

讲义01：课程介绍 & Java概述



主要内容

- 课程情况介绍

- 任课教师、课程及考核方式
- 教材及参考书
- 教学内容安排

- Java语言概述

- Java平台概述
- Java开发环境的搭建、Java IDE
- Java应用程序结构

- 内容来源资料:

- 教材第1章、第2章
- Java官网: <http://java.sun.com>

1 教师介绍

任课教师

工作单位

办公地点

联系电话

电子邮件

课程QQ群

专业兴趣

2 课程介绍

- 面向对象程序设计 Object Oriented Programming
- 这是关于软件开发的重要的基础性课程
- 理论课：32学时
- 实验课：32学时

2 课程介绍——考核方式

- **总评成绩：期末考试*60%+平时成绩*40%**
 - 期末考试：闭卷考试120分钟，**一票否决!!!**
 - 平时成绩：考勤+实验作业+课堂测验+课外交流
 - 其中，实验作业：验证性实验*60%+综合性实验*40%
- **作业及实验提交：通过网络平台提交**
 - 一般作业：提交答案
 - 编程训练：提交可运行程序
 - 综合实验：提交可运行程序+实验报告
- **网址： <http://192.168.120.252/oops/>**

2 课程介绍——教材

- 主教材:

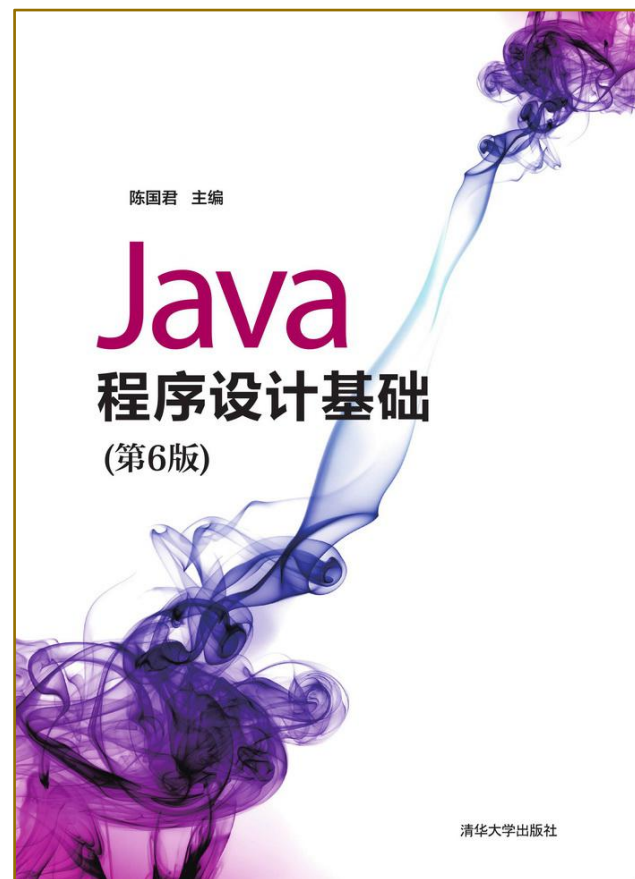
- Java程序设计基础(第6版)

- 参考书:

- Java语言程序设计.基础篇
(第10版)

- Java核心技术 卷I: 基础知识
(第9版)

- Java编程思想 (第4版)



