

声明：百科词条人人可编辑，词条创建和修改均免费，绝不存在官方及代理商付费代编，请勿上当受骗。 [详情>>](#)

AWT

编辑

AWT(Abstract Window Toolkit)，中文译为抽象窗口工具包，该包提供了一套与本地图形界面进行交互的接口，是Java提供的用来建立和设置Java的图形用户界面的基本工具。AWT中的图形函数与操作系统所提供的图形函数之间有着一一对应的关系，称之为peers，当利用AWT编写图形用户界面时，实际上是在利用本地操作系统所提供的图形库。由于不同 操作系统的图形库所提供的样式和功能是不一样的，在一个平台上存在的功能在另一个平台上则可能不存在。为了实现Java语言所宣称的“一次编写，到处运行(write once, run anywhere)”的概念，AWT不得不通过牺牲功能来实现平台无关性，也即AWT所提供的图形功能是各种操作系统所提供的图形功能的交集。

中文名

抽象窗口工具包

缩 写

AWT

外文名

Abstract Window Toolkit

编程语言

Java

功 能

图形用户界面(GUI)工具包

目录

1 术语名称

2 术语解释

区别

优势

3 层次结构

层次

独立

4 使用方法

组件

容器

布局管理器

实例

术语名称

编辑

抽象窗口工具箱（AWT）

术语解释

编辑

抽象窗口工具包（Abstract Windowing Toolkit）（AWT）是Java的平台独立的窗口系统，图形和用户界面器件工具包。AWT是Java基础类（JFC）的一部分，为Java程序提供图形用户界面(GUI)的标准API。

AWT提供了JavaApplet和Java Application中可用的用户图形界面GUI中的基本组件(components)。由于Java是一种独立于平台的程序设计语言，但GUI却往往是依赖于特定平台的，Java采用了相应的技术使得AWT能提供给应用程序独立于机器平台的接口，这保证了同一程序的GUI在不同机器上运行具有类似的外观（不一定完全一致）。

Java1.0的AWT(旧AWT)和Java1.1以后的AWT(新AWT)有着很大的区别，新的AWT克服了旧AWT的很多缺点，在设计上有较大改进，使用也更方便，这里主要介绍新的AWT, 但在Java1.1及以后版本中旧的AWT的程序也可运行。

Abstract Window Toolkit（AWT）是一个图形过程库，使用Java语言通过位图显示来操纵窗口。最后设计者又将AWT扩充为Alternative Window工具箱和Applet Widget工具箱。最新的GUI接口称为Swing，扩展了AWT，程序开发人员可以利用Swing生成独立于平台的GUI对象。^[1]

区别

抽象窗口工具包AWT（AbstractWindow Toolkit）是 API为Java 程序提供的建立图形用户界面GUI（Graphics User Interface）工具集，AWT可用于Java的applet和applications中。它支持图形用户界面编程的功能包括： 用户界面组件；事件处理模型；图形和图像工具，包括形状、颜色和字体类；布局管理器，可以进行灵活的窗口布局而与特定窗口的尺寸和屏幕分辨率无关；数据传输类，可以通过本地平台的剪贴板来进行剪切和粘贴。

优势

Java释出的时候，AWT作为Java最弱的组件受到不小的批评。最根本的缺点是AWT在原生的用户界面之上仅提供了一个非常薄的抽象层。例如，生成一个AWT的复选框会导致AWT直接调用下层原生例程来生成一个复选框。不幸的是，一个Windows平台上的复选框同MacOS平台或者各种UNIX风格平台上的复选框并不是那么相同。这种糟糕的设计选择使得那些拥护Java“一次编

Component

Container

Window

Frame

Dialog

FileDialog

Panel

Button

Label

Checkbox

TextComponent

TextArea

AWT图册

V百科

往期

Ai

秀

Ps

图解百科 | 第630期

这些美图软件，你真的会用了

分享

相关人物

松本行弘

黎活明

毕向东

张孝祥

相关术语

swing

科学可视化

gui

序列化

泛型

java.s

计算机类书籍

java项目开发...

javaswing...

java网络

http://baike.baidu.com/link?url=IMvR7ZfI_No_RzGJF4vdABWGZddLsLLCbJ_yNfUwUH8O_0uX7xUvIYue4KuyJsJLQmeEEhJmqDF4edY36KBcStq

