Learn Plover!

Lesson 7: Non-Letters

Contents

1 Punctuation Strokes
2 Useful Strokes

Punctuation Strokes

So far you've learned how to stroke a vast number of words, but without punctuation it's still more of a parlor trick than a useful skill. There are a lot of punctuation marks in Plover. To get started, just learn the ones you find yourself needing:

- '`': KH-FG (Example: **'files = `/bin/ls`'** is stroked TPAOEULS KWA*LS KH-FG OEU PWEUPB OEU HR* S* KH-FG)
- '^': KR-RT (Example: **'x^5 is equal to x^2 times x^3.'** is stroked KP* KR-RT #A S KWAL TO KP* KR-RT #T- TAOEUPLS KP* KR-RT #P-)
- '~': T*LD. The tilde with a space after it. (Example: **'Yaaayy~ baby!'** is stroked KWR* A* A* A* KWR* KWR* T*LD PWAEUB KWREU SKHRAPL)
- '<': AEPBGT (Example: **'x<10'** is stroked KP* AEPBGT #SO)
- '=': KWA*LS (Example: **'x=10'** is stroked KP* KWA*LS #SO)
- '>': A*EPBGT (Example: **'x>10'** is stroked KP* A*EPBGT #SO)
- '|': PAO*EUP. The vertical bar, with no space after it. (Example: **'ls -R | less'** is stroked HR* S* S-P H-PB R*P S-P PAO*EUP S-P HRES)
- '_': R*UPBD. An underscore character, with no space after it. (Example: **'camelCase vs. under_scores'** is stroked KAPL/EL KPA* KAEUS SR* S* P-P S-P TPH-P R*UPBD SKORS)
- '-': H-PB. A single hyphen, with no space after it. (Example: **'It was a bit-coin moment'** is stroked EUT WAS AEU PWEUT H-PB KOEUPB PHOEPLT)
- '--': TK-RB. An em-dash (i.e. two hypens), with a space after it. (Example: **'Wait -- I'm going too'** is stroked WAEUT TK-RB AOEUPL TKPWOG TAO)
- ',': KW-BG. A comma, with a space after it. (Example: **'Be good, and be kind'** is stroked -B TKPWAOD KW-BG APBD -B KAOEUPBD)
- ';': SKWR*RBGS. A semicolon, with a space after it, and the next word uncapitalized. (Example: **'Be good; be kind'** is stroked -B TKPWAOD SKWR*RBGS KAOEUPBD)
- ':': STPH-FPLT. A colon, with a space after it, and the next word is capitalized. (Example: **'Plover: The winnah'** is stroked PHRO*FR STPH-FPLT W* EU* TPH* TPH* A* H*)
- ':': KHR-PB. A colon, with no space after it, and the next word is uncapitalized. Primarily used for specifying clock time. (Example: **'24:00'** may be stroked #TH KHR-PB #0D)
- '!': SKHRAPL. An exclamation point, with a space after it, and the next word is capitalized. (Example: **'Why? Because!'** is stroked KWR H-F BWAUS SKHRAPL)
- '?': H-F. A question mark, with a space after it, and the next word is capitalized. (Example: **'Why? Why not?'** is stroked KWR H-F KWR PHOT H-F)
- '/': OEU (Example: **'x/y=10'** is stroked KP* OEU KWR* KWA*LS #SO)
- '.': TP-PL. A regular period, with a space after it, and the next word is capitalized. (Example: **'There it is. Better'** is stroked THR EUT S TP-PL PWET/ER)
- '.': P-P. A period used as a decimal point, with no space after it, and the next word uncapitalized. (Example: **'It's.here.for.you'** is stroked T-S P-P HAOER P-P TPOR P-P U P-P)
- '...': SKWR-RBGS. Ellipses, with a space after it, and the next word is capitalized. (Example: **'No... Just no...'** is stroked TPHO SKWR-RBGS SKWRUFT TPHO SKWR-RBGS)
- ''': A*E. Open single quote, with no space after it.
- ''': AE. Close single quote, with a space after it. (Example: **'Just say 'hi' and see what happens!'** is stroked SKWRUFT SAEU A*E HEU AE APBD SAOE WHA HAPS SKHRAPL)
- '"': KW-GS. Open double quote, with no space after it.
- '"': KR-GS. Close double quote, with a space after it. (Example: **'I think "love" can be hard to say.'** is stroked EU THEU KW-GS HROF KR-GS K -B HARD TO SAEU TP-PL)
- '(': PREPB
- ')': PR*EPB (Example: **'I (love you) want some milk'** is stroked EU PREPB HROF U PR*EPB WAPBT SOPL PH*EULG TP-PL)
- '[': PWR-BGT
- ']': PWR*BGT (Example: **'I luv [sic] you.'** is stroked EU HR* *U SR* PWR-BGT S*EUBG PWR*BGT U TP-PL)
- '{': TPR-BGT
- '}': TPR*BGT (Example: **'if(x) { print "hello, world" }'** is stroked TP PREPB KP* PR*EPB TPR-BGT PREUPBT KW-GS HEL/HRO KW-BG WORLD)
- '@': KWRAT (Example: **'zacharyb@gmail.com'** is stroked STKPW* A* KR* H* A* R* KWR* PW* KWRAT TKPW* PH* A* EU* HR* P-P KR* O*)

