



# Keyboard Shortcuts in the IPython Shell

## IPython Shell中的键盘快捷键

If you spend any amount of time on the computer, you've probably found a use for keyboard shortcuts in your workflow. Most familiar perhaps are the Cmd-C and Cmd-V (or Ctrl-C and Ctrl-V) for copying and pasting in a wide variety of programs and systems. Power-users tend to go even further: popular text editors like Emacs, Vim, and others provide users an incredible range of operations through intricate combinations of keystrokes.

如果你已经使用计算机一段时间了，你会发现键盘快捷键在你的工作中经常会被用到。这里面最常用的莫过于Cmd-C和Cmd-V（或者Ctrl-C和Ctrl-V），用来复制和粘贴。熟练的用户可能走得更远：流行的文本编辑器如Emacs、Vim等会给用户提供更多的组合快捷键。

The IPython shell doesn't go this far, but does provide a number of keyboard shortcuts for fast navigation while typing commands. These shortcuts are not in fact provided by IPython itself, but through its dependency on the GNU Readline library: as such, some of the following shortcuts may differ depending on your system configuration. Also, while some of these shortcuts do work in the browser-based notebook, this section is primarily about shortcuts in the IPython shell.

IPython shell没有像上述的文本编辑器那么复杂，但是也提供了不少的快捷键能让用户在输入命令的时候提高工作效率。这些快捷键实际上并不是IPython本身提供的，是基于它所依赖的GNU Readline库提供的：因此，下面介绍的某些快捷键可能会根据你的系统设置不同而发生改变。虽然本小节介绍的一些快捷键也在浏览器中的notebook应用，但是目前我们聚焦在IPython shell上。

Once you get accustomed to these, they can be very useful for quickly performing certain commands without moving your hands from the "home" keyboard position. If you're an Emacs user or if you have experience with Linux-style shells, the following will be very familiar. We'll group these shortcuts into a few categories: *navigation shortcuts*, *text entry shortcuts*, *command history shortcuts*, and *miscellaneous shortcuts*.

一旦你习惯了这些快捷键，你会发现它们能大大提高你在shell中输入命令的效率，甚至在你的手指不需要离开键盘主位置的情况下。如果你是一个Emacs编辑器的用户，或者是一个Linux shell的用户，以下内容对你来说不会陌生。我们将这些快捷键分为几组：[导航快捷键](#)，[文字输入快捷键](#)，[命令历史实现快捷键](#)以及[杂项快捷键](#)。

## Navigation shortcuts

### 导航快捷键

While the use of the left and right arrow keys to move backward and forward in the line is quite obvious, there are other options that don't require moving your hands from the "home" keyboard position:

Keystroke	Action
Ctrl-a	Move cursor to the beginning of the line
Ctrl-e	Move cursor to the end of the line
Ctrl-b or the left arrow key	Move cursor back one character
Ctrl-f or the right arrow key	Move cursor forward one character

显然使用左右箭头键来在一行命令中前后移动是很明显的，但是也有其他的选择让你无需将手移动到主键盘位置之外：

按键	动作
Ctrl-a	将光标移动到本行开始位置
Ctrl-e	将光标移动到本行结束位置
Ctrl-b 或者 左箭头	将光标向左移动一个字符
Ctrl-f 或者 右箭头	将光标向右移动一个字符

译者注：如果你熟悉BASH，这四个快捷键一定不陌生。

## Text Entry Shortcuts

### 文字输入快捷键

While everyone is familiar with using the Backspace key to delete the previous character, reaching for the key often requires some minor finger gymnastics, and it only deletes a single character at a time. In IPython there are several shortcuts for removing some portion of the text you're typing. The most immediately useful of these are the commands to delete entire lines of text. You'll know these have become second-nature if you find yourself using a combination of Ctrl-b and Ctrl-d instead of reaching for Backspace to delete the previous character!

Keystroke	Action
Backspace key	Delete previous character in line
Ctrl-d	Delete next character in line
Ctrl-k	Cut text from cursor to end of line
Ctrl-u	Cut text from beginning of line to cursor
Ctrl-y	Yank (i.e. paste) text that was previously cut
Ctrl-t	Transpose (i.e., switch) previous two characters

我们都知道使用回退键可以删除前一个字符，去按下这个键有时也需要将手移出主键盘位置，而且这个键每次只能删除一个字符。在IPython中，有一些快捷键可以删除部分你正在输入的文字。这其中最有用的可能就是删除整行文字。当你熟练之后，你也可能本能的使用Ctrl-b和Ctrl-d来代替回退键。