1/5

		设置为Frame对象frm的主菜单
下拉菜单(Menu)	Menu=newMenu("Menu");	通过mb.add(m)可将m下拉列表添加到菜单栏中
菜单项(MenuItem)	MenuItem=newMenuItem("Openit");	通过mi.add(mi)将菜单项mi添加到下一个下拉列表m中。
颜色(Color)	//非构造方法，两种方法设置组件背景色 setBackground(Color.BLUE); setBackground(newColor(100,100,100));	通过setForeground()方法设置组件前景色，通过setBackground()方法设置组件背景色。颜色值可使用Color类的颜色常量来设置，也可通过指定RGB颜色对象值设置
字体(Font)	Font=newFont("宋体",Font.BOLD,10); //宋体，10号，加粗	按顺序向Font()构造方法传递有关字体设置的参数

容器

容器(Container)也是一个类，实际上是Component的子类，因此容器本身也是一个组件，具有组件的所有性质，但是它的主要功能是容纳其它组件和容器。容器可以简化图形化界面的设计，以整体结构来布置界面。所有的容器都可以通过add()方法向容器中添加组件。常用的容器3种：Panel, Frame, Applet。^[2]

1. 框架(Frame)

构造方法：

```
1 | Frame frm = new Frame("New Window");
```

要生成一个窗口，通常使用Window的子类Frame类进行实例化，而不是直接使用Window 类，框架的外观就像平常Windows系统下的窗口，有标题、边框、菜单和大小等。setSize()方法可以设置框架尺寸的大小，setVisible()方法可以设置窗口的可见性。

1. 面板(Panel)

构造方法：

```
1 | Panel pnl = new Panel();
```

面板是一种透明的容器，没有标题和边框。与Frame 不同，Panel不能作为最外层的容器单独存在，它首先必须作为一个组件放置在其他容器（一般为Frame）中，然后把组件添加到它里面。在java.swing中则是JPanel。

1. 小程序(Applet)

构造方法：

```
1 | Applet apt = new Applet();
```

框架一般用作Java应用程序的窗口，而Applet是Java小程序的窗口。与Frame不同，Applet是在网页中显示的，也可以通过添加Panel进行组件布局。

布局管理器

布局管理器（LayoutManager）：每个容器都有一个布局管理器，当容器需要对某个组件进行定位或判断其大小、尺寸时，就会调用其对应的布局管理器。使用布局管理器可以实现跨平台的特性，并且获得动态的布局效果。布局管理器负责管理组件的排列顺序、大小和位置。不同的布局管理器使用不同的布局策略，容器可以通过选择不同的布局管理器来决定如何布局。^[2]

流布局管理器(FlowLayout)

FlowLayout是Panel 和 Applet 的默认布局管理器。在该布局管理器中，组件在容器中按照从上到下，从左到右的顺序进行排列，行满后则换行。

构造方法较多，例如：

- FlowLayout();
默认构造方法，设置居中对齐方式，横向和纵向间隔均为默认5像素
- FlowLayout(FlowLayout.LEFT);
设置居左对齐，横向和纵向间隔均为默认5像素
- FlowLayout(FlowLayout.RIGHT,20,40);
设置居右对齐，横向间隔20像素，纵向间隔40像素

边界布局管理器(BorderLayout)

构造方法：

```
1 | BorderLayout bl = new BorderLayout();
```

BorderLayout是Window、Frame和Dialog的默认布局管理器，其将容器分成North、South、East、West和Center 5个区域，每个区域只能放置一个组件。在使用add()方法添加组件到容器时，必须指定将其放置在哪个区域中。使用BorderLayout

分享

参考资料

1. Abstract Window Toolkit . TechTarget[引用日期2015-08-23]
2. 希赛教育等考学院 卢艳芝. 二级Java语言程序设计. 北京：机械工业出版社，2014：149-156

词条标签：科技产品，科学，学科，互联网

猜你喜欢

- java基础教程
- java教学
- java怎么学习
- 学习java怎么样
- java课程
- java怎么学
- 微信营销
- 耐酸砖
- 58同城网租房子
- 星际战甲官网

新手上路

- 成长任务
- 编辑入门
- 编辑规则
- 百科术语

我有疑问

- 我要质疑
- 我要提问
- 参加讨论
- 意见反馈

投诉建议

- 举报不良信息
- 未通过词条申诉
- 投诉侵权信息
- 封禁查询与解封

分享

★

👁

💬

🐼