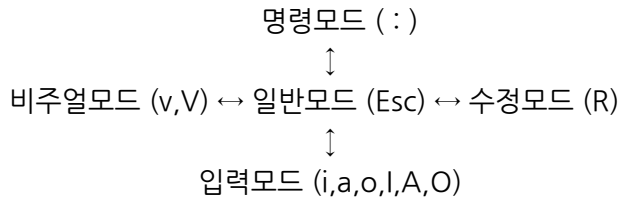


Vim Cheat Sheet for Beginners

0. 모드



[일반모드]

- Vim 시작시 자동 진입
- 다른 모드에서 Esc 로 진입

[일반모드 → 입력모드]

- i - 커서부터 입력
- a - 커서 다음부터 입력
- o - 아랫 줄을 만들고 입력
- I - 줄의 글자가 있는 가장 처음에 입력 // = 커서가 첫 글자의 앞
- A - 줄의 가장 마지막에 입력 // = 커서가 EOL 의 앞
- O - 윗 줄을 만들고 입력

[일반모드 → 명령모드]

- 명령모드 진입

(내린 명령을 최종적으로 실행하려면 Enter 를 눌러야 함)

(터미널에서 “vi 파일이름”으로 vim 을 실행한 경우)

:w Enter - 저장

:wq Enter - 저장 및 종료 // ZZ 로도 가능

(터미널에서 단순히 “vi”로 vim 을 실행한 경우)

:w 파일이름 Enter - 파일이름으로 저장

:wq 파일이름 Enter - 저장 및 종료 // 파일이름을 입력해야 하기에 ZZ 불가능

(저장이 된 상태에서)

:q Enter - vim 종료

(저장이 안 된 상태에서)

:q! Enter - 마지막 저장 상태 이후의 수정은 반영 안하고 vim 종료 // ZQ 로도 가능

/ 모든 명령어는 대문자와 소문자를 구분함 */**

[일반모드 → 비주얼모드]

- v - 글자 단위로 선택
- V - 줄 단위로 선택

[일반모드 → 수정모드]

- R - 수정모드 진입

/ 이 줄 이후 모든 명령어는 일반모드(Esc)에서 사용 */**

1. 커서 이동

[방향키] (직관적이지 못하지만 익숙해져야 한다)

	k	
h		l
	j	

[점프키]

0 - 줄의 가장 처음으로 이동 (hard begin)

\$ - 줄의 가장 마지막(EOL 전)으로 이동 (hard end)

^ - 줄의 첫단어의 가장 처음으로 이동

w - 단어 첫 글자로 이동, 다음 단어 첫 글자로 이동

W - 단어 첫 글자로 이동, (공백 기준) 다음 단어 첫 글자로 이동

e - 단어 마지막 글자로 이동, 다음 단어 마지막 글자로 이동

E - 단어 마지막 글자로 이동, (공백 기준) 다음 단어 마지막 글자로 이동

/ \$, w, e 는 (뒤로 탐색하기에) 명령어 조합에서 자주 사용됨 */**

b - 단어 첫 글자로 이동, 이전 단어 첫 글자로 이동

B - 단어 첫 글자로 이동, (공백 기준) 이전 단어 첫 글자로 이동

gg - 파일의 가장 첫 줄 첫 단어의 앞으로 이동

G - 파일의 가장 마지막 줄 첫 단어의 앞으로 이동

ge - 단어 마지막 글자로 이동, 이전 단어 마지막 글자로 이동

gE - 단어 마지막 글자로 이동, (공백 기준)이전 단어 마지막 글자로 이동

Vim Cheat Sheet for Beginners

2. 잘라내기 (한 글자) (추후 불허넣지 않으면 삭제와 사실상 같음)

x - 커서 오른쪽의 한 글자 잘라내기 (=Del)

X - 커서 왼쪽의 한 글자 잘라내기 (=BS)

x, X는 숫자와 조합이 가능

2x - 커서의 한 글자와 그의 다음 한 글자까지 글자 2 개 잘라내기

2X - 커서 왼쪽의 한 글자와 그의 왼쪽 한 글자까지 글자 2 개 잘라내기

3. 잘라내기 (여러 글자) (추후 불허넣지 않으면 삭제와 사실상 같음)

d - 잘라내기

(d 하나만으로는 효과가 없으며, 조합을 해야 함)

d\$ - 현재 커서부터 EOL 전까지 잘라내기

dw - 현재 커서부터 다음 단어 이전의 모든 것 잘라내기 (공백 포함)

de - 현재 커서부터 현재 단어의 마지막까지 잘라내기

dd - 커서가 있는 줄 전체를 잘라내기

d는 숫자와 조합이 가능

2d\$ - 현재 커서부터 다음 줄의 EOL 전까지 잘라내기 (현재 줄 EOL 포함)

2dw - 현재 커서부터 다다음 단어 이전의 모든 것 잘라내기 (공백 포함)

2de - 현재 커서부터 다음 단어의 마지막까지 잘라내기 (공백 포함)

2dd - 커서가 있는 줄과 그 아랫줄 전체를 잘라내기 (현재 줄 EOL 포함)

