

(vim) 光标轨迹

手稿

封面

vim 支持按照单词、句子、段落和行移动光标。

概述

前文讲过，在命令模式下，按下 h / l 键分别将光标向左 / 向右移动一个字符的位置。除了单个字符外，vim 还支持光标以单词（word）、句子（sentence）、段落（paragraph）或者行（line）等单位移动光标。

单词

vim 有 word 和 WORD 两种单词。

◆ word

由字母、数字和下划线组成，或者是由空白（空格、制表符、回车、换行等）分隔的非空白字符组成。

◆ WORD

由空白分隔的非空白字符组成，空行也被认为是 WORD。

按 word 划分

T h e r e a r e 2 f o r m s o f w o r d s (w o r d a n d W O R D) .

① ② ③ ④ ⑤ ⑥ ⑦ ⑧ ⑨ ⑩ ⑪

按 WORD 划分

T h e r e a r e 2 f o r m s o f w o r d s (w o r d a n d W O R D) .

① ② ③ ④ ⑤ ⑥ ⑦ ⑧

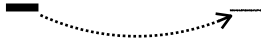
单词轨迹

如果想要 vim 按照单词来移动光标，可以按下：

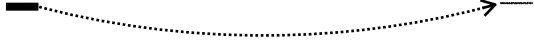
- ✦ w 移至下一个 word
- ✦ W 移至下一个 WORD
- ✦ e 移至下一个 word 结尾处（跳过空行）
- ✦ E 移至下一个 WORD 结尾处（跳过空行）
- ✦ b 移至前一个 word
- ✦ B 移至前一个 WORD
- ✦ ge 移至前一个 word 结尾处
- ✦ gE 移至前一个 WORD 结尾处

单词轨迹图示

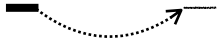
按下 w 键 T h e r e a r e 2 f o r m s o f w o r d s (w o r d a n d W O R D) .



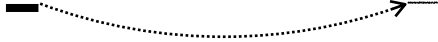
按下 W 键 T h e r e a r e 2 f o r m s o f w o r d s (w o r d a n d W O R D) .



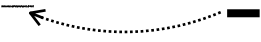
按下 e 键 T h e r e a r e 2 f o r m s o f w o r d s (w o r d a n d W O R D) .



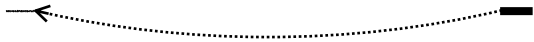
按下 E 键 T h e r e a r e 2 f o r m s o f w o r d s (w o r d a n d W O R D) .



按下 b 键 T h e r e a r e 2 f o r m s o f w o r d s (w o r d a n d W O R D) .



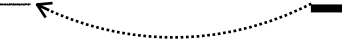
按下 B 键 T h e r e a r e 2 f o r m s o f w o r d s (w o r d a n d W O R D) .



按下 ge 键 T h e r e a r e 2 f o r m s o f w o r d s (w o r d a n d W O R D) .



按下 gE 键 T h e r e a r e 2 f o r m s o f w o r d s (w o r d a n d W O R D) .



以句点 (.)、感叹号 (!) 或问号 (?) 结尾，后跟行结束标志（比如回车、换行等）或者空白。

This a sentence. This is another sentence. This

① ②

is the 3rd sentence.

③

句子轨迹

如果想要 vim 按照句子来移动光标，可以按下：

- ✦ (移至上一个句子
- ✦) 移至下一个句子

This a sentence. This is another sentence. This
←————— (————— (—————) ————— →
is the 3rd sentence.

段落

vim 中段落的定义是：

段落开始于每个空行之后，或者位于段落宏定义标记（由 paragraphs 选项定义）之间。
注意：空白行（只有空白字符的行）不视作段落边界。

①

Function `linear()` is simple, it is used to calculate general linear result.

②

`linear()` conforms to the folloing formula:

$$y = f(x) = a + bx$$

there are three parameters in function `linear()` which named as `a`, `b` and `x` corresponding to the above formula.

③

Among these three parameters, `a` is the constant term or the `y` intercept, `b` is the coefficient of the independent variable, `x` is independent variable. `y` or `f(x)` is dependent variable, and is returned as return value.

段落轨迹

如果想要 vim 按照段落来移动光标，可以按下：

- ✦ { 移至上一个段落
- ✦ } 移至下一个段落

Function linear() is simple, it is used to calculate general linear result.

```
{ ←  
linear() conforms to the folloing formula:  
    y = f(x) = a + bx  
there are three parameters in function linear() which named as a, b and x  
    corresponding to the above formula.  
} ←
```

Among these three parameters, a is the constant term or the y intercept,
b is the coefficient of the independent variable, x is independent variable.
y or f(x) is dependent variable, and is returned as return value.

行

vim 中的行以行结束标志（比如回车、换行）结尾。

当窗口中一行无法容纳一段文字，它们被放到“多行”时，vim 将这些行视作一行。

- ① Function `linear()` is simple, it is used to calculate general linear result.
- ②
- ③ `linear()` conforms to the folloing formula:
- ④ $y = f(x) = a + bx$
- ⑤ there are three parameters in function `linear()` which named as `a`, `b` and `x`
- ⑥ corresponding to the above formula.
- ⑦
- ⑧ Among these three parameters, `a` is the constant term or the `y` intercept,
- ⑨ `b` is the coefficient of the independent variable, `x` is independent variable.
- ⑩ `y` or `f(x)` is dependent variable, and is returned as return value.

行轨迹

如果想要 vim 按照行来移动光标，可以按下：

- ✦ j 移至下一行
- ✦ k 移至上一行
- ✦ 0 移至行首
- ✦ ^ 移至行中首个非空白字符
- ✦ \$ 移至行尾
- ✦ <RETURN> 移至下一行
- ✦ - 移至前一行的行首

```
      k
-linear() conforms to the folloing formula:  
      y = f(x) = a + bx $  
0  there are three parameters in function linear() . . .  
      ^ j / <RETURN>
```

文本对象选择

对于单词 (word)、句子 (sentence)、段落 (paragraph) 等对象, vim 还允许在引用它们时加上前缀 a 或者 i, 称之为文本对象选择。

以 a 开始选择对象时, 光标轨迹包括周围的空白。举例来说, 对于下面一行:

```
void do_something()
```

当光标位于 d 时, aw 引用的是 “ *do_something*”, 包括了 *do* 前面的空格。

与 a 相反, 以 i 开始选择对象时, 光标轨迹不包括周围的空白, 或者只有空白。