

Scala-week2

陈超
@CrazyJvm



- 类(class)

- 声明类（一个源文件中可以包含很多类，并且可以都是public级别）
- getter 和 setter
- 构造函数(primary constructor & auxiliary constructor)
- 继承(extends)
- 重写父类方法(override def)
- 重写字段(override val , override var)

- 抽象类 (abstract class)
 - 类的一个或者多个方法没有完整的定义
 - 声明抽象方法不需要加abstract关键字,只需要不写方法体
 - 子类重写父类的抽象方法时不需要加override
 - 父类可以声明抽象字段(没有初始值的字段)
 - 子类重写父类的抽象字段时不需要加override

- 特质(trait) —对比下JAVA8的接口
 - 字段和行为的集合
 - 混入类中
 - 通过with关键字，一个类可以扩展多个特质

- trait 续
 - 当做接口
 - 带有具体实现的接口
 - 带有特质的对象
 - 特质从左到右被构造

- apply方法
- 单例对象

- 包(package com.xx.data)
 - 支持嵌套，下层可以访问上层作用域中的名称
 - 可串联
 - 顶部标记
 - 包对象
 - 包可见性
 - 包在任何地方都可以引入，作用域至该语句所在块的末尾
 - 重命名引入成员(xx => yy)
 - 隐藏方法(xx => _)
 - 自动引入(java.lang._ scala._ Predef._)

- 模式匹配

- 标准用法(match)
- 使用守卫
- 匹配类型

- case class(多用在模式匹配中)
 - 构造器中的每一个类型都为val， 不建议用var。
 - 不用new就可以直接产生对象(为什么? apply方法)

文件访问

- 按行读取
- 按字符读取
- 从网络读取

Spark

SparkContext.scala

RDD.scala

再次强调

Option

String Interpolation

谢 谢

欢迎关注我的公众微信号 *ChinaScala*

