



目录

[Java串讲1](#)

[1 数据类型](#)

[2 运算符](#)

[3 流程控制](#)

[4 数组](#)

[5 变量、方法](#)

[6 面向对象](#)

[7 基础API](#)

[8 集合](#)

[9 异常](#)

[10 io](#)

[11 线程](#)

[12 网络](#)

[13 反射](#)

Java串讲1

1 数据类型

- 基本类型（8种）
- 引用类型

2 运算符

- & | ^ ~ >> >>> <<

3 流程控制

switch

只能判断byte, short, char, int

enum

4 数组

- 创建

- `int[] a = new int[5];`
- `int[] a = {3,65,2,4};`
- `a = new int[]{4,2,26};`
- ~~`int a[] = new int[5];`~~

- 遍历

```
for(int i=0;i<a.length;i++) {  
    String s = a[i];  
    //s  
}  
  
for(String s : a) {  
    //s  
}
```

5 变量、方法

- 方法递归

简化到最简，反向倒推

6 面向对象

- 什么是面向对象

- 封装，继承，多态

- 封装

- 类，模板
- 对象，实例
- 引用，遥控器
- 构造方法
 - ◆ 默认构造
 - ◆ 重载
 - ◆ 构造方法间调用 `this(...)`
- 重载
 - ◆ 同名不同参
- 隐藏，方便维护修改

- 继承
 - 作用：代码重用，复用
 - 子类对象
 - ◆ 父类对象和子类对象绑定，整体作为一个对象
 - ◆ 调用成员，先找子类，再找父类
 - 父类构造方法
 - ◆ 默认 `super()`
 - ◆ 手动调用 `super(...)`
 - `super`
 - ◆ `super.xxxx()`
 - ◆ `super(...)`
 - 重写
 - ◆ 继承的方法，在子类中重新定义，重新编写
- 多态
 - 作用：一致的类型
 - 向上转型、向下转型
 - `instanceof`
 - ◆ 对真实类型和父类型，都返回true
 - `obj.getClass() == A.class`
 - ◆ 只对真实类型返回true
- 抽象类，半成品
- `final`
 - 常量，不可变
 - 方法，不能重写
 - 类，不能继承
- `static`
 - 属于类
 - 不是面向对象的语法
 - 静态初始化块
 - `static {`
 - `}`
- 访问控制符
 - `public`, `protected`, `[default]`, `private`
- 对象的创建过程（10步）
- 接口
 - 作用：结构设计工具，解耦合、隔离实现
 - 规范、协议
- 内部类
 - 非静态内部类、静态内部类、局部内部类
 - 匿名内部类

7 基础API

- `Object`
 - `toString()`
 - `equals()`

- hashCode()
- getClass()
- wait()
- notify()
- notifyAll()
- String
 - 常量池
 - 不可变
- StringBuilder/StringBuffer
 - append() 代替字符串, 做高效字符串连接
- 正则
- 基本类型包装类
 - Integer.valueOf(5)
256个缓存对象, -128到127
- BigDecimal/BigInteger
 - 精确浮点数运算, 超大整数运算
 - setScale() 舍入运算
- Date
- SimpleDateFormat

8 集合

- ArrayList
 - 数组
 - 默认长度 10
 - 1.5倍增长
 - 访问任意位置效率高
 - 增删数据, 效率可能降低
- LinkedList
 - 双向链表
 - 两端效率高
- HashMap
 - 哈希运算过程
 - ◆ key.hashCode() 获得键的哈希值
 - ◆ 用哈希值计算下标 i
 - ◆ 新建Entry对象封装键值对, 放入i位置
 - 空位置, 直接放入
 - 有数据, 依次用equals()比较是否相等
 - 有相等的, 覆盖值
 - 没有相等的, 链表连接在一起
 - 负载率、加载因子, 0.75
 - 新建翻倍长度的新数组
 - 所有数据重新哈希计算下标位置, 放入新数组
 - jdk1.8
 - 链表长度到8, 转成红黑树
 - 树上的数据减少到6, 转回链表

- HashSet
- LinkedHashMap
- ConcurrentHashMap
- ConcurrentHashMap
 - 分段加锁

9 异常

NullPointerException
ArrayIndexOutOfBoundsException
ClassCastException
ClassNotFoundException

10 io

File
字节流
字符流

- ObjectInputStream/ObjectOutputStream
 - 序列化
 - 序列化对象必须实现 Serializable

11 线程

- 创建
 - 继承Thread
 - 实现Runnable
- 方法
 - sleep()
 - join()
 - interrupt()
 - ...
- synchronized
 - synchronized(对象) {
}
 - synchronized void f() {
}
 - ◆ 当前对象的锁
 - static synchronized void f() {
}
 - ◆ 类对象的锁

- 生产者、消费者
 - 线程之间，传递数据
- 等待通知
 - 必须在synchronized代码中才能调用
 - 必须在加锁的对象上等待或发送通知

12 网络

- ServerSocket
 - accept()
 - close()
- Socket
 - getInputStream()
 - getOutputStream()
 - setSoTimeout()
 - close()

13 反射

- 获取类对象
 - A.class
 - a.getClass()
 - Class.forName("xxx.xxx.A")
- 新建实例
 - c.newInstance()
 执行无参构造