【教师】清点上课人数，记录好考勤  ■【学生】班干部报请假人员及原因 | 培养学生的组织纪律性,掌握学生的出勤情况 |
| **知识讲解**  （40min） | **【教师】**展示在网页中运行 Java 程序（一）  **任务描述**  编写 Applet 程序，在界面上输出提示文字。  **任务目标**  1. 能够理解 Applet 的基本概念。  2. 能够掌握 Applet 的执行过程。  3. 能够掌握 Applet 的编写和运行方法。  4. 能够让 Java 代码在网页上“翩翩起舞”。  5. Java Web 是当今流行软件开发模式之一，Applet 就是为此服务的，在理解、掌握该技术的前提下，熟练应用该技术，为科教兴国战略出一份力。  **预备知识**  Applet（小应用程序）是一种特殊的 Java 程序，它只能在支持 Java 的浏览器中运行。  **一、Applet 概述**  Applet 是嵌入在网页中的 Java 程序，必须要在支持 Java 的浏览器中运行。Applet利用了浏览器的资源，当它被载入时，它的外部环境是浏览器预先定义好的执行环境，然后浏览器会根据 HTML 代码中的信息提供 Applet 所需的环境。Applet 程序执行后，可以显示图像、演奏声音等，Applet 就是通过使用这些浏览器所提供的功能来工作的。要运行 Applet，需要在 HTML 文件中加入相应代码。嵌入 Applet 所需的 HTML 代码格式为：  <APPLET  CODEBASE = codebaseURL  CODE = classfilename  ALT = alternatetext  ALIGN = alignment  WIDTH = pixels  HEIGHT = pixels  >  <PARAM NAME = appletattribute VALUE = value>  ...  </APPLET>  其中，各参数的含义如下：  CODEBASE：用 URL 格式指定 Applet 字节码文件所在的位置。如果未指定该参数，则使用 HTML 文件所在的 URL。  CODE：指定 Applet 字节码文件的文件名。该文件的路径是相对于 Applet 的基本URL 的。它不能是绝对 URL。  ALT：指定当浏览器能识别 Applet 标记，但不能运行 Applet 时应显示的正文内容。  ALIGN：指定 Applet 的对齐方式。可能的值包括：left、right、top、texttop、middle、absmiddle、baseline、bottom 和 absbottom。  WIDTH 和 HEIGHT：以像素为单位指定 Applet 显示区域的初始宽度和高度，但不包括 Applet 显示的任何窗口和对话框。  以上参数中，CODE、WIDTH 和 HEIGHT 这三项是必须有的，其他各项都可以不选。因此，<APPLET> 标记最简单的形式为：  <APPLET CODE = \*.class WIDTH = 100 HEIGHT = 100>  </APPLET>  **【学生】**思考、讨论。 | **展示在网页中运行 Java 程序（一），让学生更加仔细的阅读，从而激发学生的学习欲望。** |
| **作业布置**（3min） | **【教师】**布置课后作业  简述Applet的定义。 | 通过课后练习，使学生巩固所学新知识 |
| **知识讲解**  （40min） | **【教师】**展示在网页中运行 Java 程序（二）  **二、Applet 程序的执行过程**  **（一）Applet 程序的执行过程**  Applet 程序的执行过程如图 6−1 所示。    Applet 程序的解释、执行由支持 Java 的 Web 浏览器完成。Applet 程序的字节码文件和 HTML 文件一样，都保存在 Web 服务器中。当用户在浏览器的地址栏中输入一个URL 向 Web 服务器发出请求时，浏览器首先将 HTML 文件下载到客户机，并解释其中的各种 HTML 标记。当遇到 <APPLET>...</APPLET> 标记时，浏览器会根据参数指定的 Applet 程序的地址和文件名，把字节码文件从 Web 服务器上下载到本地客户机上，并由浏览器自身的 Java 解释器直接解释和执行这个字节码文件。所以 Applet 程序的执行过程就是一个网络程序的发布过程，如果 Applet 程序发生变化，只需要对 Web 服务器上的程序进行更新就可以了。  **（二）Applet 程序的生命周期**  Applet 在 Web 浏览器中执行时有自己的生命周期。当 Applet 所在的页面和字节码被浏览器载入后，Applet 就自动产生。此时系统自动调用 Applet 的 init 方法，进行一些必要的初始化操作。接着，系统将调用 start 方法启动 Applet 的执行。当浏览器切换到其他页面时，stop 方法被调用，以终止 Applet 的执行。如果浏览器又切换回 Applet 所在页面，那么系统将重新调用 start 方法开始 Applet 的又一次运行。当浏览器关闭时，系统先调用 stop 方法终止 Applet 的执行，然后调用 destroy 方法来完成资源回收等收尾工作，如图 6−2 所示。    