2 课程介绍——讲授内容组织

学时：8+8

Java语言基础

讲义01：课程介绍与Java语言概述

讲义02：Java语言基础与流程控制

讲义03：数组与字符串

学时：12+12

Java面向对象

讲义04：类与对象

讲义05：类的特性

讲义06：继承、抽象类、接口

学时：12+12

Java基本应用

讲义07：异常处理

讲义08：输入输出、文件处理

讲义09：泛型与容器类

讲义10：注解、内部类与Lambda表达式

注：教材内其他内容为自学内容，下学期程序设计与综合实习III使用

2 课程介绍——学习目标

Goal of 面向对象程序设计

能力

语言运用

方法应用

团队合作

继续学习

内容

Java语言

OOP方法

协作开发软件

OO技术
Java高级应用

环节

理论课

理论课

课程设计

理论课

实验课

实验课

课外研讨

实验课

综合实习

课外研讨



错误：这门课就是学Java语言！

2 课程介绍——学习目标

问题：我们如何才能达到这个目标呢？

最根本的问题：

- 1. 我们会花多少时间在编程训练上？**
- 2. 我们会花多少时间与老师和其他人交流？**



3 Java介绍 —— 了解 Java 技术

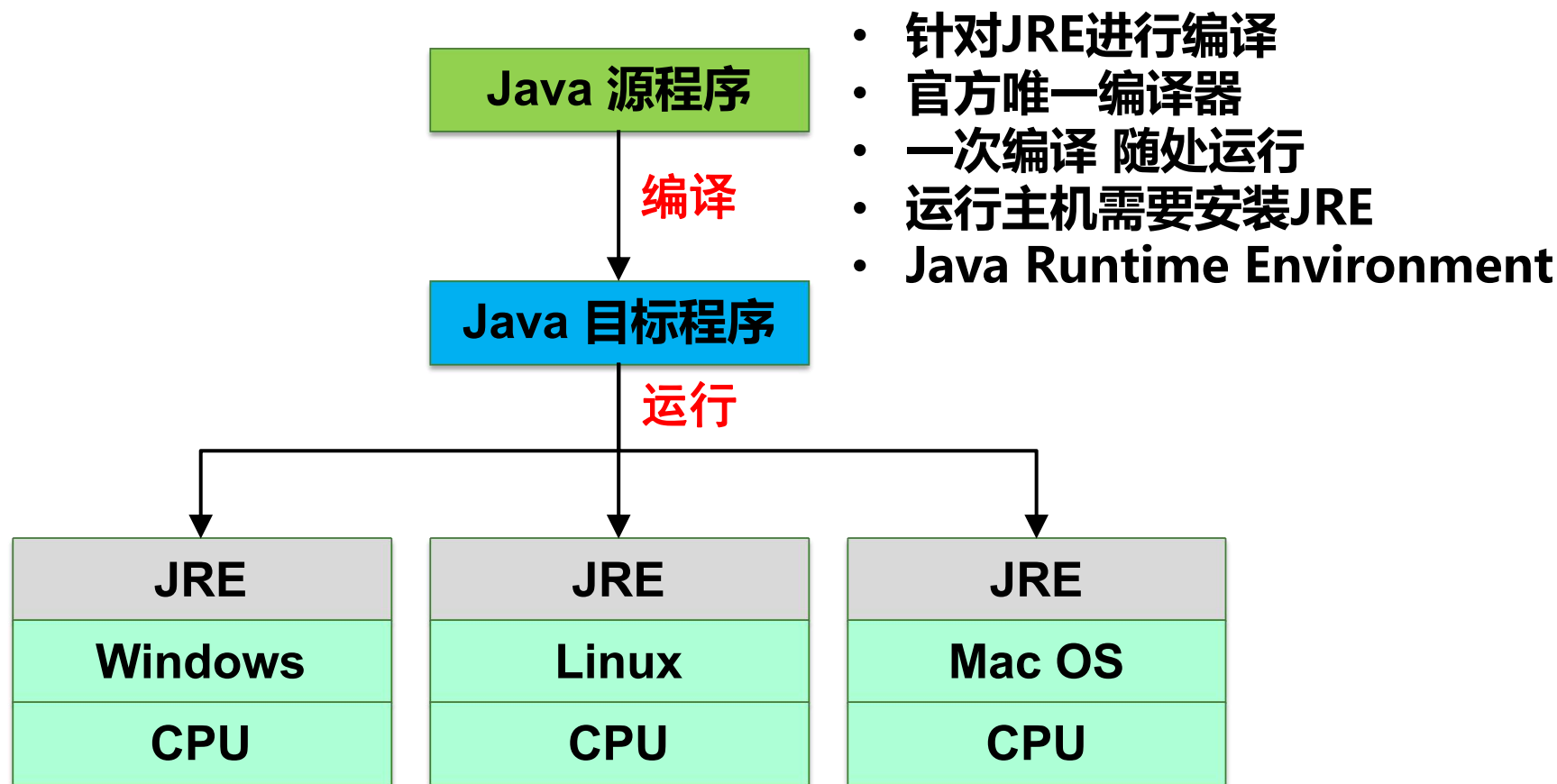
Java是几乎所有类型的网络应用程序的基础，也是开发和提供嵌入式和移动应用程序、游戏、基于Web的内容和企业软件的全局标准。



