

Java 应用程序设计

Java 概论

KevinW@OUC

wangxiaodong@ouc.edu.cn

中国海洋大学

September 21, 2017



参考书目

1. 张利国、刘伟 [编著], Java SE 应用程序设计, 北京理工大学出版社, 2007.10.
2. Bruce Eckel, Thinking in Java (3rd)



本章学习目标

1. 从 Java 的视角回顾 OOP
2. 了解 Java 的发展历程
3. 理解 Java 平台的相关概念和机制
4. 掌握基本 Java 开发环境配置



大纲

Java 技术概述

Java 平台核心机制

Java 开发环境

Java 基本开发流程



接下来...

Java 技术概述

Java 平台核心机制

Java 开发环境

Java 基本开发流程



LOGOs



ORACLE®



Sun 公司大事记

- 1982 Sun 公司成立 (安迪·贝托谢姆和麦克尼利)。
- 1986 Sun 公司上市。
- 1985 Sun 公司推出著名的 Java 语言。
- 2011 9.11 事件前，Sun 市值超过 1000 亿美元；此后，由于互联网泡沫的破碎，其市值在一个月内跌幅超过 90%。
- 2004 Sun 公司和微软在旷日持久的 Java 官司中和解，后者支付前者高达 10 亿美元的补偿费。
- 2006 共同创始人麦克尼利辞去 CEO 一职，舒瓦茨担任 CEO 后尝试将 Sun 从设备公司向软件服务型公司转型，但不成功。
- 2010 Sun 公司被甲骨文公司收购。



Java 发展简史

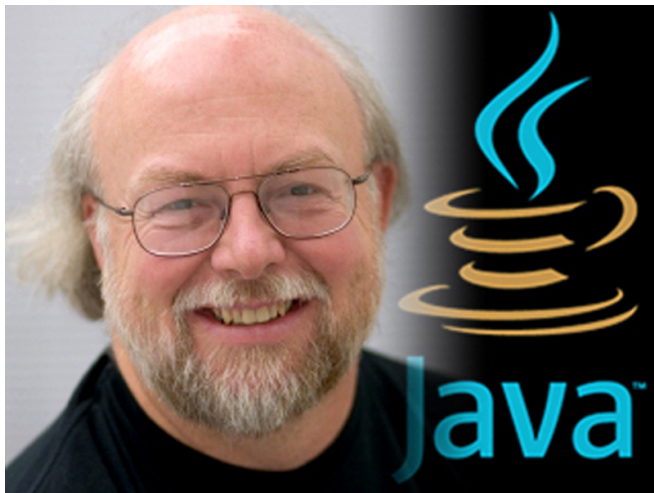


图: Java 之父詹姆斯·高斯林 (James Gosling)



Java 发展简史



接下来...

Java 技术概述

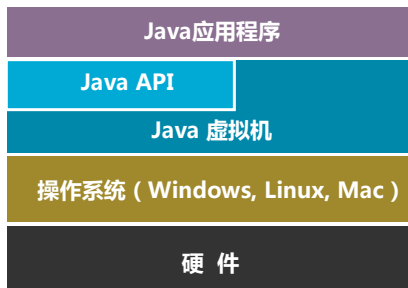
Java 平台核心机制

Java 开发环境

Java 基本开发流程



Java 平台

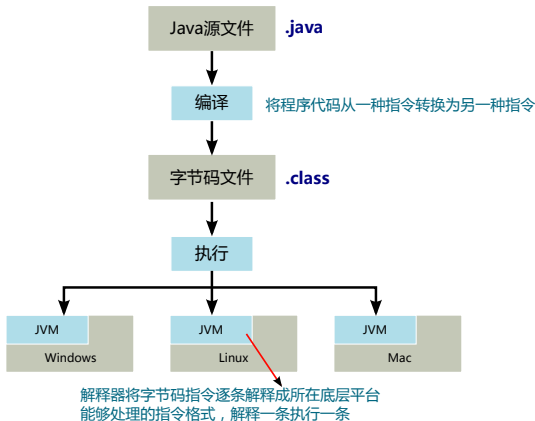


❖ 核心概念

- ▶ Java 虚拟机
- ▶ 垃圾回收机制
- ▶ Java 运行时环境 (Java Runtime Environment, JRE)



Java 程序的运行过程



JIT, Just-In-Time

传统解释器的解释执行是转换一条，运行完后就将其扔掉；JIT 会自动检测指令的运行情况，并将使用频率高（如循环运行）的指令解释后保存下来，下次调用时就无需再解释（相当于局部的编译执行），显著提高了 Java 的运行效率。



接下来...

