进度安排

第1天（第一章）Java基础入门

- 概述、历史
- JavaSE JavaME JavaEE
- JDK JRE JVM
- Java特点
- hello world案例
- 跨平台原理
- 编译javac及运行java命令
- package
- API文档
- 补充内容
  - JVM【可以在讲解JDK JRE 时稍微提一下】
  - 垃圾回收器
  - 字节码验证

第2天（第二章）标示符、关键字、变量（Eclipse）

- STS安装配置及使用（常用快捷键及设置，大家自行决定何时讲解）
- 注释

- 关键字
- 标识符
- **字面值常量**
- 变量
- **数据类型**
- 进制及转换
- 原反补码
- **隐式类型转换**
- 显式类型转换

第3天（第三章）操作符、流程控制

- 操作符
- 流程控制
- if
- switch
- for
- while

第4天（第三章）流程控制

- do-while
- break

- continue
- **循环嵌套**
- label
- Random
- **方法定义、调用、重载**

第5天（第四章）数组

- 数组概述
- 数组定义
- 数组下标、长度、默认值
- 初始化

  动态初始化、静态初始化、默认值
- 内存构成
- 数组异常
- 工具类
- 排序案例

第6天（第五章）类和对象

- 数组拷贝
- 二维数组【不用过多花费时间精力】
- 可变参数列表【面试常考】

- 面向对象和面向过程理解
  - 面向对象思想关键：由执行者 转变为 指挥者
  - 有现成的类对象可以解决问题，就借助、指挥对象去解决问题
  - 如果没有则创建一个对象去解决问题
  - 如果没有这样的类，则设计一个具有指定功能的类，再创建对象去解决问题，设计的类后续可以复用
- 对象
- 类定义格式
- 对象创建
- 对象内存构成【重要】
  - 可以使用pdf上的图，也可以自己画图帮助学生理解
- 引用数据类型
  - 可结合上述对象内存图理解

第7天（第五章）类和对象

- 成员变量、局部变量区别【面试常考】
- 封装概念及优点
- 目前封装实现
  - private数据成员
  - 提供public的get|set方法
- this关键字
  - 要理解this内存构成【重要】
- 构造方法

    - 构造方法的注意事项要罗列清楚
- this补充：调用构造方法【必须第一行有效代码】

第8天（第六章）static、继承、重写、多态

- static关键字
    - 静态数据成员
    - 静态成员方法
- 代码块
    - 局部代码块【了解】
    - 构造代码块
    - 静态代码块【面试常考】
- 对象创建过程【重要，一定要站在内存的角度区分析】
- 静态导入【了解】
- 继承
    - 单继承
    - 多层继承
    - 子类对象内存构成【重要】
        - 堆空间开辟内存，只存储普通数据成员【含从父类继承部分+新增部分】
        - 普通成员方法放在方法区
        - static成员放在专门的静态区
    - 继承理解
        - Java官方文档描述：**从继承的概念来说，private修饰的成员不被继承**

        - 实际内存效果：子类全部继承父类内容，但对父类中private成员没有直接访问权限【建议认真阅读PDF文档进行理解】

第9天（第六章）static、继承、多态

- super关键字
    - 如果局部变量、子类新增数据成员、父类继承数据成员出现同名情况
    - 优先级：局部变量 > 子类新增 > 父类继承成员【重要】
- 子类构造方法
    - 借助父类构造方法实例化父类部分数据成员
- 权限修饰符

  public > protected > default > private

  主要掌握public private

  【另外两个稍微演示下即可，不用花费过多时间精力】
- 方法重写
    - 可以暂不讨论异常、返回不同类型，超纲内容，后期讲到异常和引用类型转换时候再补充即可
- final关键字
    - 修饰类
    - 修饰方法
    - 修饰局部变量
    - 修饰成员变量【初始化 较为复杂，要特别注意】
    - 修饰的变量如果是引用类型，其引用内存里面的值可以改变
- 多态

第10天（第七章）抽象、接口、内部类、枚举

- 引用类型转换
  - 向上转型
  - 向下转型
  - instanceof
- abstract
  - 抽象理解：不明确的、不具体的。在计算机科学和编程中，抽象是一种关注问题的本质和关键特征，而忽略具体实现细节的方法。
  - 抽象方法
  - 抽象类
- interface 接口
  - 接口理解：它不是类，它是对Java单继承的补充，用于额外增强子类的【功能】
  - 接口定义
  - 类和接口的实现关系
  - 接口和接口可以进行多继承
  - 类可以继承1个父类，同时实现多个接口
  - JDK8、JDK9新特性【了解即可，后续JDK8课程会补充】
- 抽象类和接口区别【面试常考】
  - 语法区别
  - 设计理念区别【十分重要】
- 内部类

