**Dictionaries as data structures**

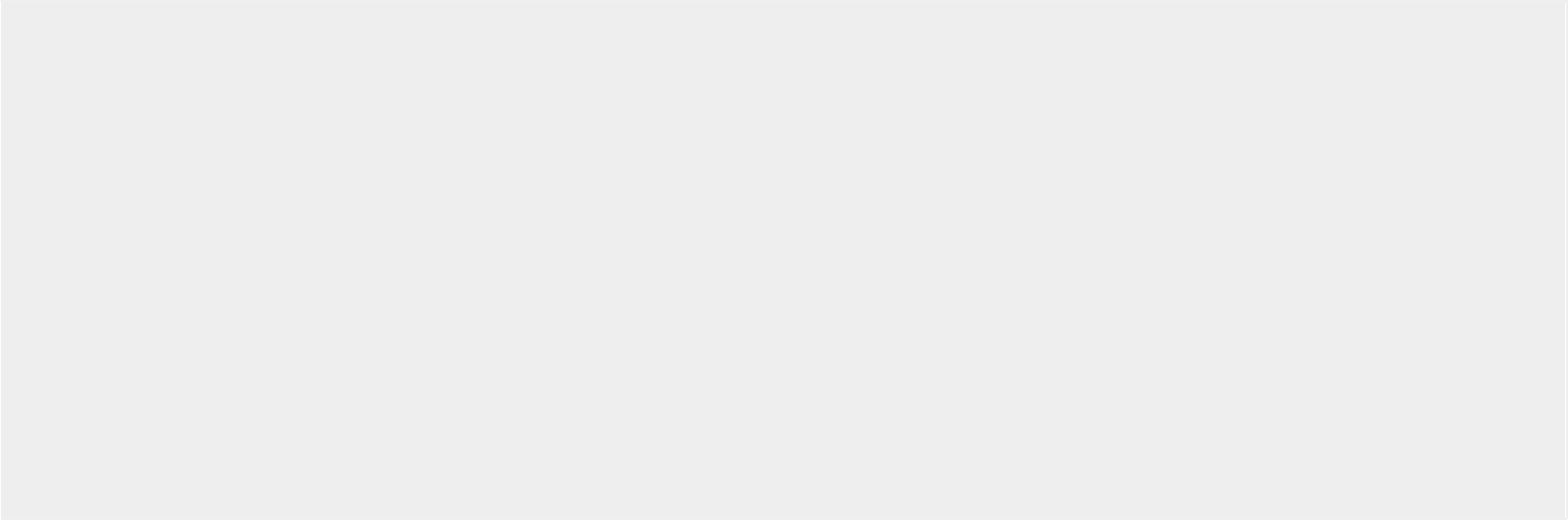
使用以下Vim内置的排序命令，可以删除文件中的重复行：



:%sort u

但是原始行的顺序，也将随之改变。

**Listing 8. A function for order-preserving uniqueness**



function! Uniq () range

* Nothing unique seen yet...

let have\_already\_seen = {} let unique\_lines = []

* Walk through the lines, remembering only the hitherto-unseen ones...

for original\_line in getline(a:firstline, a:lastline) let normalized\_line = '>' . original\_line

if !has\_key(have\_already\_seen, normalized\_line) call add(unique\_lines, original\_line)

let have\_already\_seen[normalized\_line] = 1 endif

endfor

* Replace the range of original lines with just the unique lines...

exec a:firstline . ',' . a:lastline . 'delete' call append(a:firstline-1, unique\_lines)

endfunction

以下自定义Uniq()函数，将利用字典数据结构来去除重复行，同时保留原始的行顺序。

unction is declared to take a range, so it will only be called once, even whe on a range of lines in the buffer.

首先，新建一个空字典”have\_already\_seen”) to track which lines have already been encountered within the specified range.

Lines that haven't seen before will then be added to the listunique\_lines.

然后在循环中，It grabs the specified range the buffer withgetline() and iterates through each. It first adds'>' leadingtoeach line to ensure it is not empty. Vimscript dictionaries cannot store an entry whose key is an emp so empty lines from the buffer would not be correctlyhave\_already\_seen.

Once the line is normalized, the function then checks whether that line has already been a key in thehave\_already\_seen dictionary. If so, an identical line must already have been s and added tounique\_lines , so the copy can be ignored. Otherwise, the line is being encoun for the first time, so the original (un-normalized) line mustunique\_lines, toand the normalized version must be added as a keyhave\_already\_seen.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| When all | the lines have been filtered in |  | will contain only the unique subset |  |
| unique\_lines |  |
| of them, | in the order in which they were first encountered. All that remains is to delet | | |  |

set of lines and replace it append() ) with these accumulated unique lines.

使用以下命令，可以针对整个文件执行函数：



:%call Uniq()

Or you could apply it to a specific set of lines (for example, a range that had been sel Visual mode), like so:



vmap u :call Uniq()<CR>