Applet 程序的生命周期中的四个常用方法的功能如下。  （1）init 方法：在 Applet 第一次载入计算机内存时被调用。使用 init 方法，可以完成一些初始化工作。例如：可以初始化变量，加入诸如文本标签、文本框、按钮、复选框等组件到界面中。init 方法在整个 Applet 生命周期中只执行一次。  （2）start 方法：执行完 init 方法后，浏览器将调用 start 方法启动 Applet，另外当浏览器焦点回到 Applet 所在页面时也会调用 start 方法。例如：包含动画的 applet 就需要使用 start 方法播放动画。  （3）stop 方法：当浏览器焦点离开当前页面，或在调用 destory 方法终止 Applet 执行之前将调用 stop 方法。例如：当用户离开网页，applet 已经开始但未完成的任何线程都将继续运行。这时，应该使用 stop 方法挂起这些正在运行的线程，以便不活动的 Applet不再占有系统资源。  （4）destory 方法：关闭 Applet 所在页面时，调用 stop 方法之后，会调用 destory 方法，以释放载入 Applet 时为其分配的系统资源。  **任务实施**  /\* 源程序文件名：Ex0601.java\*/  import java.applet.\*;  import java.awt.\*;  public class Ex0601 extends Applet  {  public void paint(Graphics g)  {  g.drawString(" 这是一个小应用程序 ",5,20);  g.drawString(" 它只能在浏览器中运行 ",10,50);  }  }  我们可以编写下面这样的 HTML 文件：  <HTML>  <HEAD>  小应用程序示例  </TITLE>  </HEAD>  <BODY>  <APPLET CODE = Ex0601.class WIDTH = 400 HEIGHT = 300>  </APPLET>  </BODY>  </HTML>  将这个 HTML 文件放在编译后的 Ex0601.class 文件所在的目录下，双击该 HTML文件即可在浏览器中观察到 Applet 的运行结果。  程序运行结果如图 6−3 所示。    上例中的类 Ex0601 继承于 Applet 类，说明了这是一个小应用程序类，所以需要导入java.applet 包。小应用程序没有 main 方法，而是用 paint 方法执行在界面中绘制文字和图形等一些主要操作，这里需要使用到 Graphics类，所以需要导入 java.awt 包。  小应用程序只能在浏览器中运行，如果使用通常的命令行方式运行，就会有找不到main 函数的提示。在 eclipse 中，通过调用小应用程序查看器，模拟浏览器的运行环境。如果需要将 Applet 放到网页中，需要在网页中加入 HTML 代码，这点将在下面的任务中详细说明。  **【学生】**思考、讨论。 | **通过教师讲解，了解在网页中运行 Java 程序（二）的基本理论知识。** |
| **课堂小结**  （3min） | 【**教师**】**回顾和总结本节课的知识点。**  **这节课我们一起学习了在网页中运行 Java 程序（二），让学生知道Applet 程序的执行过程。** | 通过对所学知识的回顾，培养学生的归纳总结能力 |
| **作业布置**（2min） | **【教师】**布置课后作业  简述Applet 程序的执行过程。 | 通过课后练习，使学生巩固所学新知识 |
| **知识讲解**（45min） | **【教师】**展示在网页中绘图（一）  **任务描述**  编写 Applet 程序，绘制如图 6-4 所示的图形。    **任务目标**  1. 能够掌握在 Applet 中绘制图形的方法。  2. 能够掌握 paint 和 repaint 方法的使用。  3. 能够掌握 Graphics、Color 和 Font 类的使用。  4. 能够在 Applet 中画画，做一个能在浏览器描绘“美好蓝图”的人。  **预备知识**  Applet 中除了可以显示字符外，还可以绘制图形。  **一、paint 和 repaint 方法**  paint 方法是在 Applet 小程序首次被载入，以及每次窗口缩放和刷新时被浏览器自动调用。该方法有一个 Graphics 类参数，可使用这个参数所提供的方法在 Applet 中绘制各种图形。  paint 方法是自动调用而不是由程序员调用的，当程序员希望更新界面内容时，使用repaint 命令调用 paint 方法。  **二、Graphics 类**  Graphics 类提供了绘制基本图形的方法，包括画直线、矩形、椭圆、弧形和多边形。  Graphics 类中的常用方法见表 6−1。    下面的代码用于绘制多个图形，大家可以自行试验：  int xPoint[] = {30,63,115,72,67};  int yPoint[] = {40,20,95,74,106};  int nPoints = xPoint.length;  g.drawPolygon(xPoint, yPoint, nPoints);  g.drawArc(110,20,100,60,35,-140);  g.fillArc(310,20,100,60,35,140);  **【学生】**思考、讨论。 | **教师通过在网页中绘图（一）展示，让学生了解在网页中绘图的基本理论知识。** |
