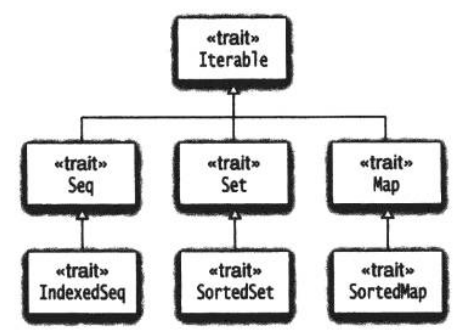
# 课堂笔记

## Scala中Iterable类继承关系



Seq是有先后顺序的序列，子类有Array、ArrayBuffer、List、Range、Vector...，可以通过下标访问。Array中元素是相同类型，List中可以是不同类型。

Set是无先后顺序，通过hashcode对元素去重。在SortedSet中，元素按某种顺序被访问（存储是无顺序的），不可以通过下标访问元素。

Map是key-value键值对，存储无顺序，SortedMap支持按某种顺序访问元素。

**统一创建原则：**

每个Iterable下的Trait或class都有一个带apply()的伴生对象，所以都可以这么建实例对象：

val m = Map((“k”,”v”),(“1”,”2”))

Set(1, 2 ,3)

List(3,4,5)

Array(5,7,9)

....

## 相关Trait或class

之前讲过ArrayBuffer，对应不可变的Scala.Array

**scala.collection.immutable 不可变包**

Map、HashMap

List

Set、HashSet、TreeSet

**scala.collection.mutable 可变包**

Map、HashMap

ListBuffer

Set、HashSet、TreeSet

另外，Seq子类还有：

Vector(向量)，相当于Array，不可变的ArrayBuffer，基本没区别。

Range是整数序列，必须Int类型，不可承载其他类型。可以通过to 和 until 方法来创建。

0 to 9

操作符和函数功能方面和List类似。

## List

//两种创建List的方式，List的元素内容不可变

**val** days = List("Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday")

**val** list = "aa" :: "bb" :: List()

**for**( i<- days)

{

println(i)

}

**var** l = **new** ListBuffer[Int]()

l.append(1,2,3)

//这个变量方法明显比较繁琐

// var it = l.iterator

// while(it.hasNext)

// {

// println(it.next())

// }

// 需要控制下标的变量

**for**(i <- 0 until l.size)

{

println(l(i))

}

println("-------------------")

l.append(10)

l.append(10,11,23,45)

//操作符，操作l本身

l += 5

l += (6,7,8)

l -= 5 //删掉最左边的一个5

//l 参与运算，l本身不变

**var** list2 = l - 1

list2 = l :+ (100)

//list 间操作

**var** list3 = list2 -- l //去掉list2中包含l的元素

**var** list4 = l ++ list2 //合并内容，赋值给新对象

**var** list5 = l ++: list2

练习题：

创建一个List（）包含多个word，计算wordCount