|  |
| --- |
| Here strings is just like [here documents](https://bash.cyberciti.biz/guide/Here_documents) and syntax is:  command <<<$word  OR  command arg1 <<<"$word"  The $word (a shell [variable](https://bash.cyberciti.biz/guide/Variables)) is expanded and supplied to the command on its [standard input](https://bash.cyberciti.biz/guide/Standard_input). The following [wc command](https://bash.cyberciti.biz/guide/Wc_command) will count words from given argument:  wc -w <<< "This is a test."  Sample outputs:  4 |

|  |
| --- |
| Read –ar  -a array assign the words read to sequential indices of the array  variable ARRAY, starting at zero  -r do not allow backslashes to escape any characters  -s do not echo input coming from a terminal |

|  |  |
| --- | --- |
|  | You can set the [internal field separator](http://en.wikipedia.org/wiki/Internal_field_separator) (IFS) variable, and then **let it parse into an array**. When this happens in a command, then the assignment to IFS only takes place to that single command's environment (to read ). It then parses the input according to the IFS variable value into an array, which we can then iterate over.  IFS=';' read -ra ADDR <<< "$IN"  for i in "${ADDR[@]}"; do  # process "$i"  done  It will parse one line of items separated by ;, pushing it into an array. Stuff for processing whole of $IN, each time one line of input separated by ;:  while IFS=';' read -ra ADDR; do  for i in "${ADDR[@]}"; do  # process "$i"  done  done <<< "$IN" |