**《Java程序开发》课程教学大纲**

1. **课程名称：** Java程序开发
2. **课程代码:** CST31103
3. **课程英文名称：**Java Programming
4. **课程负责人：** 刘骥
5. **学时与学分**： 3学分，总学时56，其中理论课40学时，实验课16学时
6. **课程性质：** 专业课
7. **课程类型:** 必修课
8. **适用专业**： 计算机科学与技术/信息安全/物联网工程
9. **选课对象**： 本科生
10. **预修课程**： 面向对象程序设计、程序设计基础
11. **使用教材：**

[1] 朱庆生,古平, Java程序设计（第2版），清华大学出版社，2018。

教材选用说明：

教材[1]为国家“十一五”规划教材，由本校长期从事Java教学的教师，根据教学实践编写。教材内容已经在教学中使用多年，包含Java SE、Java EE和Java开发工具等内容。选用该教材有利于学生快速掌握Java基础知识。

1. **参考书目：**

[1] 埃克尔,Think in Java (Java编程思想•第4版）,机械工业出版社 ,2007。

[2] Y. Daniel Liang, Introduction to Java Programming, Prentice Hall, 2013。

[3] Joshua Bloch, Effective Java, Prentice Hall, 2008。

[4] Jason Hunter, William Crawford , Java Servlet Programming（Java Servlet编程）, O'Reilly Media Inc., 2001。

[5] Elliotte Rusty Harold ,Java Network Programming（Java网络程序设计）, O'Reilly Media Inc., 2004。

1. **开课单位**： 计算机学院
2. **课程描述**

Java是目前IT业界使用最为广泛的编程技术。它以面向对象语言Java为基础，包含Java SE（Java Standard Edition）、Java EE（Java Enterprise Edition）、Java ME（Java Mobile Edition）三大平台以及庞大的开源软件库。Java具有简单、面向对象、分布式、健壮、安全、平台独立、可移植、多线程、动态性等特点。使用Java可以开发图像界面、网络、智能手机、Web、嵌入式、大数据和人工智能等多种应用。无论是科学研究，还是工程应用，Java都是软件开发的主流技术。

本课程将重点讲解Java SE。在课程后期，为方便学生后续的学习，简要讲解Java EE和相关开源软件。课程通过大量编程案例和实践，培养学生软件开发的能力。通过课程的学习，学生不仅可以掌握Java语言的基本语法、开发环境、开发技术，还能够掌握用Java技术解决复杂问题的基本方法。本课程的主要教学内容包括以下3个部分：

（1）Java语言，讲解Java语法和基于Java的面向对象编程方法。

（2）Java SE，讲解集合框架、输出输出、异常处理、多线程、图形用户界面和网络程序等内容。

（3）Java EE，讲解Servlet、JSP、JDBC和Web开发框架等内容。

1. **课程支撑的毕业要求指标点及课程目标**

**1.本课程支撑的毕业要求指标点**

（1）指标点R1.4：能够运用计算机专业知识，设计和实现计算机软件核心模块。

（2）指标点R3.1：能针对特定的计算机复杂工程问题进行调研完成需求分析。

（3）指标点R5.2：针对复杂计算机工程问题，能够运用图书馆、互联网、数据库等多种资源，检索和分析所需的软硬件开发工具的相关资料，熟练掌握开发环境与开发工具的使用方法。

**2.本课程拟达到的教学目标**

通过本课程的理论学习，引导学生应用课堂学得的理论知识，完成各种实践项目，从而掌握Java的编程知识，达到课程的培养目标。具体来说，课程的主要教学目标包括：

（1）学生能够理解Java程序设计的原理，应用Java进行程序开发，拥有阅读和编写软件模块级的Java程序代码的能力；

（2）学生拥有对软件模块进行需求调研，完成需求分析的能力；

（3）学生拥有运用图书馆、互联网、数据库等资源，检索相关资料，学习并熟练使用开发环境和开发工具的能力。

**3.课程教学目标与毕业要求指标点的关系**

|  |  |  |  |
| --- | --- | --- | --- |
| 课程教学目标 | 毕业要求二级指标点 | | |
| R1.4 | R3.1 | R5.2 |
| 子目标1 | ✓ |  |  |
| 子目标2 |  | ✓ |  |
| 子目标3 |  |  | ✓ |