| **课堂小结**  （3min） | 【**教师**】**回顾和总结本节课的知识点。**  **这节课我们一起学习了在网页中绘图（一），能够在 Applet 中画画，做一个能在浏览器描绘“美好蓝图”的人。** | 通过对所学知识的回顾，培养学生的归纳总结能力 |
| **作业布置**（2min） | 【**教师**】**布置课后作业**  **简述paint 和 repaint 方法。** | 通过课后练习，使学生巩固所学新知识 |
| **知识讲解**（45min） | **【教师】**展示在网页中绘图（二）  **三、Color 类**  Color 类用于定义颜色。在 Applet 中绘制文字和图形时，默认颜色都是黑色。我们可以调用 Graphics 类提供的 setColor 方法重新指定当前的绘图颜色，用 getColor 方法获取当前所使用的颜色。  setColor 方法的参数是一个 Color 类对象。Java 中的每一种颜色都被看成是由红（R）、绿（G）、蓝（B）三基色调和而成的，因此 Color 类的构造函数采用如下格式：  Color(int r, int g, int b)  其中，每个参数的值都在 0 到 255 之间，数值越大表明这种颜色的成分越重。如（0，0，0）代表黑色，（255，255，255）代表白色，而（255，0，0）代表红色。另外，Color 类提供了几个静态成员变量，用于表示几种标准颜色。如 Color.blue 表示蓝色，Color.red 表示红色，Color.black 表示黑色等。  如果使用 setColor 方法设置了颜色，它将一直作为当前颜色，除非再次使用setColor 方法设置其他颜色。  下面的代码用于绘制一排用随机颜色填充的小方块：  int r, gr, b;  for(int i = 10; i<400; i+=40)  {  r = (int)(Math.random()\*256);  g = (int)(Math.random()\*256);  b = (int)(Math.random()\*256);  gr.setColor(new Color(r,g,b));  gr.fillRect(i,20,30,30);  }  **四、Font 类**  Font 类用于定义字体格式。它的构造函数格式为：  Font(String name, int style, int size)  参数中的 name 是以字符串格式给出的字体名称，如宋体、楷体、隶书、华文彩云等，如果系统中没有安装指定字体，则使用默认字体显示。Font 类的 getFontName 方法可以得到当前所使用的字体名称。  参数 style 指定字体风格，用 Font 类的一些静态成员变量来表示。如 Font.BOLD 表示粗体，Font.ITALIC 表示斜体，Font.PLAIN 表示普通体。如果需要定义字体既为粗体又为斜体，则只要定义 style 为 Font.BOLD + Font.ITALIC 就可以了。Font 类的 getStyle方法可以得到当前字体风格。参数 size 为字体大小，即字高，以像素为单位。Font 类的getSize 方法可以得到当前字体的字高。  Graphics 类的 setFont 方法用于设置当前使用的字体。与 setColor 方法一样，如果使用 setFont 方法设置了字体，它将一直作为当前字体，除非再次使用 setFont 方法重新设置其他字体。  下面的代码用于在界面上绘制一些不同字体的文字：  Font ftp20 = new Font("TimeRoman",Font.PLAIN,20);  Font fai15 = new Font("Arial",Font.ITALIC,15);  Font fcb24 = new Font("Courier",Font.BOLD,24);  Font fsib30 = new Font(" 宋体 ",Font.ITALIC+Font.BOLD, 30);  g.setFont(ftp20);  g.drawString(" Font name TimeRoman, Style plain, Size 20",10,20);  g.setFont(fai15);  g.drawString("Font name Arial, Style italic, Size 15");  g.setFont(fcb24);  g.drawString("Font name Courier, Style bold, Size 24");  g.setFont(fsib30);  g.drawString(" 字体名 宋体 , 风格 斜体＋粗体 , 尺寸 30");  **任务实施**  /\* 源程序文件名：Ex0602.java\*/  import java.applet.\*;  import java.awt.