按键	动作
回退键	删除光标前一个字符
Ctrl-d 或者 删除键	删除光标所在字符
Ctrl-k	剪切光标所在位置直至末尾的字符
Ctrl-u	剪切开头直至光标所在前一个位置的字符
Ctrl-y	粘贴字符到光标所在位置
Ctrl-t	交换光标前一位置和光标所在位置的字符

## Command History Shortcuts

### 命令历史快捷键

Perhaps the most impactful shortcuts discussed here are the ones IPython provides for navigating the command history. This command history goes beyond your current IPython session: your entire command history is stored in a SQLite database in your IPython profile directory. The most straightforward way to access these is with the up and down arrow keys to step through the history, but other options exist as well:

Keystroke	Action
Ctrl-p (or the up arrow key)	Access previous command in history
Ctrl-n (or the down arrow key)	Access next command in history
Ctrl-r	Reverse-search through command history

本小节讨论的快捷键中，可能提供在命令历史中导航的快捷键最令人震撼。命令的历史不仅仅是当前的IPython会话有效，所有的命令历史都会被记录到一个SQLite的数据库中，保存在你的IPython配置目录下。最直接使用命令历史的方法就是向上的箭头和向下的箭头，下表列示了命令历史的快捷键：

按键	动作
Ctrl-p 或者 上箭头	获取上一条命令历史
Ctrl-n 或者 下箭头	获取下一条命令历史
Ctrl-r	反向搜索命令历史

The reverse-search can be particularly useful. Recall that in the previous section we defined a function called `square`. Let's reverse-search our Python history from a new IPython shell and find this definition again. When you press Ctrl-r in the IPython terminal, you'll see the following prompt:

反向搜索有时会非常有用。回忆一下上一节中我们定义了一个函数名叫 `square`。让我们在IPython shell中使用命令历史回查这个函数的定义。当我们在IPython终端中按下Ctrl-r时，你会看到如下的提示符：

```
In [1]: (reverse-i-search)␣':
```

If you start typing characters at this prompt, IPython will auto-fill the most recent command, if any, that matches those characters:

如果你在这个提示符下输入，IPython会根据你输入的部分内容自动补充最近使用的命令：

```
In [1]: (reverse-i-search)␣sqa': square??
```

At any point, you can add more characters to refine the search, or press Ctrl-r again to search further for another command that matches the query. If you followed along in the previous section, pressing Ctrl-r twice more gives:

在这种情况下，你还可以输入更多的字符来精准搜索，或者继续按键Ctrl-r来查找下一个（更早）能匹配的命令。如果你输入了上例中的 `sqa`，再按下一次Ctrl-r会得到：

```
In [1]: (reverse-i-search)␣sqa': def square(a):
      """Return the square of a"""
      return a ** 2
```

Once you have found the command you're looking for, press Return and the search will end. We can then use the retrieved command, and carry-on with our session:

一旦你找到了你需要的命令，敲击回车将结束反向搜索。然后你就能使用找到的命令继续了：

```
In [1]: def square(a):
      """Return the square of a"""
      return a ** 2

In [2]: square(2)
Out[2]: 4
```

Note that Ctrl-p/Ctrl-n or the up/down arrow keys can also be used to search through history, but only by matching characters at the beginning of the line. That is, if you type `def` and then press Ctrl-p, it would find the most recent command (if any) in your history that begins with the characters `def`.

注意Ctrl-p/Ctrl-n或者上箭头/下箭头键也可以用来进行反向搜索，但是仅能匹配命令开头的那些字符。意思是，如果你输入 `def` 然后键入Ctrl-p或者向上箭头，IPython会试图寻找最新的一条命令历史，并且以 `def` 开头。

## Miscellaneous Shortcuts

### 杂项快捷键

Finally, there are a few miscellaneous shortcuts that don't fit into any of the preceding categories, but are nevertheless useful to know:

Keystroke	Action
Ctrl-l	Clear terminal screen
Ctrl-c	Interrupt current Python command
Ctrl-d	Exit IPython session

最后，还有一些杂项的快捷键不属于上述的组别中，但是也挺有用：

按键	动作
Ctrl-l	清除终端窗口内容
Ctrl-c	终止当前的Python语句执行
Ctrl-d	退出iPython会话

The Ctrl-c in particular can be useful when you inadvertently start a very long-running job.

Ctrl-c当你在不小心运行了一个非常花时间（或者无限循环）的任务时会很有用。

While some of the shortcuts discussed here may seem a bit tedious at first, they quickly become automatic with practice. Once you develop that muscle memory, I suspect you will even find yourself wishing they were available in other contexts.

虽然本节列出的快捷键看起来很冗余，但是很快你会发现它们在实践中的作用。一旦你形成了肌肉记忆，你甚至会希望在其他环境中也能使用它们。