课程教学子目标1：通过课堂教学讲授Java程序设计的理论和方法，包括Java的基本语法、基于Java的面向对象编程方法、Java SE技术（集合框架、输出输出、异常处理、多线程、图形用户界面和网络程序等）和Java EE技术（Servlet、JSP、JDBC和Web开发框架等）。在此基础上，通过在线判题系统答题（Online Judge，以下简称OJ）、实验项目等环节使学生应用这些基础知识，培养学生阅读和编写软件模块级Java程序代码的能力，从而对毕业要求指标点R1.4提供支撑。

课程教学子目标2：课程要求学生在实验项目中，对软件需求进行详细的分析、规范的描述，从而培养学生针对软件模块需求进行调研，完成需求分析的能力，以支撑毕业要求指标点R3.1。

课程教学子目标3：课程要求学生在使用OJ、完成实验项目等教学环节时，运用图书馆、互联网、数据库等检索资料，学习并掌握Eclipse、Tomcat、Git、Maven等开发环境和开发工具的使用，从而对毕业要求指标点R5.2提供支撑。

1. **课程教学方式/方法**

课程教学目标1：采用课堂教学、课堂讨论和课后作业等方式，使学生掌握Java程序开发基础知识；通过OJ答题，布置若干编程任务，使学生经过训练，掌握Java程序开发的基本方法；通过实验项目，使学生能够熟练的阅读和编写软件模块级的Java程序代码。

课程教学目标2：在课堂教学中，通过分析、讲解优秀的设计案例，使得学生理解软件需求分析的基本方法；在实验项目中，要求学生在实验报告中进行需求分析，训练学生需求分析的能力。

课程教学目标3：布置自学任务，由学生通过图书馆、互联网、数据库等资源，学习开发环境和开发环境的使用；在实验项目中使用Java开发环境和工具，逐渐熟练并掌握。

1. **课程教学内容与学时**

**1.课堂教学（40学时）**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **一级知识点** | **二级知识点** | **三级知识点** | **掌握层次** | **学时数分配** |
| Java技术基础 | Java技术概述 | Java技术的发展历史 | 记忆 | 1 |
| Java技术的特点 | 记忆 |
| Java8后的版本演变与区别 | 记忆 |
| Java平台、开发环境和开发工具 | Java平台（Java SE、Java EE和Java ME） | 应用 | 1 |
| Java开发环境（eclipse、netbeans、intellj） | 应用 |
| Java开发工具（java命令行工具、Maven、git等） | 应用 |
| Java虚拟机（JVM） | Java程序的运行过程 | 理解 | 2 |
| Java内存区域 | 理解 |
| JVM垃圾回收 | 理解 |
| 类文件结构 | 理解 |
| 类加载过程 | 理解 |
| 类加载器 | 理解 |
| JVM常用配置参数和常用GC调优策略 | 理解 |
| Java语言基础 | 变量和常量 | 应用 | 4 |
| 基本语句 | 应用 |
| 常用关键字 | 应用 |
| 运算符 | 应用 |
| 注释和注解（Annotation） | 应用 |
| 枚举 | 应用 |
| 数组 | 应用 |
| 引用 | 应用 |
| 异常处理 | 应用 |
| Java面向对象程序设计 | 类和对象 | 应用 | 4 |
| 构造方法 | 应用 |
| 类的可见性 | 应用 |
| 对象数组 | 应用 |
| 不变类 | 应用 |
| this指针 | 应用 |
| 继承 | 应用 |
| super | 应用 |
| 多态 | 应用 |
| 类型转换 | 应用 |
| 抽象类 | 应用 |
| 接口 | 应用 |
| 内部类和匿名类 | 应用 |
| lambda表达式 | 应用 |
| Java SE | Java String | String的创建 | 应用 | 1 |
| String常用方法 | 应用 |
| StringBuffer、StringBuilder和StringTokenizer | 应用 |
| Java I/O | File类 | 应用 | 2 |
| Text I/O | 应用 |
| 二进制I/O | 应用 |
| 序列化 | 应用 |
| BIO、NIO和AIO | BIO（Blocking I/O） | 理解 | 1 |
| NIO（New I/O） | 理解 |
| AIO (Asynchronous I/O) | 理解 |
| Java集合框架 | Java集合类的框架和继承层次 | 应用 | 4 |
| Set（HashSet、SortedSet、TreeSet） | 应用 |
| List（ArrayList、LinkedList） | 应用 |
| Map（HashMap、TreeMap、LinkedHashMap） | 应用 |
| Stream API（foreach、map、filter、Collectors） | 应用 |
| Java GUI | Java Swing的层次结构 | 应用 | 4 |
| Java Swing容器（JFrame、JPanel、JDialog） | 应用 |
| Java Swing布局 | 应用 |
| Java Swing事件处理 | 应用 |
| 使用Java 2D进行图形绘制 | 应用 |
| Java多线程 | 线程的创建 | 应用 | 4 |
| 线程的生命周期与控制 | 应用 |
| 线程同步与互斥 | 应用 |
| Java网络程序设计 | 客户/服务器端通信 | 应用 | 4 |
| Socket和ServerSocket | 应用 |
| Datagrampacket和DatagramSocket | 应用 |
| Java EE基础 | Servlet和JSP | Eclipse创建并运行动态Web项目 | 应用 | 2 |
| Tomcat服务器上运行Web程序 | 应用 |
| Servlet | 应用 |
| JSP | 应用 |
| Session、Filter和Listener | 应用 |
| JSP Model 2 | JSP Scriptlet的概念和优缺点 | 理解 | 2 |
| JSP Model1的概念和优缺点 | 理解 |
| JSP Model2的概念和优缺点 | 理解 |
| 常见Web框架简介 | 理解 |
| JDBC | JDBC的基本使用 | 应用 | 2 |
| 用JDBC操作关系型数据库（MySQL） | 应用 |
| 开发实例 | 表示层设计与实现 | 理解 | 2 |
| 控制层设计与实现 | 理解 |
| 业务层设计与实现 | 理解 |

