

Java软件设计基础





JAVA™

1. Java概述

课程简介

- 课程参考书目

- 《Java教程》，Sharon Zakhour，人民邮电出版社
- 《Introduction to Java Programming》，Y.Daniel Liang，机械工业出版社
- 《Java编程思想》，Bruce Eckel，机械工业出版社

- 课程安排

- 平时成绩：形式为程序类作业；
- 考试成绩：考试形式为笔试。

- 其他信息

- Email: renkx@ustc.edu.cn
- 课件下载地址：学校个人邮箱网盘

1.1 关于Java

- Java起源与发展
 - 1991年由美国Sun公司开发的具有平台无关性的语言——Oak。1992年10月更名为Java。Java和Internet迅速融合并互相推动快速发展，Java语言已由最初的版本发展到今天的10。
- Java的技术现状
 - J2SE: Java 2 Platform Standard Edition (Java标准版)
 - 包含构成Java语言核心的类。
 - J2EE: Java 2 Platform Enterprise Edition (Java企业版)
 - 包括J2SE中的核心类以及用于开发企业级应用的类。
 - J2ME: Java 2 Platform Micro Edition (Java企业版)
 - 用于消费类电子产品的软件开发。

1.2 Java与其他语言的对比

C#

- Microsoft发布
- C#与Java都是传统面向对象程序设计在组件化软件时代的革新成果，两种编程语言甚至有“90%的重叠”
 - 都对传统C/C++语法规义进行了改进；
 - 都摒弃了函数、全局变量、全局函数等地方；
 - 都采用了单继承多接口实现方案；
 - 都提出了声明与实现于一体的逻辑封装……

提供了Windows任务相关的大量基类，更善于利用Windows平台

Ruby

- 松本行弘研发。是一种简洁的解释型语言，也是跨平台语言。
- 2004年，Ruby语言阵营中出现了优秀的MVC框架——Ruby On Rails，是真正意义上的敏捷开发框架，提供了大量代码生成器，提高应用开发效率。
- Rails平台的简洁性、易用性会在中小型应用上发挥更大吸引力。

Python

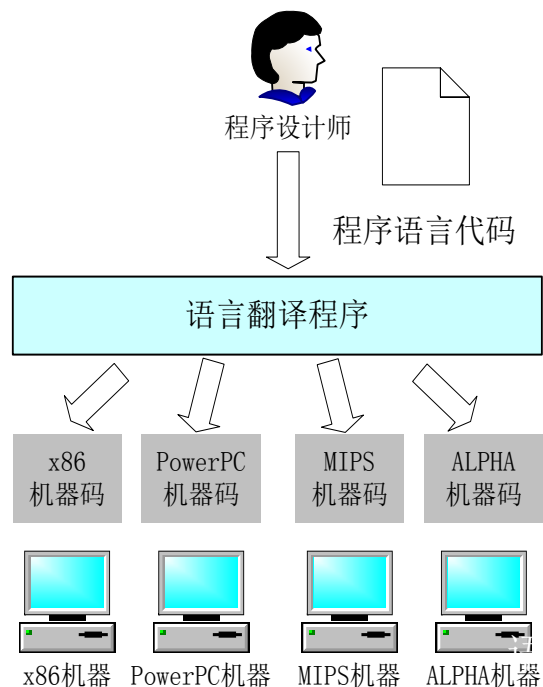
- Guido于1989年开发。面向对象的解释型语言，具有二十年的发展历史，成熟稳定。
- 具有脚本语言中最丰富和强大的类库，足以支持绝大多数日常应用。
- 非格式自由的语言。



- 97% 的企业桌面运行 Java
- 美国有 89% 的桌面（或计算机）运行 Java
- 全球有 900 万 Java 开发人员
- 开发人员的头号选择
- 排名第一的部署平台
- 有 30 亿部移动电话运行 Java
- 100% 的蓝光盘播放器附带了 Java
- 有 50 亿张 Java 卡在使用
- 1.25 亿台 TV 设备运行 Java
- 前 5 个原始设备制造商均提供了 Java ME

1.3 Java运作机制

- 编译型语言翻译的一般过程
 - 使用专门的编译器，针对特定平台将某种高级语言源代码一次性翻译成可被该平台执行的机器码，并包装成该平台可识别的可执行代码



C、C++.....