- '$' `TK`-`PL` (Example: '**I got the $$$. You got the stuff?**' is stroked `EU` `TKPWOT` -`T` `TK`-`PL` `TK`-`PL` `TK`-`PL` `TP`-`PL` `U` `TKPWOT` -`T` `STUF` `H`-`F`)
- '*': `STA*R` (Example: '**Love you babe :-* :-***' is stroked `HRUF` `U` `PWAEUB` `KHR`-`PB` `H`-`PB` `STA*R` `S`-`P` `KHR`-`PB` `H`-`PB` `STA*R`)
- '\': `SPWHRAERB` (Example: '**C:**' is stroked `KR*P` `KHR`-`PB` `SPWHRAERB`)
- '&': `SP`-`PBD` (Example '**U R my <3 & soul**' is stroked `U*P` `S`-`P` `R*P` `PHAOEU` `AEPBGT` `#P`- `S`-`P` `SP`-`PBD` `SOUL`)
- '#': `HAERB` (Example: '**You're my #1 guy!**' is stroked `UR` `PHAOEU` `HAERB` `#S` `TKPWAOEU` `SKHRAPL`)
- '%': `PERS` (Example: '**What % is that? 5%?**' is stroked `WHA` `PERS` `S` `THA` `H`-`F` `#A` `PERS` `H`-`F`)
- '+': `PHR*US` (Example: '**y=x+10**`' is stroked `KWR*` `KWA*LS` `KP*` `PHR*US` `#SO`)

If your dictionary file is missing any punctuation marks, add a dictionary entry for that mark, using the definitions above. See [Appendix: The Dictionary Format](#) for instructions on how to do that.

Useful Strokes

- Carriage Return: `R`-`R`
- New paragraph. Insert two carriage returns and capitalize the next word: `SKWRAURBGS`
- Undo stroke: `*`
- Letter-by-letter backspace: `PW`-`FP`
- Insert a space: `S`-`P`
- Insert a space and capitalize the next word: `KPA`
- Do not insert a space before the next word, and don't capitalize the next word `TK`-`LS`
- Do not insert a space before the next word, and capitalize the next word `KPA*`

Plover takes care of a lot of spacing and capitalization, but it can't detect when you want a new paragraph. For that, you still need to a carriage return. If you're using a converted Qwerty keyboard with Plover, you may wonder why Plover bothers providing a whole keystroke for the carriage return, when the original carriage return key is right there on your keyboard. Why learn something new when the old thing is so easily accessible? The reason for this will become clearer and clearer as your typing speed increases. Keeping your fingers at or near their home position is very important, both for typing speed and hopefully for reducing repetitive strain injuries.

Even more convenient is Plover's ability to map keystrokes to just about anything you need. The `SKWRAURBGS` chord isn't intended to sound like anything in particular - it's just a pretty easy stroke to remember. It creates a blank line with two carriage returns, and capitalizes the first word of the new paragraph.

The `*` key is heavily used as a disambiguator in Plover. When used alone, however, the `*` key undoes the most recent keystroke. If you press it repeatedly, you can remove up to a dozen or so keystrokes this way. If you're in the middle of typing a multi-stroke word and accidentally press the wrong chord, the `*` key will remove the unintended stroke, and still remember where you are in the multi-stroke word.

As with the carriage return, the original backspace key is not used in Plover, but the `PW`-`FP` stroke accomplishes the same behavior, without the fingers having to leave the home position. And even though Plover outputs complete words, there are still occasions when deleting individual characters is useful.

Plover inserts spaces automatically between words and around punctuation as needed; but you may still need to insert spaces yourself from time to time. In particular, if you're dealing with a word boundary error, inserting a space with `S`-`P` is an appropriate way to force a word division where you want it. For example, the word '**sunlight**' is stroked `SUPB`/`HRAOEUT`. But what if you want to type the two words '**sun light**' separately? The word '**sun**' is stroked `SUPB`, and the word '**light**' is stroked `HRAOEUT`. If you type them one after the other, Plover will think you mean '**sunlight**'. To make sure that doesn't happen, you should manually add a space between the words. So '**sun light**' is stroked `SUPB` `S`-`P` `HRAOEUT`.

Another use for `S`-`P` is if you're fingerspelling something that you doubt you'll use often enough to warrant adding its own dictionary entry. In that case, you may want to use `S`-`P` to include spaces in what you fingerspell.

`KPA*` is used for [camelcase](#), in other words for a series of words that have no spaces between them, but where each word in the series must be capitalized. For instance, if you wanted to write '**YouTube**', it would be `KPA` `U` `KPA*` `TAOUB`. The initial `KPA` ensures there is a space before '**YouTube**' and that the '**Y**' will be capitalized, while the `KPA*` ensures that the '**T**' will be capitalized with *no* preceding space.

Or, if you're writing software, a variable named **palindromeWordCount**' would be stroked `PAL`/`*EUPB`/`TKROEPL` `KPA*` `WORD` `KPA*` `KOUPBT`.