    - 成员内部类【一般】
    - 静态内部类【一般】
    - 局部内部类【了解】
    - 匿名内部类【重点】

第11天（第七章）抽象、接口、内部类、枚举

- 包装类
    - Integer常量池【难点 面试常考】
- Object类
    - toString() 之前已经讲过
    - equals() 重点讨论
    - hashCode() 了解，Set集合会专门讨论
    - getClass() 了解即可，反射会重点讨论
- String类
    - 字符串常量池【难点 面试常考】
- 枚举
    - 枚举概念及意义
    - 枚举类定义1，最基本方式【最常用】
    - 枚举类定义2，包含构造方法【关键点：构造方法必须private修饰】
    - 枚举定义3，包含抽象方法【复习抽象、重写】
- 面向对象部分复习回顾

第12天（第八章）集合、泛型、注解

- 集合
  - 概述，注意和数组对比
  - Collection接口，注意继承体系
  - 实例化集合对象固定格式

    `Collection<String> coll = new ArrayList<>();`【先这么用，后续泛型补充】
  - 基本方法使用
    - 基础方法 add size clear isEmpty等
    - 相对复杂方法 addAll contains remove XxxAll方法...
    - contains、remove相关方法，底层借助equals方法实现比较
    - 集合存储自定义类对象，注意一定要重写自定义类equals方法【重要】
  - 集合遍历
    - toArray()遍历【了解】
    - 迭代器遍历【重点】
    - foreach循环【简单，使用最多】
  - List集合
  - List实现类
    - ArrayList 底层借助数组实现，今后使用最多
    - LinkedList 底层借助双向链表，可以观察源码【关键点】
    - Vector　　**JDK1.0版本，早期方法繁琐，线程安全**
  - 数据结构

    数组、链表、栈、队列、红黑树、哈希表

【大致介绍，让学生掌握各种数据结构特点即可，不用过度讲解】

第13天（第八章）集合、泛型、注解

- 集合
  - Set集合
  - Set实现类
    - HashSet

      如果存储自定义类对象，一定要重写hashCode和equals方法【重要】

      HashSet插入元素步骤：先**获取元素的hashCode()值**，如果相同**再获取equals()值**进一步比较，相同则插入失败，不同则插入成功【重要】

      另外注意`hashCode()`复习
    - TreeSet

      先存入Integer对象，观察其特点

      再存入自定义Person对象，运行异常：ClassCastException，引出2种排序规则

      通过自然排序，解决上述案例中异常问题

      再通过比较器排序，解决上述问题

      最后自然排序、比较器排序同时用上，得出结论：比较器排序优先级高于自然排序
  - Map集合概述及遍历
    - keySet遍历
    - entrySet遍历
  - Map实现类

- HashMap

  key所属类型**一定要重写hashCode和equals方法【重要】**

  补充：HashSet底层借助HashMap实现功能

- Hashtable

  JDK1.0版本，线程安全，方法繁琐，key value不可以为null

- TreeMap

  红黑树实现，key所属类型应该指定排序规则：**自然排序、比较器排序**

- LinkedHashMap

  HashMap的一个子类，底层在哈希表的基础上，通过维护一个双向链表来保持键值对的有序性

第14天（第八章）集合、泛型、注解

- 集合工具类
- 斗地主综合案例

  注意分析过程【重要】

- 泛型
  - 概述
  - 集合章节中泛型的应用
  - 自定义泛型及使用
    - 泛型类【学生掌握基本定义格式，会简单使用即可】
    - 泛型接口
    - 泛型方法

    - 泛型使用注意事项【= 号俩边的所指定的泛型类型，必须是要一样的】

      `ArrayList<Object> list = new ArrayList<Integer>();`编译报错

      `ArrayList<Integer>`**和**`ArrayList<Object>`**之间没有子父类型的关系**，它们是俩种不同类型
    - 通配符
    - 泛型边界【不建议过度解读，学生能看得懂代码，会简单使用即可】
        - ? extends E
        - ? super E
    - 类型擦除【与后续反射章节呼应】
- 注解
    - 概述
    - 格式
    - 范围
    - 保持
    - 元注解

