



# Rust 编程语言入门



Microsoft®  
Most Valuable  
Professional

杨旭，微软 MVP  
Rust、Go 开发者

# 18. 模式匹配

# 模式

- 模式是 Rust 中的一种特殊语法，用于匹配复杂和简单类型的结构
- 将模式与匹配表达式和其他构造结合使用，可以更好地控制程序的控制流
- 模式由以下元素（的一些组合）组成：
  - 字面值
  - 解构的数组、enum、struct 和 tuple
  - 变量
  - 通配符
  - 占位符
- 想要使用模式，需要将其与某个值进行比较：
  - 如果模式匹配，就可以在代码中使用这个值的相应部分

# 18.1 用到模式的地方

## match 的 Arm

- match VALUE {  
    PATTERN => EXPRESSION,  
    PATTERN => EXPRESSION,  
    PATTERN => EXPRESSION,  
}
- match 表达式的要求;
  - 详尽（包含所有的可能性）
- 一个特殊的模式：\_（下划线）：
  - 它会匹配任何东西
  - 不会绑定到变量
  - 通常用于 match 的最后一个 arm；或用于忽略某些值。

## 条件 if let 表达式

- if let 表达式主要是作为一种简短的方式来等价的代替只有一个匹配项的 match
- if let 可选的可以拥有 else，包括：
  - else if
  - else if let
- 但，if let 不会检查穷尽性
- （例子）

## while let 条件循环

- 只要模式继续满足匹配的条件，那它允许 while 循环一直运行
- （例子）

# for 循环

- for 循环是 Rust 中最常见的循环
- for 循环中，模式就是紧随 for 关键字后的值
- （例子）



# let 语句

- let 语句也是模式
- let PATTERN = EXPRESSION;
- （例子）

# 函数参数

- 函数参数也可以是模式
- （例子）

再见

