Bash Keyboard Shortcuts

Key shortcuts are shown as capital to match your keyboard but are hit without <Shift> or <Caps>. <Esc> will sometimes replace <Alt> that is already attributed to another function.

### Process control:
- `<Ctrl>`+`C` - Interrupt/Kill whatever you are running (SIGINT)
- `<Ctrl>`+`L` - Clear the screen
- `<Ctrl>`+`S` - Stop output to the screen (for long running verbose commands)
- `<Ctrl>`+`Q` - Allow output to the screen (if previously stopped using command above)
- `<Ctrl>`+`D` - Send an EOF marker, unless disabled by an option, this will close the current shell (EXIT)
- `<Ctrl>`+`Z` - Send the signal SIGTSTP to the current task, which suspends it. To return to it later enter `fg` 'process name' (foreground).

### Copy & Paste text between windows:
Highlight text with the mouse then move to terminal and hit `<Shift>`+`<Ins>` to paste the text (under Apple OSX, press CMD+`V`). Users with a mouse equipped with a wheel can also press the `wheel-button>` to paste the highlighted text. Finally, `Shift+click` allows copying / pasting text from / to other windows.

### Moving the cursor:
- `<Ctrl>`+`A` - Go to the beginning of the line (Home)
- `<Ctrl>`+`E` - Go to the End of the line (End)
- `<Ctrl>`+`P` - Previous command (Up arrow)
- `<Ctrl>`+`N` - Next command (Down arrow)
- `<Alt>`+`B` - Back (left) one word
- `<Alt>`+`E` - Backward one character
- `<Alt>`+`F` - Forward (right) one word
- `<Ctrl>`+`F` - Forward one character
- `<Ctrl>`+`X` - Toggle between the start of line and current cursor position

### Editing:
- `<Shift>`+`<Ins>` - Clear the Screen, similar to the clear command
- `<Ctrl>`+`W` - Cut/Delete the line before the cursor position.
- `<Alt>`+`<Del>` - Delete the Word before the cursor.
- `<Alt>`+`D` - Delete the Word after the cursor.
- `<Ctrl>`+`D` - Delete character under the cursor.
- `<Ctrl>`+`H` - Delete character before the cursor (Backspace)
- `<Ctrl>`+`W` - Cut the Word before the cursor to the clipboard.

- `<Ctrl>`+`K` - Cut the Line after the cursor to the clipboard.
- `<Alt>`+`T` - Swap current word with previous
- `<Ctrl>`+`T` - Swap the last two characters before the cursor (typo).
- `<Esc>`+`+` - Swap the last two words before the cursor.
- `<Alt>`+`Y` - Paste the last thing to be cut (yank)
- `<Alt>`+`L` - Paste the last thing to be cut (yank)
- `<Alt>`+`U` - Upper capitalize every character from the cursor to the end of the current word.
- `<Alt>`+`C` - Capitalize the character under the cursor and move to the end of the word.
- `<Ctrl>`+`+` - Cancel the changes and put back the line as it was in the history (revert).
- `<Tab>` - Undo

- `<Tab>` - TAB completion for file/directory names. For example, to move to a directory 'sample1': Type cd sam; then press `<Tab>` and ENTER. Type just enough characters to uniquely identify the directory you wish to open.

- TAB completion does not work for environment variables with early bash 4 versions. To fix this you can type the variable name (eg `$HOME`), then hit <Esc> followed by <Tab>. This should have expanded the variable to its value and can proceed with <Tab> completion.

### History:
- `<Ctrl>`+`R` - Recall the last command including the specified character(s) (equivalent to : vim ~/.bash_history).
- `<Ctrl>`+`T` - Previous command in history (i.e. walk back through the command history)
- `<Ctrl>`+`N` - Next command in history (i.e. walk forward through the command history)
- `<Ctrl>`+`S` - Go back to the next most recent command. (beware to not execute it from a terminal because this will also launch its XOFF).
- `<Ctrl>`+`O` - Execute the command found via `<Ctrl>`+`R` or `<Ctrl>`+`S`
- `<Ctrl>`+`G` - Escape from history searching mode
- `<Ctrl>`+`T` - Repeat last command starting with abc
- `<Ctrl>`+`P` - Print last command starting with abc
- `<Esc>` - Last argument of previous command
- `<Alt>`+`<` - Last argument of previous command
- `<Alt>`+`>` - All arguments of previous command
- `<Alt>`+`<` - Run previous command, replacing abc with def