Java 技术概述

Java 平台核心机制

Java 开发环境

Java 基本开发流程



获取和安装 Java 开发工具集

❖ Download

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

❖ Install

```
1 D:\Program Files\Java
```

```
1 /opt/jdk1.6.0_14
```

❖ Environment Variable

变量名 Path

变量值 D:\Program Files\Java\jdk1.6.0\bin



获取和安装 Java 开发工具集

❖ JDK6.0 Directories

```
1 bin      jre      README_ja.html   sample
2 COPYRIGHT lib     README_zh_CN.html src.zip
3 db       LICENSE  register.html    THIRDPARTYLICENSEREADME.txt
4 demo    man      register_ja.html
5 include  README.html register_zh_CN.html
```

bin Java 开发工具，包括编译器、虚拟机、调试器、反编译器；

jre Java 运行时，包括 Java 虚拟机、类库和其他资源文件；

lib 类库和所需支持性文件；

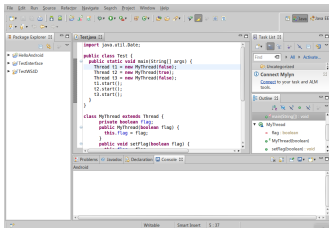
include 用于调试本地方法（底层平台）的 C++ 头文件；

src.zip 类库的源代码；

db Java DB 数据库，JDK6.0 新增项目，一种纯 Java 的关系型数据库；



- Eclipse

[illegible]

接下来...

Java 技术概述

Java 平台核心机制

Java 开发环境

Java 基本开发流程



Java 基本开发流程

1. 创建源文件 HelloWorld.java，文件命名必须与类名相同。

```
1 public class HelloWorld {  
2     public static void main(String[] args) {  
3         System.out.println("Hi, Java!");  
4     }  
5 }
```

2. 将源文件编译为字节码文件

```
1 > javac HelloWorld.java && ls  
2 HelloWorld.class HelloWorld.java
```

3. 运行程序

```
1 > java HelloWorld  
2 Hi, Java!
```



Java 基本开发流程

1. 创建源文件 HelloWorld.java，文件命名必须与类名相同。

```
1 public class HelloWorld {  
2     public static void main(String[] args) {  
3         System.out.println("Hi, Java!");  
4     }  
5 }
```

2. 将源文件编译为字节码文件

```
1 > javac HelloWorld.java && ls  
2 HelloWorld.class HelloWorld.java
```

3. 运行程序

```
1 > java HelloWorld  
2 Hi, Java!
```



Java 基本开发流程

1. 创建源文件 HelloWorld.java，文件命名必须与类名相同。

```
1 public class HelloWorld {  
2     public static void main(String[] args) {  
3         System.out.println("Hi, Java!");  
4     }  
5 }
```

2. 将源文件编译为字节码文件

```
1 > javac HelloWorld.java && ls  
2 HelloWorld.class HelloWorld.java
```

3. 运行程序

```
1 > java HelloWorld  
2 Hi, Java!
```



Java 应用程序结构需掌握的几条规则

1. **Java 语言拼写是大小写敏感的 (Case-Sensitive);**
2. 一个源文件中可以定义多个 Java 类，但其中最多只能有一个类被定义为 Public 类;
3. 如果源文件中包含了 public 类，则源文件必须和该 public 类同名;
4. 一个源文件包含多个 Java 类时，编译后会生成多个字节码文件，即每个类都会生成一个单独的 “.class” 文件，且文件名与类名相同。



Java 应用程序结构需掌握的几条规则

1. Java 语言拼写是大小写敏感的 (Case-Sensitive);
2. 一个源文件中可以定义多个 Java 类，但其中最多只能有一个类被定义为 Public 类;
3. 如果源文件中包含了 public 类，则源文件必须和该 public 类同名;
4. 一个源文件包含多个 Java 类时，编译后会生成多个字节码文件，即每个类都会生成一个单独的 “.class” 文件，且文件名与类名相同。



Java 应用程序结构需掌握的几条规则

1. Java 语言拼写是大小写敏感的 (Case-Sensitive);
2. 一个源文件中可以定义多个 Java 类，但其中最多只能有一个类被定义为 Public 类;
3. 如果源文件中包含了 public 类，则源文件必须和该 public 类同名;
4. 一个源文件包含多个 Java 类时，编译后会生成多个字节码文件，即每个类都会生成一个单独的“.class”文件，且文件名与类名相同。



Java 应用程序结构需掌握的几条规则

1. Java 语言拼写是大小写敏感的 (Case-Sensitive);
2. 一个源文件中可以定义多个 Java 类，但其中最多只能有一个类被定义为 Public 类;
3. 如果源文件中包含了 public 类，则源文件必须和该 public 类同名;
4. 一个源文件包含多个 Java 类时，编译后会生成多个字节码文件，即每个类都会生成一个单独的 “.class” 文件，且文件名与类名相同。





THE END

wxd2870@163.com

