

JAVA 语言基础

DAY01

Java+大数据 课程体系

Java+大数据课程体系简介

阶段一：Java基础工程师(入门)

实现目标：打造入门级Java软件工程师，并培养出基本的软件开发能力。

实战项目：兄弟连在线考试系统

1、JAVA设计和编程基础(OOD和OOP)

DAY01	Java 语言基础	Java 语言、Java环境变量、变量、运算符、表达式、分支语句、循环语句、数组，数组应用，行业规范。双色球抽奖程序实现。
DAY02		
DAY03		
DAY04		
DAY05		
DAY06	Java面向对象	类、对象、属性、方法、构造、封装、继承、多态、重写、重载、访问权限控制符、this和super、static、单例设计模式、final、抽象类和接口、模板设计模式、内部类
DAY07		
DAY08		
DAY09		
DAY10		
DAY11	JavaSE核心类库	Java核心类、Java字符串、日期处理、包装类、集合、数据结构、异常和异常处理、Java IO、多线程编程、线程同步机制、并发包、JAVA网络编程、Java泛型、Java反射机制。
DAY12		
DAY13		
DAY14		
DAY15		
DAY16		
DAY17		
DAY18		
DAY19		
DAY20	Java设计	设计原则、设计模式、常见算法、Java新特性

学习资料和流程(手册)

计算机体系结构简介

计算机的基本概念

- 计算机(Computer)俗称电脑，是现代一种用于高级算计，使用非常广泛的设备，主要由计算机硬件和计算机软件两个部分组成。
- 计算机硬件是客观存在的各种计算机相关设备，而计算机软件是用于控制各种硬件设备完成各种功能。

常见的主要硬件

7

- 计算机硬件(Computer Hardware)主要包括：中央处理器(CPU)、内存、硬盘、输入输出设备、主板、机箱和电源等辅助设备。



常见的主要软件

- 计算机软件(Computer Software)可分为系统软件和应用软件，系统软件就是操作系统，是其他软件的基础。



Linux 操作系统简介

- 类Unix系统
- 起源于1991年
- 1995年随着互联网流行开
- 主流的互联网操作系统
- 应用广泛: 云计算, Java, Android
- Linux是主流的Java部署环境



计算机体系结构

应用程序/JAVA 应用
Application

操作系统外壳(GUI/命令)
OS Shell(KDE bash csh)

操作系统内核
OS Kernel

硬件
Hardware

Java语言的基本概述

Java发展历史



- 1991年 Sun Green。
- 1992年 James Gosling Oak。
- 1995年 JAVA问世。
- 1996年 JDK 1.0。
- 1999年 Java 2发布 (JAVA SE\JAVA EE\JAVA ME) 。
-
- 2004年 Tiger 发布(JAVA5.0), Java 登录火星。
- 2007年JDK6.0。
- 2009年Oracle以超过70亿美元的交易总值收购了Sun。
- 2011年7月由Oracle正式发布Java7 。
- 2014年3月19日, 甲骨文公司发布Java 8.0的正式版。

Java的江湖地位

Jun 2017	Jun 2016	Change	Programming Language	Ratings	Change
1	1		Java	14.493%	-6.30%
2	2		C	6.848%	-5.53%
3	3		C++	5.723%	-0.48%
4	4		Python	4.333%	+0.43%
5	5		C#	3.530%	-0.26%
6	9	▲	Visual Basic .NET	3.111%	+0.76%
7	7		JavaScript	3.025%	+0.44%
8	6	▼	PHP	2.774%	-0.45%
9	8	▼	Perl	2.309%	-0.09%
10	12	▲	Assembly language	2.252%	+0.13%
11	10	▼	Ruby	2.222%	-0.11%
12	14	▲	Swift	2.209%	+0.38%
13	13		Delphi/Object Pascal	2.158%	+0.22%
14	16	▲	R	2.150%	+0.61%
15	45	▲	Go	2.044%	+1.83%
16	11	▼	Visual Basic	2.011%	-0.24%
17	17		MATLAB	1.996%	+0.55%
18	15	▼	Objective-C	1.957%	+0.25%
19	22	▲	Scratch	1.710%	+0.76%
20	18	▼	PL/SQL	1.566%	+0.22%