注释1：课程内容按照相对独立性，用层次结构列出1、2、3级知识点单元。

注释2：对知识学习的要求一般分为“记忆，理解，应用”三个层次。

**2.实验教学（16学时）**

（1）实验目的

让学生通过上机操作，领会Java程序设计的基本原理、方法，掌握相关软件开发环境和开发工具，掌握软件模块需求分析的方法，培养阅读和编写软件模块的能力。

（2）实验内容及学时分配（根据教学情况从6个题目中选择2个）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **实验名称** | **实验内容** | **实验软硬件环境要求** | **学时数分配** |
| 1 | 基于命令行的文件管理器 | （1）实现文件夹创建、删除、进入；  （2）实现当前文件夹下的内容罗列；  （3）实现文件拷贝和文件夹拷贝（文件夹拷贝指深度拷贝，包括所有子目录和文件）；  （4）实现指定文件的加密和解密；  （5）使用Eclipse、NetBeans或Intellj作为开发环境，使用Maven管理项目，使用Git进行源代码控制；  （6）在实验报告中，对软件需求进行规范、详细的描述。 | 内存2GB以上/硬盘20GB以上的个人电脑或笔记本电脑，主机需要安装Java SDK 1.8以上版本，以及Eclipse、NetBeans或Intellj集成开发环境，Maven和Git等Java工具。 | 8 |
| 2 | 基于命令行的学生成绩管理系统 | （1）设计学生类接口；  （2）使用集合框架，实现学生成绩的插入、通过名字查询、显示各科成绩排名列表、显示各科成绩通过学生学号的排名列表；  （3）将内存中的集合框架对象进行持久化，下一次运行系统时可直接调用；  （4）使用Eclipse、NetBeans或Intellj作为开发环境，使用Maven管理项目，使用Git进行源代码控制；  （5）在实验报告中，对软件需求进行规范、详细的描述。 | 内存2GB以上/硬盘20GB以上的个人电脑或笔记本电脑，主机需要安装Java SDK 1.8以上版本，以及Eclipse、NetBeans或Intellj集成开发环境，Maven和Git等Java工具。 | 8 |
| 3 | 具备验证码的Web用户登录系统 | （1）Web登陆界面；  （2）使用Servlet类，实现登陆请求的转发和验证；  （3）使用Java GUI 中的图片操作类，根据随机生成的数字构造验证码；  （4）使用Tomcat作为服务器，使用Eclipse、NetBeans或Intellj作为开发环境，使用Maven管理项目，使用Git进行源代码控制；  （5）在实验报告中，对软件需求进行规范、详细的描述； | 内存2GB以上/硬盘20GB以上的个人电脑或笔记本电脑，主机需要安装Java SDK 1.8以上版本，以及Eclipse、NetBeans或Intellj集成开发环境，Maven和Git等Java工具，安装Tomcat服务器。 | 8 |
