

Vim 实用技巧

潘潘

2021 年 5 月 17 日

1 寄存器与宏

Vim 的寄存器是一组用于保存文本的简单容器。

寄存器的设计体现了 Vim 中对于“记录”和“取用”这两个概念的思考。寄存器提供了一组文本容器，可以存放完全由可见字符组成的一段文本，也可以存放其中夹杂着不可见字符的按键操作序列。在 Vim 的使用过程中，需要进行记录并在某些时候再次取出使用的场景，是寄存器和宏完美覆盖的场景。

从简单到复杂，寄存器的使用覆盖了文本的摘取/粘贴、按键操作的记录与播放。结合 Vim 已有的多次重复某次操作机制（数字 + 操作），有很多耗时且易错的操作可以简单分解为宏命令的播放，其中可以夹杂其他寄存器的内容获取。

- 修改寄存器内容
 - visual 模式下复制选定文本到寄存器 x: "xy
 - 对于复制操作有专用的寄存器 0，而无名寄存器的内容会缺省地被复制、删除、剪切操作影响。
 - normal 模式下，qx 将会开始录制按键操作到 x 中，再按一次 q 结束录制。
- 取用寄存器内容
 - insert 或 ex 模式下读取寄存器 x 的可见字符部分到光标处: ^rx
 - normal 模式下读取寄存器 x 的内容到光标前后: "xP 或者 "xp

- normal 模式下播放寄存器 x 中的内容: @x
- ex 模式下播放寄存器 x 中的内容: <motion>normal @x
- 以及一些内置约定:

寄存器	内容
"%	当前文件名
"#	轮换文件名
".	上次插入的文本
":	上次执行的 Ex 命令
"/	上次查找的模式