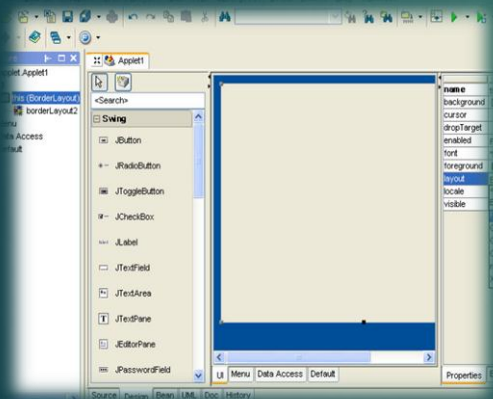
点击查看原始大小图片

Java SE 简介

桌面应用

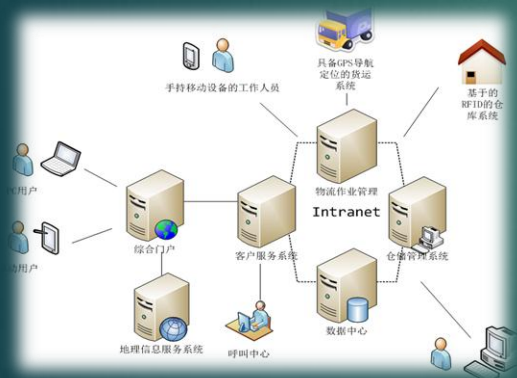
- Java SE (Java Platform, Standard Edition) 称之为“Java平台标准版”，是Java平台的基础。

- Java SE包含了运行Java应用所需要的基础环境和核心类库。
- Java SE还定义了基于桌面应用的基础类库，通过使用这些类库，我们可以编写出类似于像Office那样的丰富多彩的桌面应用。



Java EE 简介

企业应用



Java EE (Java Platform, Enterprise Edition) 称之为“Java平台企业版”。Java EE构建在Java SE基础之上，用于构建企业级应用。所谓企业级应用是指那些为商业组织、大型企业而创建的应用系统，例如：电信的“计费系统”、银行的“网银系统”、企业中的“客户关系管理系统”等等。

Java ME 和 Android

智能终端
移动互联

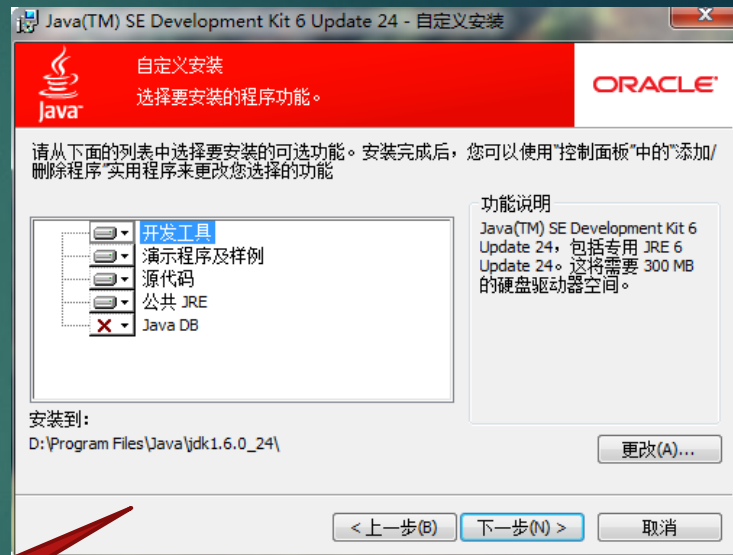


- **Java ME (Java Platform, Micro Edition)** 称之为Java平台微型版。为机顶盒、移动电话和PDA等嵌入式消费电子设备提供的Java解决方案。随着3G移动智能平台时代的到来，特别是以Java为核心编程语言的Android智能平台的迅速普及，Java ME已经走向淘汰。

Java 开发环境的搭建

JDK 下载与安装

- <http://java.sun.com>
- <http://www.oracle.com>
- Java 6/Java 7 都可以使用
- OS版本和JDK版本要对应
X86--32位版本
X64--64位版本
- 注意64位版本不能安装到32位系统上



