深圳分中心-课程体系-V2.1

- JavaSE零基础
  - 环境搭建
    - 常用dos命令及系统快捷键
    - Sublime Text或EditPlus的安装
    - 通用文本编辑快捷键
    - 安装JDK
    - 计算机的组成
    - 计算机语言发展史
    - Java语言发展史
    - Java语言特点
    - Java加载和执行的过程
    - 第一个Java程序的开发
    - Java注释
    - public class和class的区别
  - 基础语法
    - 标识符
    - 关键字
    - 字面量
    - 变量
    - 数据类型
      - 数据类型概述
        - 数据类型作用
        - java中包括哪些数据类型
        - 二进制
        - 8种基本数据类型一览
      - 整数型详解
        - 编写代码时整数型的4种表示法
        - 整数型字面量可以使用下划线分隔增强可读性（Java7新特性）
        - java中整数型字面量被默认当做int类型来处理，想以long的形式表示需要在字面量后添加L/l
        - int i = 10; (10是int类型，i是int类型，不存在类型转换)
        - long x = 10;(10是int类型，x是long类型，小容量可以自动转换成大容量，称为自动类型转换)
        - long y = 10L;(10L是long类型，y是long类型，不存在类型转换)
        - int z = 10L;(10L是long类型，z是int类型，大容量无法直接转换成小容量，编译报错，需要加强制类型转换符，int z = (int)10L，这种操作叫做强制类型转换，强制类型用可能损失精度)
        - long a = 2147483648;编译报错的原因，怎么修改
        - 当整数型字面量没有超出byte、short、char取值范围时，可以直接赋值
        - 原码、反码、补码，为什么byte b = (byte)200;的结果是：-56
      - 浮点型详解
        - java开发中浮点型字面量默认被当做double来处理，想默认被当做float处理，需要在字面量后面添加F/f
        - float f = 2.0; 编译报错的原因，怎么修改
        - 为什么java中不推荐使用双等号来判断两个浮点型数据是否相等
      - 布尔型详解
        - java中布尔型字面量只有true和false，无其他值
        - 布尔型变量主要使用在逻辑判断方面
      - 字符型详解
        - 字符编码是一张文字与二进制的转换对照表
        - 'A' 'a' '0'这三个字符的ASCII码
        - 字符编码的发展过程，以及开发常用的字符编码
        - java中规定字符char必须使用单引号括起来
        - java中的char可以存储一个汉字
        - java中的转义字符\t、\r、\n、\\、\'、\"、\u
      - 基本数据类型转换规则
    - 运算符
      - 算术运算符
      - 关系运算符
      - 逻辑运算符
      - 赋值运算符
      - 字符串连接运算符
      - 条件运算符
      - 位运算符
    - 控制语句
      - if
      - switch
      - for
      - while
      - do..while
      - break
      - continue
    - 方法初步
    - 方法重载/overload
      - 方法重载有什么好处
      - 什么时候考虑使用方法重载
      - 满足什么条件时构成方法重载
      - 演示方法重载在实际开发中的使用
    - 方法执行时的内存变化
      - 栈数据结构
      - JVM内存模型中有：栈空间
    - 方法递归
      - 什么是方法递归，怎么理解
      - 方法递归没有结束条件会怎样
      - 不使用递归计算n的阶乘
      - 使用递归计算n的阶乘
      - 方法递归时的内存图
      - 多数情况下为什么建议使用循环代替递归
    - package与import
      - 软件包的作用
      - 包名命名规范
      - 带包怎么编译和运行
      - import什么时候可以省略，什么时候必须使用
  - IntelliJ IDEA工具的使用
    - IDEA工具的安装与破解
    - IDEA工具怎么设置字体与字符集
    - IDEA中怎么创建Project
    - IDEA中怎么创建Module
    - Project与Module的关系
    - 怎么创建package
    - 怎么创建class
    - 怎么快速生成main方法
    - 怎么快速生成System.out.println()
    - 怎么运行java程序
  - 面向对象
    - 面向过程和面向对象的区别
    - 类和对象的概念
    - 类的定义
    - 对象的创建和使用
    - 构造方法
      - 怎么定义构造方法
      - 怎么调用构造方法
      - 缺省构造器
      - 构造方法的作用
      - 构造方法支持重载
    - 当实例变量是一个引用的时候
    - 面向对象三大特征
    - 封装性
      - 不封装有什么问题
      - 怎么封装，代码怎么写
      - 总结封装有什么作用
    - this关键字
    - 方法调用时参数的传递
    - static关键字
      - static关键字概述
      - 静态代码块
      - 构造代码块
      - 静态变量
      - 静态方法
      - 静态导入
    - 空指针异常
    - 类的继承
      - 继承的作用
      - 继承的基础语法
      - 子类继承父类，在创建子类对象时，构造方法是如何执行的
    - 方法覆盖/方法重写/override
      - 回顾方法重载overload
      - 什么时候考虑方法覆盖
      - 满足什么条件时构成方法覆盖
    - super关键字
    - 多态
      - 多态的基础语法
      - 多态在开发中的作用