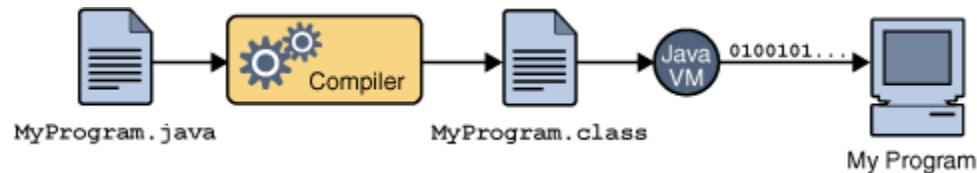
- 解释型语言的一般过程

- 使用专门的解释器对源程序**逐行**解释成特定平台的机器码并立即执行。
- 解释型语言通常不会进行整体性的编译和链接处理，而是将编译和解释过程混合到一起完成。

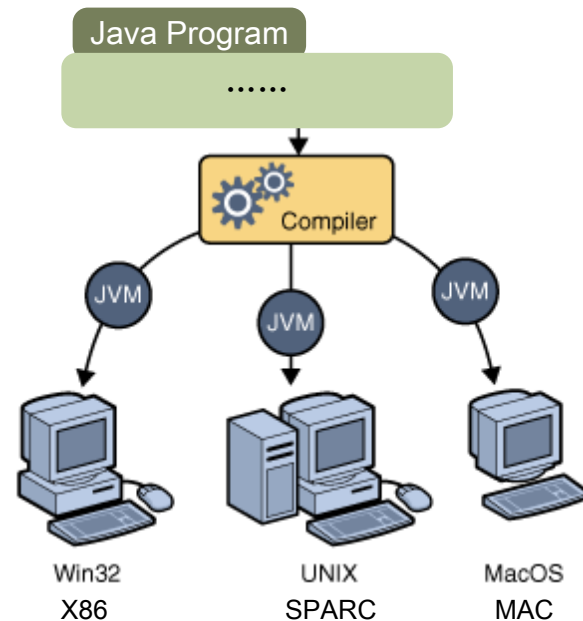
Ruby、Python.....

- Java的运行机制——中间机制

- 经过编译器生成一种与平台无关的字节码（*.class文件），再使用Java解释器来解释执行。



- Java虚拟机 (JVM: Java Virtual Machine)
 - JVM可以在任何处理器上安全、兼容地执行Java应用程序。
 - JVM可建立于不同的硬件系统、不同的操作系统平台。
 - 字节码文件由JVM转换成某种处理器的专用代码，因此可以达到“一次编写，处处运行”的理想。



1.4 Java开发与运行平台

- JDK系统的安装
 - 获取
 - 可在ORACLE官网上下载或其他互联网资源
 - 主要文件
 - Java编译器：javac.exe
 - 用于将Java源代码文件 (*.java) 编译成能被Java解释器运行的类字节码文件 (*.class) 。
 - Java字节码解释器，即JVM：java.exe
 - 用于Java程序的解释执行，直接从类字节码文件执行Java应用程序。
 - JDK由开发和测试Java程序的独立程序组成，从命令行调用。

- 第一个Java Application

- 编写Java源程序

FirstJava.java

```
//这是我们的第一个Java Application, 该程序保存在文件FirstJava.java中
public class FirstJava
{
    public static void main(String args[])
    {
        System.out.println("您好, 我的第一个Java程序!");
        System.out.println("Hello Java!!");
    }
}
```

- 编译和运行Java源程序

- 使用javac.exe来编译, 成功以后得到class文件;
 - java.exe来运行class文件。

- 类是Java的基本组成元素，所有的Java程序一定要被类管理

- 类定义最基本的形式是：

类定义

```
class ClassName
{
    .....
}
```

- 大括号括起的内容就是类体，即类的属性和方法。
- 主类：用public修饰的类，一个Java程序文件只能、必须有一个主类，并且类名与文件名一致
- main方法
 - main方法是应用程序的入口点，程序代码由此开始执行，位于主类中
 - main方法接受单一参数：String类型的元素的数组，该数组是一种机制，运行时系统通过它把消息传递给应用程序。数组中的每个字符串称为命令行参数（command-line argument）。