第15天（第九章）异常

- 异常
    - 异常体系
    - 异常种类：编译时异常、运行时异常【关键点，案例中要区分】
    - 异常传播行为
- 异常抛出

    - 系统自动抛出
    - 程序员主动抛出 throws
- 异常捕获
    - try - catch块
    - try - 多个catch块【注意事项：父子类异常同时出现时，子在上，父在下】
    - try - catch(多种异常类型)
    - finally关键字【面试常考】
- 自定义异常
- 断言 assert

第16天（第十章）线程

- 进程和线程：概念理解
- 并发和并行
    - 我们讨论的是并发，俩个或多个线程，**使用同一个CPU交替运行**
- 时间片
- 线程调度算法
- 线程的2种创建方式
    - 匿名内部类写法
- 线程名相关方法
    - `public static native Thread currentThread();`
    - `public final String getName();`
    - 通过线程对象设置线程名

```
public final synchronized void setName(String name);
```

- 创建对象时，设置线程名

```
public Thread(String name);
```

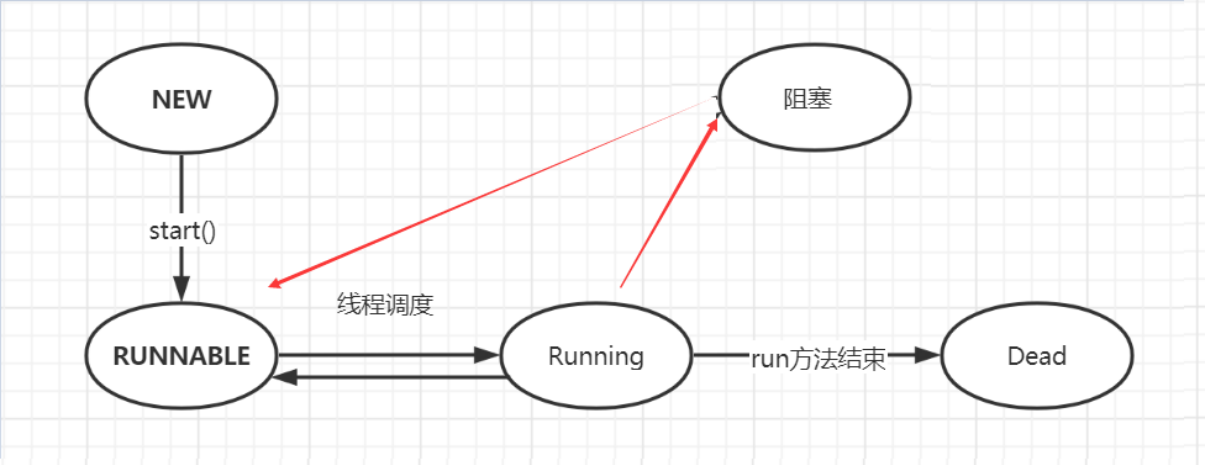
```
public Thread(Runnable target, String name);
```