- 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

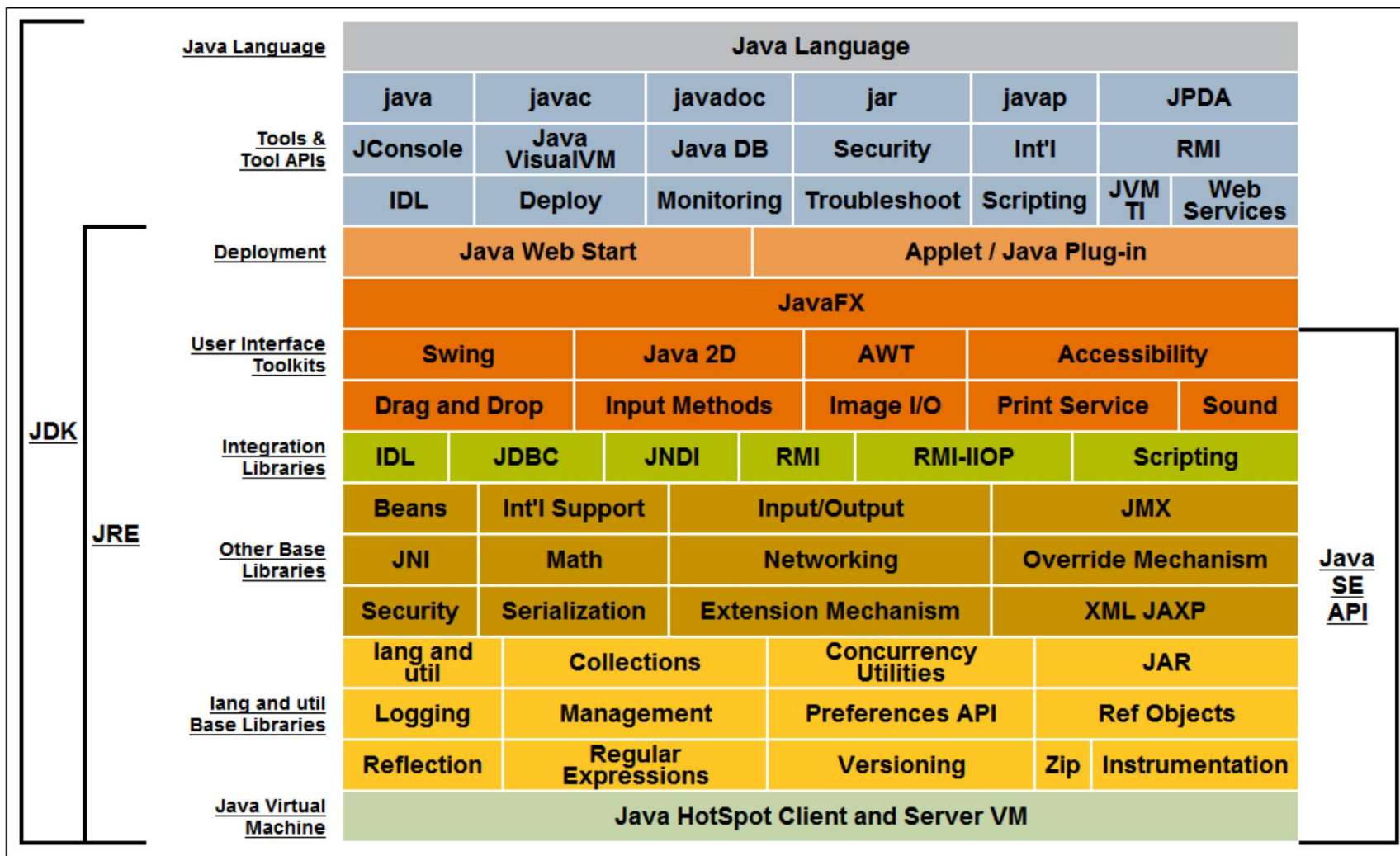
本页内容来自：http://www.java.com/zh_CN

3 Java介绍 —— Java应用程序的运行机制



3 Java介绍 —— Java平台

Java开发平台,主要包括JavaSE和JavaEE, Java语言是平台的组成部分。



4 Java开发环境搭建

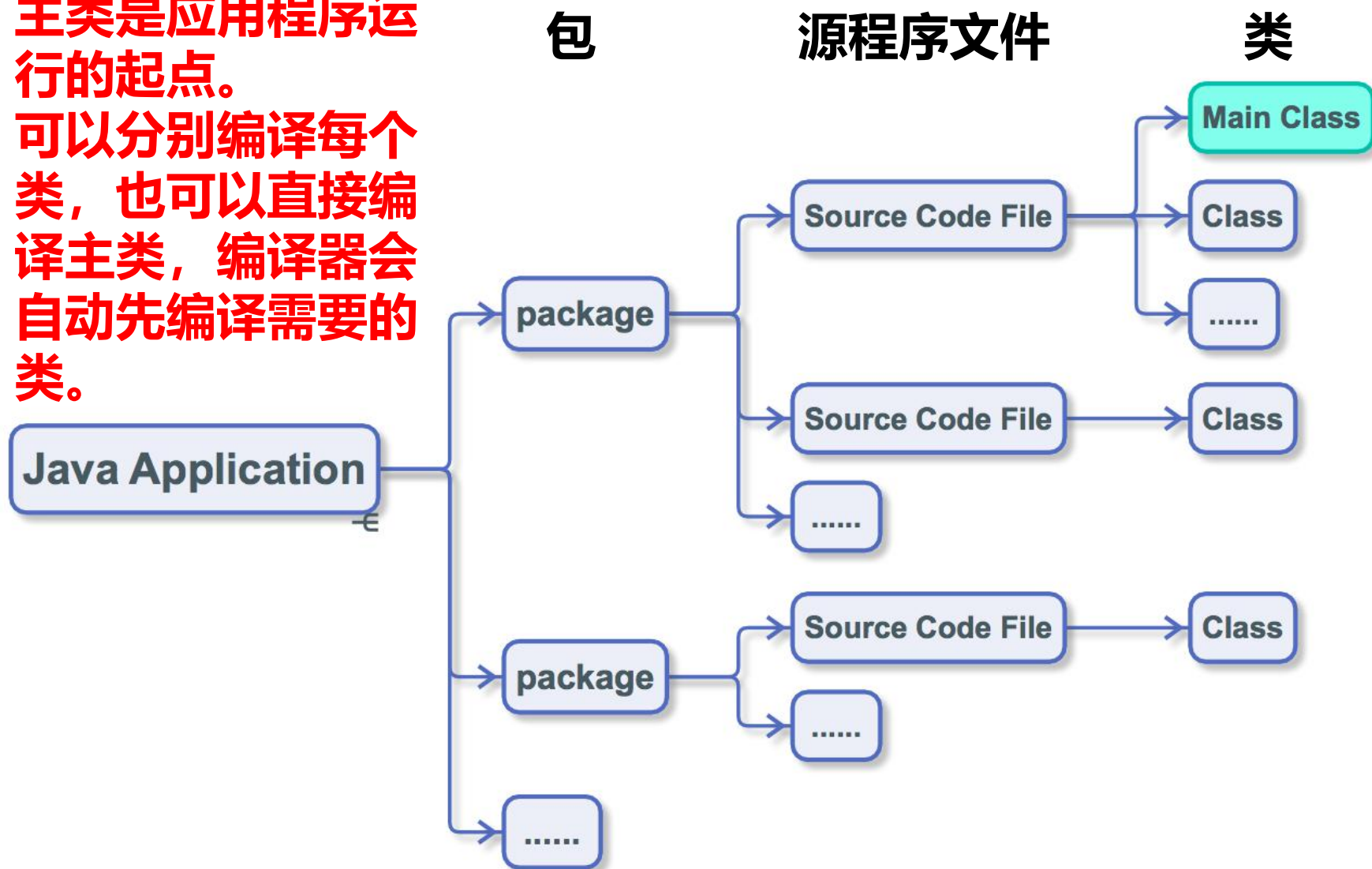
- 下载JDK (Java SE), JDK(Java Development Kit) 是 Java 语言的软件开发工具包, 网址: java.sun.com
注意根据操作系统选择JDK, 例如:
Windows64位 : `jdk-8u172-windows-x64.exe`
- 安装JDK, JRE包含在JDK中
- 下载 JDK API 文档 和 示例:
示例: `jdk-8u172-windows-x64-demos.zip`
文档: `jdk-8u172-docs-all.zip`; `javafx-8u172-apidocs.zip`
- 下载和安装IDE(集成开发环境)
 - [Eclipse, 开源, eclipse.org](http://eclipse.org)
 - Netbeans, 开源, netbeans.org
 - IntelliJ IDEA, 商业, www.jetbrains.com/idea/

