

# 25 | 在条件中使用其它类型 (L、R)

周爱民 (Aimingoo)

# 目录

- 1 在条件中使用其它类型
- 2 条件类型中的“条件” (L, R)
  - 3 惰性求值与延迟计算 (Deferred)
  - 4 条件类型带来的延迟计算
- 5 总结

# 总结

## 1. 在条件中使用其它类型（例如 L、R），本质上就是考察 L/R 的赋值兼容性

- ▶ 当 L/R 是惰性求值时，上述表达式无法完成考察，因此整个条件表达式也将是惰性求值的
- ▶ 某些情况下的“惰性求值”事实上是 TypeScript 中（友好的）显示类型的方式

## 2. 官方文档提出的延迟求值（Deferred）

- ▶ 延迟发生之前做的努力（后续内容）
- ▶ 将“延迟的条件类型”用于 extends 等类型运算（后续内容）



# 作业

➤ 尝试写出至少三个惰性求值的类型，并使用 `Trans<>` 或 `{...}` 验证。

```
type T = ...  
type T2 = Trans<T>;
```

➤ 课外阅读：尝试从官方文档中 ([release-notes/typescript-2.8.html](https://github.com/microsoft/TypeScript/wiki/Release-Notes#typescript-2.8)) 了解条件类型惰性求值的细节。

▶ 主要目的是了解问题的复杂度。

THANKS