

Rust课程2.5

Rustlings 答疑 / 拓展内容

徐启航 西安交通大学



Rustlings答疑



特征



- 本质：表示一组特定的功能
- 包含：方法、关联类型与常量
- 示例：简介内置常用特征
- 示例：使用特征中的关联类型
- 示例：使用特征中的关联常量



特征：泛型关联类型

- 泛型关联类型：关联类型中的泛型
- 示例：使用泛型关联类型创建集合工厂
- 示例：在函数式编程中使用泛型关联类型



生命周期

- 本质：一个结构单元（变量 / 类型）存活的范围
- 生命周期注解：一种泛型
 - 一个确定的变量或确定的类型，存活的范围是确定的。
 - 生命周期注解 提供构造变量或者类型的模板参数，使得在不同代码上下文环境下，构造出的变量或者类型可以存活不同、符合预期的范围。
 - 与类型泛型对比
- 示例：注解变量的生命周期
- 示例：注解类型的生命周期



生命周期与子类型 (1) :

- 生命周期的包含关系：比...活得久 (outlive)
 - 结论 $'a: 'b \Leftrightarrow 'a \text{ 比 } 'b \text{ 活得久 (} a \text{ outlives } b \text{) (示例)}$
 - 定理 对于任意生命周期 $'a$, 有 $'\text{static}: 'a$ 。
- 子类型：可替换性 (示例)
 - 定义 $'x$ 是 $'y$ 的子类型
 - 大前提：对于某处不可变引用有对于单个生命周期的要求
 - 小前提： $'y$ 满足该要求
 - 结论： $'x$ 也满足该要求
 - 爆论 类型越“子”，可以满足的要求越多
- 结合以上两组结论，我们有 $'a: 'b \Leftrightarrow 'a \text{ 是 } 'b \text{ 的子类型}$
 - 想想为什么？

生命周期与子类型 (2)

- 将“ a 是 b 的子类型”表示为偏序关系 $<$ ， a 与 b 无关表示为 $a <> b$ ，则对于任意两个生命周期 a 和 b ，仅存在3种不等关系： $a < b$ 、 $a <> b$ 和 $a > b$ 。将这3种关系统称为 $R(a, b)$ 。
- R 对于泛型的映射
 - 对于单生命周期泛型 T ，这个映射即定义为 $T: R(a, b) \rightarrow R(T(a), T(b))$
 - 单向解释：如果 a 是 b 的子类型，那么 $T(a)$ 是不是 $T(b)$ 的子类型呢？还是相反？还是无关？
- 示例：查看常用类型的泛型映射

生命周期与子类型 (3)

	'a	T	U
&'a T	covariant	covariant	
&'a mut T	covariant	invariant	
Box<T>		covariant	
Vec<T>		covariant	
UnsafeCell<T>		invariant	
Cell<T>		invariant	
fn(T) -> U		contravariant	covariant
*const T		covariant	
*mut T		invariant	



- 目前Rust生命周期的子类型关系对于泛型存在三种映射：
- 协变(covariant) $T: R(T<'a>, T<'b>) = R('a, 'b)$ 。
- 逆变(contravariant) $T: R(T<'a>, T<'b>) = \sim R('a, 'b)$ ，也就是若' $a < 'b$ ，则 $T<'a> > T<'b>$ 。
- 不变(invariant) $T: R(T<'a>, T<'b>) = "<>或="$ ，也就是无法推导子类型关系。
- 示例：在Rustonomicon中展示常用泛型的子类型映射关系



高阶生命周期绑定（HRTB）：泛型的泛型

- 示例：一个指定生命周期绑定时的的问题
- 本质：将泛型中的生命周期绑定移到了实例化。
- 类比：
 - 普通生命周期绑定：存在某一正实数 α ， $\alpha > 0$ 。
 - 高阶生命周期绑定：对于任意正实数 α ， $\alpha > 0$ 。
- 示例：使用高阶生命周期绑定解决问题



成员方法

- 可以参与点运算符 (.) 的解引用
- 示例：基本使用
- 示例：使用智能指针包裹self
- Receiver 特征：包裹self的唯一实现要求
- 示例：使用Receiver 特征

感谢聆听！

Thanks for listening

