



# Rust 编程语言入门



Microsoft®  
Most Valuable  
Professional

杨旭，微软MVP  
Rust、Go、C#开发者

## 10.5 生命周期（1/4）

# 生命周期

- Rust 的每个引用都有自己的生命周期。
- 生命周期：引用保持有效的作用域。
- 大多数情况：生命周期是隐式的、可被推断的
- 当引用的生命周期可能以不同的方式互相关联时：手动标注生命周期。

# 生命周期 – 避免悬垂引用（dangling reference）

- 生命周期的主要目标：避免悬垂引用（dangling reference）
- （例子）

# 借用检查器

- Rust 编译器的借用检查器：比较作用域来判断所有的借用是否合法。

# 函数中的泛型生命周期

- （例子）

再见