5 Java程序开发的基本步骤



6 Java应用程序的基本结构

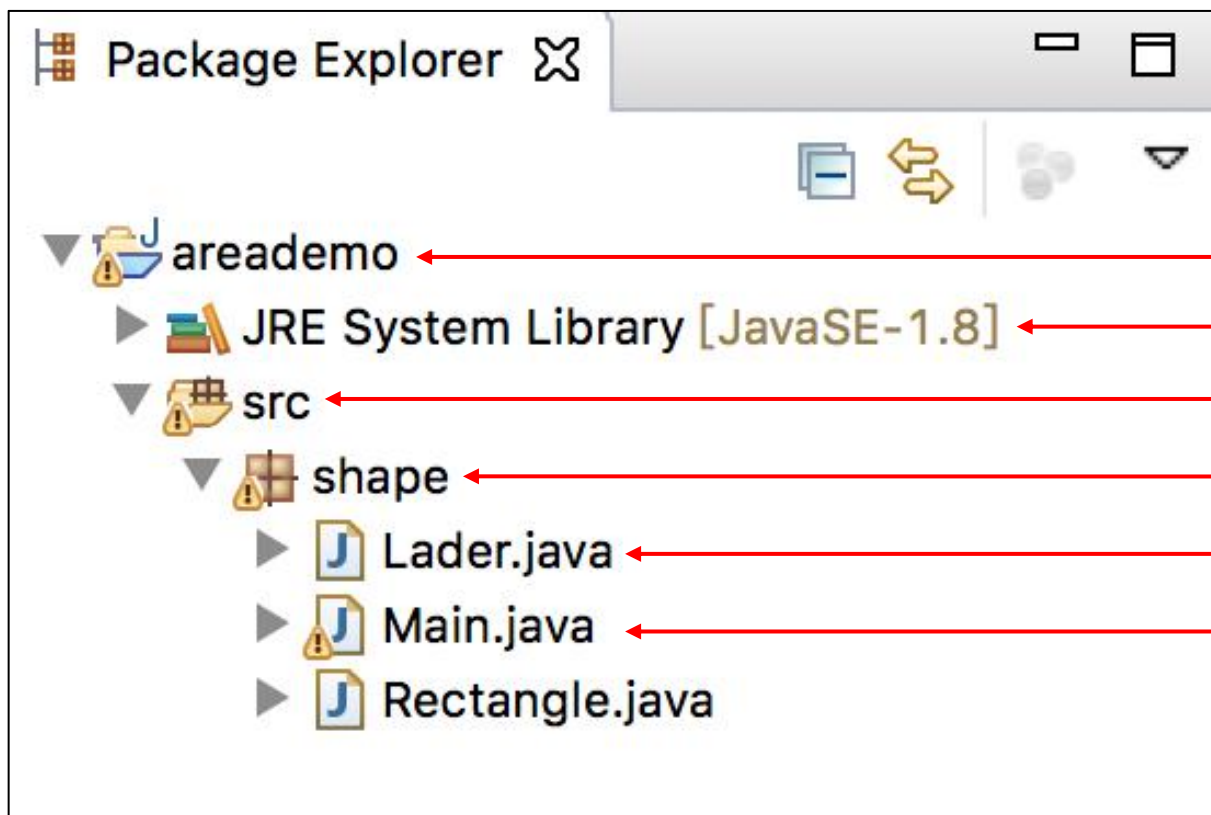
- 主类是应用程序运行的起点。
- 可以分别编译每个类，也可以直接编译主类，编译器会自动先编译需要的类。