安装界面

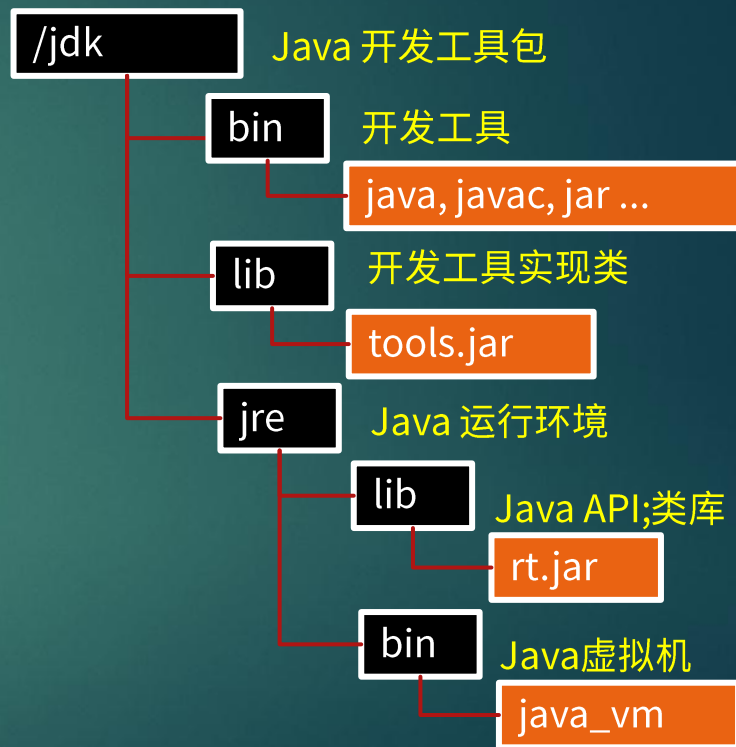
JDK 介绍

JRE (Java SE Runtime Environment) 称之为Java SE运行时环境，提供了运行Java应用程序所必须的软件环境，包含有Java虚拟机 (JVM) 和丰富的类库 (Libraries)。无论是开发Java应用还是仅仅运行一个已经开发好的Java应用都必须安装JRE。

JDK (Java Development Kit) 称为Java开发工具包，是JRE的超集，或者说JDK包含了JRE。JDK中除了包含有JRE的所有内容之外还提供了编写Java程序所必须的编译器和调试工具等。对于编写Java开发的人士一般需要下载JDK，目前的主流版本为JDK 6。

JDK目录结构

- JDK – Java Development Kit (Java开发工具包Development)
- Tools 开发工具
 - java javac jar javadoc
- JRE – Java Runtime Environment (Java运行环境)
- Java API类库
 - 是Java的系统标准类库
- JVM – Java Virtual Machines (Java虚拟机)



配置环境变量—更改PATH

- JDK 的bin中的javac命令和java命令是我们很快要用到的命令。
- 如果想用到这些命令，需要让操作系统可以找到这些命令文件所在的路径。可以通过配置PATH环境变量来实现。
- PATH环境变量是一系列的目录，在执行命令时，操作系统会依次在PATH环境变量中的每一个目录中查找该命令，如果找到就可以执行，否则就会错误提示

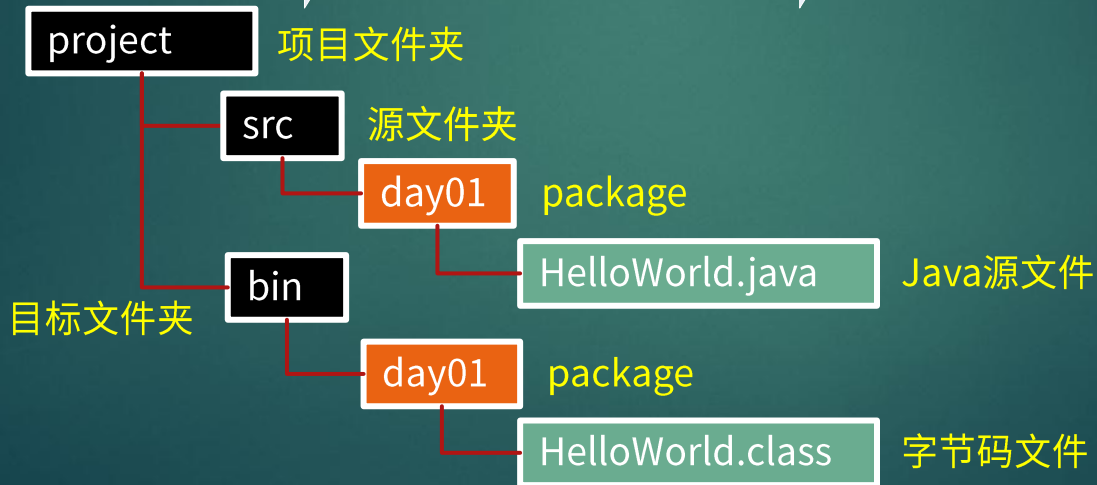
`PATH=/opt/jdk/bin:$PATH` windows在系统属性中设置
`export PATH` Linux 在 /etc/profile中设置