| 4 | GUI计算器 | （1）具备基本的加减乘除功能；支持由括号和运算符组成的表达式运算；  （2）使用GUI界面元素设计用户友好的界面；  （3）能够存储多个计算结果，并提取出来作为下一个计算的操作数；  （4）能够处理计算过程中的异常情况；  （5）使用Eclipse、NetBeans或Intellj作为开发环境，使用Maven管理项目，使用Git进行源代码控制；  （6）在实验报告中，对软件需求进行规范、详细的描述。 | 内存2GB以上/硬盘20GB以上的个人电脑或笔记本电脑，主机需要安装Java SDK 1.8以上版本，以及Eclipse、NetBeans或Intellj集成开发环境，Maven和Git等Java工具。 | 8 |
| 5 | GUI文件管理系统 | （1）实现文件夹创建、删除、进入；  （2）实现当前文件夹下的内容罗列；  （3）实现文件拷贝和文件夹拷贝（文件夹拷贝指深度拷贝，包括所有子目录和文件）；  （4）实现指定文件的加密和解密；  （5）实现指定文件和文件夹的压缩；  （6）实现压缩文件的解压；  （7）文件管理器具有图形界面；  （8）使用Eclipse、NetBeans或Intellj作为开发环境，使用Maven管理项目，使用Git进行源代码控制；  （9）在实验报告中，对软件需求进行规范、详细的描述。 | 内存2GB以上/硬盘20GB以上的个人电脑或笔记本电脑，主机需要安装Java SDK 1.8以上版本，以及Eclipse、NetBeans或Intellj集成开发环境，Maven和Git等Java工具。 | 8 |
| 6 | 联网对战俄罗斯方块游戏 | （1）实现俄罗斯方块的基本游戏逻辑；  （2）利用Socket编写服务器端和客户端程序；  （3）实现俄罗斯方块的联网对战功能。  （4）实现计分、音效等辅助功能；  （5）使用Eclipse、NetBeans或Intellj作为开发环境，使用Maven管理项目，使用Git进行源代码控制；  （6）在实验报告中，对软件需求进行规范、详细的描述。 | 内存2GB以上/硬盘20GB以上的个人电脑或笔记本电脑，主机需要安装Java SDK 1.8以上版本，以及Eclipse、NetBeans或Intellj集成开发环境，Maven和Git等Java工具。 | 8 |

备注：有关各次上机实验的具体内容、实验开展方式等要求，请参见“附录：《Java程序开发》实验报告”中的详细说明。

（3）实验要求

1）学生每2-4人一组；

2）学生课前自学掌握Java开发环境（Eclipse、NetBeans或Intellj）和Java开发工具（Maven、Git和Tomcat等）的安装和使用；

3）学生实验前在个人电脑上安装Java SDK 1.8以上版本，安装 Eclipse、NetBeans或Intellj集成开发环境，安装Maven、Git和Tomcat等Java开发工具；

4）课内实验与课外实验结合，课堂无法完成的部分允许学生在课外完成。

5）实验报告必须包含软件需求分析，并进行规范、详细的描述。

（4）实验教材（讲义、指导书）

同课程教材。

（5）实验对课程教学目标的支持

通过实验，综合应用学到的Java知识，培养学生阅读和编写软件模块级Java程序代码的能力，从而支持课程教学子目标1。

通过在实验报告中对软件需求进行详细的分析、规范的描述，培养需求分析的能力，从而支持课程教学子目标2。

在实验中，学生需要通过查找资料，自学Eclipse、Tomcat、Git、Maven等开发环境和开发工具的使用，从而支持课程教学子目标3。

1. **能承担此课的教师**

刘骥、杨瑞龙、杨广超、曾令秋、吴全旺

教学大纲制订者：刘骥

教学大纲审定者：杨瑞龙