6 Java应用程序的基本结构

直接使用JDK提供的开发工具管理Java应用程序是非常复杂和繁琐的事情。因此实际开发中使用**IDE**来提高开发效率。

Eclipse的Java Application结构如下图：



项目名称

项目使用的JRE

项目的源程序目录

项目的一个包

一个源程序文件

主类,
其中定义main方法

7 一个简单的Java应用程序

一个定义主类的源程序文件：FirstApplication.java

包语句	→	package package01;
导入语句	→	import java.util.Scanner;
定义类	→	public class FirstApplication {
主方法	→	public static void main(String[] args) {
		Scanner input = new Scanner(System.in);
		System.out.print("Input 2 integers: ");
		int a = input.nextInt();
		int b = input.nextInt();
		int c = a + b;
		System.out.println(a + "+" + b + "=" + c);
		}
		}

演示使用eclipse的开发方法。

8 在一个源程序文件中编写多个类

- Java允许在一个源程序文件中编写多个类。
- 同一个源程序文件中最多只能有一个类用public修饰。并且要求源程序文件名与public类的类名相同(包括字母大小写)。
- 编译源程序时，每个类生成一个扩展名为class的字节文件。

强烈建议：

**一个源程序文件中只编写一个类；
并且使用public修饰该类。**

9 编程风格

现代软件架构都需要协同开发完成，高效协作即降低协同成本，提升沟通效率，所谓无规矩不成方圆，无规范不能协作。

对软件来说，适当的规范和标准绝不是消灭代码内容的创造性、优雅性，而是限制过度个性化，以一种普遍认可的统一方式一起做事，提升协作效率。

码出高效，码出质量

一种符合规范的说明，请参照：[阿里巴巴 Java 开发手册 – 编程规约部分](#)

不同的开发团队可能有不同的习惯，但是一些通用的规范是大家应该遵守的。

9 编程风格 - Allmans风格

```
class Allmans
```

左、右大括号单独一行

```
{
```

```
    public static void main(String[] args)
```

```
{
```

```
        int sum = 0;
```

```
        for(int i = 0; i <= 100; i++)
```

```
{
```

```
            sum += i;
```

```
}
```

```
        System.out.println(sum);
```

```
}
```

```
}
```



9 编程风格 - Kernighan风格

左大括号在上一行的行尾
右大括号单独一行

```
class Kernighan {  
    public static void main(String[] args) {  
        int sum = 0;  
        for(int i = 0; i <= 100; i++) {  
            sum += i;  
        }  
        System.out.println(sum);  
    }  
}
```



10 Java语言的注释

```
package shape;
/**
 * 类 Lader 表示一个梯形 .....
 * @author xiaolei
 */
public class Lader {
    /** 梯形的上底 */
    double above;
    // 省略部分代码
    /**
     * 计算梯形的面积
     * @return 当前梯形对角的面积
     */
    double getArea() {
        double area; //定义表示面积的局部变量
        /*
         * 使用梯形面积公式计算梯形面积
         */
        area = (above + bottom) * height / 2;
        return area;
    }
}
```

- **文档注释** `/** */`
用于对类、数据域、方法进行说明

- **单行注释** `//`

- **多行注释** `/* */`



课后工作

- 在自己的电脑上安装JDK和IDE
- 完成作业网站：练习1