Java HelloWorld

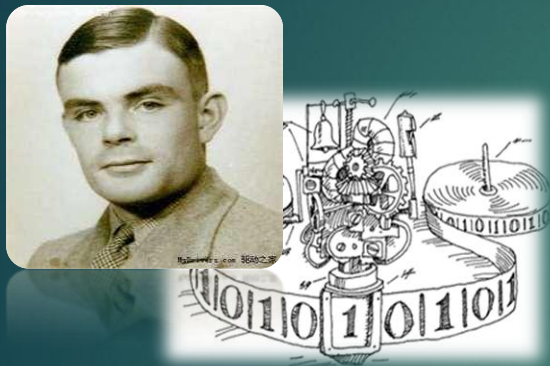
HelloWorld.java

```
/** 第一个Java 程序 */  
public class HelloWorld {  
    /**  
     * 软件入口方法  
     */  
    public static void main(String[] args) {  
        // 输出 Hello World!  
        System.out.println("Hello World!");  
    }  
}
```

Java的编译与运行



计算机程序运行原理

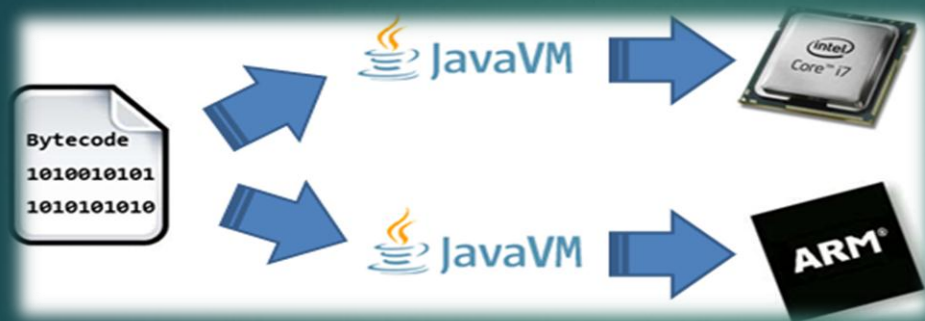


阿兰·图灵与图灵
机模型
程序: 指令(机器码)
和数据有序序列



通常的高级语言，比如
C/C++，需要通过编译、连
接后才可以转换为在特定系
统平台上执行的机器码，而
机器码是平台相关的

Java 跨平台原理



Java字节码可以通过JVM翻译为具体平台能够执行的机器指令。由于Sun定义了JVM规范，而且不同的操作系统大多提供了JVM实现，才使得相同的一个字节码文件可以在不同的系统上运行，从而使Java赢得了“一次编译，到处使用”的美名。

- 编程规范不是语法，是业内约定俗成的规则，程序员的潜规则。
- Java的编程规范如下：
 - 1 每个部分之间用空行隔开。
 - 2 要有空格和缩进。
 - 3 所有类名要首字母大写，其它单词首字母也大写。
 - 4 变量名和方法名首字母小写，其它单词首字母大写。
 - 5 包名全小写，不同单词用点或者下划线隔开。
 - 6 常量所有字母都大写，不同单词之间用下划线隔开。
 - 7 最好望文知意，尽量不要使用中文。

Java API 的使用

- JDK中带有大量的API类, 是有Java系统带来的工具库, 这些工具来数以万计! 是Java官方程序员的技术积累.
如: `java.lang.String`, `java.lang.System`
- 这里类存储在 `JAVA_HOME/jre/lib/rt.jar` 等文件中
如: `java.lang.String.class` `java.lang.System.class`
- 这些类可以大大简化编程, 提高开发效率
- 使用API类要用import语句导入类
如: `import java.util.Scanner`
- 具体的API类功能, 可以参阅Java的参考手册

控制台信息的读取

- 利用Scanner类可以便捷的从控制台读取数据

```
import java.util.Scanner;
public class HelloWorld {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.println("请输入你的姓名：");
        String name = scanner.next();
        System.out.println(name + ", 欢迎你进入Java的世界");
    }
}
```

今日重点：

- 1、环境变量的配置
- 2、JAVA跨平台的原理
- 3、JAVA的HelloWorld.java