



Rust 编程语言入门



Microsoft®
Most Valuable
Professional

杨旭，微软 MVP
Rust、Go 开发者

17. Rust 的面向对象 编程特性

17.1 面向对象语言的特性

Rust 是面向对象编程语言吗？

- Rust 受到多种编程范式的影响，包括面向对象
- 面向对象通常包含以下特性：命名对象、封装、继承

对象包含数据和行为

- “设计模式四人帮”在《设计模式》中给面向对象的定义：
 - 面向对象的程序由对象组成。
 - 对象包装了数据和操作这些数据的过程，这些过程通常被称作方法或操作
- 基于此定义：Rust 是面向对象的
 - struct、enum 包含数据
 - impl 块为之提供了方法
 - 但带有方法的 struct、enum 并没有被称为对象

封装

- 封装：调用对象外部的代码无法直接访问对象内部的实现细节，唯一可以与对象进行交互的方法就是通过它公开的 API
- Rust: pub 关键字
- （例子）

继承

- 继承：使对象可以沿用另外一个对象的数据和行为，且无需重复定义相关代码
- Rust：没有继承
- 使用继承的原因：
 - 代码复用
 - Rust：默认 trait 方法来进行代码共享
 - 多态
 - Rust：泛型和 trait 约束（限定参数化多态 bounded parametric ）
- 很多新语言都不使用继承作为内置的程序设计方案了。

再见