\*;  public class Ex0602 extends Applet  {  public void paint(Graphics g)  {  g.drawString("init()",60,80);  g.drawString("start()",135,80);  g.drawString("stop()",215,80);  g.drawString("destory()",290,80);  g.drawString(" 进入或重写页面 ",15,60);  g.drawString(" 开始执行 ",125,60);  g.drawString(" 关闭页面 ",240,60);  g.drawString("stop()",135,130);  g.drawString(" 离开页面 ",80,110);  g.drawString(" 返回页面 ",175,110);  g.setColor(Color.blue);  Font titleFont = new Font(" 隶书 ",Font.BOLD,22);  g.setFont(titleFont);  g.drawString(" 图 6-4 Applet 程序的生命周期 ",20,40);  g.drawOval(45,65,50,20);  g.drawRect(125,65,50,20);  g.drawRect(205,65,50,20);  g.drawOval(285,65,55,20);  g.drawRect(125,115,50,20);  g.setColor(Color.red);  g.drawLine(15,75,45,75);  g.drawLine(95,75,125,75);  g.drawLine(175,75,205,75);  g.drawLine(255,75,285,75);  g.drawLine(130,85,130,115);  g.drawLine(170,85,170,115);  }  }  在这个例子中，使用了多种方法在 Applet 中绘制各种文字和图形。除绘制文字的drawString 方法外，还有 drawLine 方法用于绘制直线，drawRect 方法用于绘制矩形，drawOval 方法用于绘制椭圆。另外，还使用了 setFont 方法以改变输出文字的格式，用setColor 方法改变输出文字和图形的颜色。这些方法的具体使用，将在下面的任务中详细介绍。  **【学生】**思考、讨论。 | **教师通过在网页中绘图（二）展示，让学生了解在网页中绘图（二）的基本理论知识。** |
| **课堂小结**  （3min） | 【**教师**】**回顾和总结本节课的知识点。**  **这节课我们一起学习了在网页中绘图（二），了解参数中的 name 是以字符串格式给出的字体名称。** | 通过对所学知识的回顾，培养学生的归纳总结能力 |
| **作业布置**（2min） | 【**教师**】**布置课后作业**  **简述Font 类。** | 通过课后练习，使学生巩固所学新知识 |
| **知识讲解**（45min） | **【教师】**展示思考与练习  **一、简答题**  （1）Applet 程序和 Application 程序的主要区别是什么？  （2）简述 Applet 程序的执行过程。  （3）在将 Applet 程序嵌入到 HTML 文件时需要加入哪些 HTML 代码？  **二、选择题**  （1）下面关于 Applet 的说法正确的是（　　）。  A. Applet 也需要 main 方法  B. Applet 必须继承 Java.awt.Applet  C. Applet 能访问本地文件  D. Applet 程序不需要编译  （2）下面说法正确的是（　　）。  A. Applet 类需要从 Java 的一个包中载入  B. Applet 不能有 main 方法  C. Applet 不是 Object 的子类  D. Applet 是 Java 语言的关键字  （3）当浏览器从另一个 URL 返回包含一个 Applet 对象的页面时，应调用方法（　　）。  A. init( ) B. start( ) C. stop( ) D. destropy( )  （4）下面方法与 Applet 的显示无关的是（　　）。  A. update( ) B. draw( ) C. repaint( ) D. paint( )  **实训任务**  （1）编写 Applet 程序，在界面上绘制如图 6−5 所示的图形。    （2）仿照任务 2，编写 Applet 程序，输出如图 6−1 所示的图形。  （3）编写 Applet 程序，在界面上绘制如图 6−6 所示的图形。    **【学生】**思考、讨论。 | **教师通过思考与练习展示，让学生了解思考与练习的基本理论知识。** |
| **课堂小结**  （3min） | 【**教师**】**回顾和总结本节课的知识点。**  **这节课我们一起学习了思考与练习，让学生能够更好的巩固学习知识。** | 通过对所学知识的回顾，培养学生的归纳总结能力 |
| **作业布置**（2min） | 【**教师**】**布置课后作业**  **怎样在 Applet 中绘图？** | 通过课后练习，使学生巩固所学新知识 |
| **教学反思** | 教师把经验、方法讲给学生听了，不等于学生就获得了这个经验、方法，必须要让学生及时的、有针对性的操作、练习去进行巩固。 | |