- main线程
- 守护线程
- 线程优先级【建议性】

第17天（第十章）线程

- 线程组【简单了解，使用即可】
- 线程状态

  可以先大致介绍，让学生记住最基本那个转换图，后续慢慢补充。



- sleep方法

  补充状态转换内容

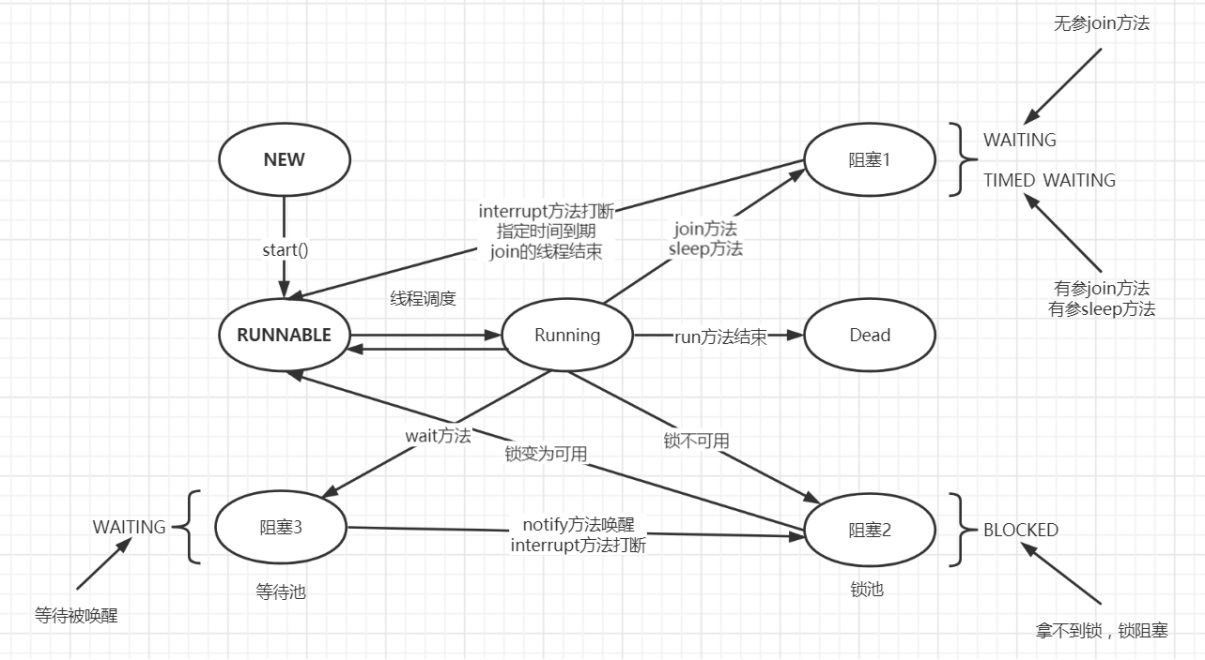
- join方法

  补充状态转换内容

- 线程安全
- 同步代码块
- 同步方法
- 线程通信

第18天（第十章）线程

- 等待唤醒机制
  - wait和notify
  - 注意wait内部原理
- 2个线程通信【要提前熟悉案例】
- 多个线程通信【要提前熟悉案例】
- 线程状态总结



- 死锁

  避免死锁情况

- 线程池
- 复习总结

第19天（第十一章）IO流

- File类相关
  - 基础操作
  - 遍历功能
- 流的概念
  - 基本概念
  - 继承体系
- 流的分类
  - 输入、输出流
  - 字节、字符流
- 字节流
  - 文件输入流：3个read方法详解
  - 文件输出流：3个write方法详解
  - 文件追加功能
  - 内存流

第20天（第十一章）IO流

- 字符流

    - 文件流
    - 操作字节文件，会出现问题，得出结论：字符流只能用来操作字符文本文件
- 异常处理
    - 标准方式
    - JDK7新方式
- 节点流总结
    - 节点流
    - 增强流（包装流）：底层借助其他流实现功能，一般情况下**节点流则作为增强流的基础，提供最底层的读写功能。**
- 缓冲流【重点】
    - 缓冲思想
    - 缓冲字节流：拷贝百十兆的文件，与文件字节流做性能对比
    - 缓冲字符流：`readLine()、newLine()`方法补充
- 标准流
    - PrintStream打印流，增强流，提供方便地输出各种数据类型的值到输出流
    - 标准输入、输出、出错流【了解】
    - System.setIn() System.setOut() 可修改
- 中文乱码问题，原理分析：编码不一致
- 转换流【重点】

第21天（第十一章）IO流

- 序列化与反序列化

- 对象流【重要】
  - 操作的类必须实现序列化接口
  - 传递单个对象案例
  - 传递多个对象，借助集合传输
  - transient关键字
  - serialVersionUID序列号
- 数据流

  提供专门用于读写基本数据类型和字符串的方法
- 随机访问流

  既可以读也可以写

  可以随便调整文件位置
- Properties类【实用】

  load() getProperty() stringPropertyNames() setProperty() store()
- 复习总结

第22天（第十二章）网络编程

- 网络编程基本概念
  - 计算机网络
  - 软件结构
  - 通信三要素
- IP地址
- 端口
- 协议

    - OSI七层模型【了解】
    - TCP/IP四层模型【了解】
- TCP和UDP【面试常考】
    - 两者对比
    - TCP三次握手
    - TCP四次挥手
- TCP网络编程【重案例】
    - 基础案例
    - 反转字符串【采用增强流】
    - 对象传输案例
    - 多线程服务器案例
- UDP网络编程【了解即可】

第23天（第十三章）反射及复习

- JVM虚拟机
- 垃圾回收机制
- 类加载
- 反射
- 复习回顾