4. 불허넣기

p - 커서 뒤에 불허넣기

P - 커서 위치에 불허넣기

p,P는 숫자와 조합이 가능

2p - 커서 뒤에 두 번 불허넣기

2P - 커서 위치에 두 번 불허넣기

/ 모든 명령어는 대문자와 소문자를 구분함 */**

5. 복사

y - 복사

(y 하나만으로는 효과가 없으며, 조합을 해야 함)

y\$ - 현재 커서부터 EOL 전까지 복사

yw - 현재 커서부터 다음 단어 이전의 모든 것 복사 (공백 포함)

ye - 현재 커서부터 현재 단어의 마지막까지 복사

yy - 커서가 있는 줄 전체를 복사

y는 숫자와 조합이 가능

2y\$ - 현재 커서부터 다음 줄의 EOL 전까지 복사 (현재 줄 EOL 포함)

2yw - 현재 커서부터 다다음 단어 이전의 모든 것 복사 (공백 포함)

2ye - 현재 커서부터 다음 단어의 마지막까지 복사 (공백 포함)

2yy - 커서가 있는 줄과 그 아랫줄 전체를 복사 (현재 줄 EOL 포함)

6. 실행 취소

u - 직전실행 취소

Ctrl + r - 직전실행 취소의 취소

U - 마지막으로 수정한 줄의 모든 변경사항 원상복구

7. 검색

/검색어 Enter - 커서 이후의 가장 가까운 검색어 검색

?검색어 Enter - 커서 이전의 가장 가까운 검색어 검색

(검색된 상태에서)

n - 검색 결과 이후 탐색

N - 검색 결과 이전 탐색

n, N은 숫자와 조합이 가능

n - 현재 검색된 상태에서 두 개 이후 탐색

N - 현재 검색된 상태에서 두 개 이전 탐색

Vim Cheat Sheet for Beginners

8. 대체 (한 글자)

r 대체할문자 - 커서의 한 글자를 대체할문자로 대체

r은 숫자와 조합이 가능

2r 대체할문자 - 커서와 커서 다음의 두 글자를 대체할 문자로 대체

9. 대체 (여러 글자)

c - 대체

(c 하나만으로는 효과가 없으며, 조합을 해야 함)

c\$ - 현재 커서부터 EOL 전까지 삭제 후 삽입

cw - 현재 커서부터 다음 단어 이전의 모든 것 삭제 후 삽입 (공백 포함)

ce - 현재 커서부터 현재 단어의 마지막까지 삭제 후 삽입

cc - 커서가 있는 줄 전체를 지운 후 삽입

c는 숫자와 조합이 가능

2c\$ - 현재 커서부터 다음 줄의 EOL 전까지 삭제 후 삽입 (현재줄 EOL 포함)

2cw - 현재 커서부터 다다음 단어 이전의 모든 것 삭제 후 삽입 (공백 포함)

2ce - 현재 커서부터 다음 단어의 마지막까지 삭제 후 삽입 (공백 포함)

2cc - 커서가 있는 줄과 그 아랫줄 전체를 지운 후 삽입

10. 검색 & 대체 (일반모드 → 명령모드)

- (아무것도 x) - 현재 줄에서
- 줄번호 - 특정 줄에서
- : 줄번호시작,끝 s/검색어/대체어/g - 특정 줄들에서
- % - 파일 전체에서

(마지막의 /g 를 /gc 로 바꾸면, 해당되는 부분을 바꿀지 매 번 물어봄)

/** 모든 명령어는 대문자와 소문자를 구분함 */

/** 11. 날코딩 시 트릭들 */

비주얼모드 - 시각적으로 영역 선택, 직관적이고 강력함

v - 글자 단위로 선택

시작 : 비주얼 모드에 진입할 때의 커서 위치부터

끝 : 방향키, 점프키 등 다양한 방법으로 이동한 커서 위치까지

V - 줄 단위로 선택

시작 : 비주얼 모드에 진입할 때의 커서가 있는 줄부터

끝 : 방향키, 점프키 등 다양한 방법으로 이동한 커서가 있는 줄까지

(비주얼모드로 선택한 상태에서)

= - 들여쓰기 자동 정렬

:줄번호 Enter - 해당 줄번호 첫 단어의 첫 글자로 이동
(일반모드에서 “줄번호 G”로도 가능하긴 하나 썩 편하지는 않아서…)

>> - Tab

<< - Shift + Tab

zz - 현재 커서 위치가 중앙이 되도록 문서를 스크롤

{ - 이전 단락으로 이동

} - 다음 단락으로 이동

% - 짝을 이루는 소, 중, 대괄호로 이동

H - 화면 기준 첫 행으로 커서를 이동

M - 화면 기준 중간 행으로 커서를 이동

L - 화면 기준 마지막 행으로 커서를 이동

Ctrl + f - 다음 페이지로 커서를 이동

Ctrl + b - 이전 페이지로 커서를 이동

J - 현재 커서가 있는 줄의 EOL 과 다음 줄의 첫 단어 이전의 모든 공백 삭제

Vim Cheat Sheet for Beginners

/ 모든 명령어는 대문자와 소문자를 구분함 */**

12. 블록 단위 편집 (일반모드 → 명령모드)

지금까지 설명하기로

d - 잘라내기 (여러 글자)

y - 복사

c - 대체 (여러 글자)

는 operator 하나로는 효과가 없으며

“(**op**) (\$ / w / e / **op**)”의 형태로 조합하여 사용한다 함

이 명령어들은

“(**블록 범위**) (**op**)”의 형태로도 사용 가능

[블록범위]

:줄번호 - 한 줄 선택

:줄번호시작,끝 - 여러 줄 선택

[줄번호]

n - n 번째 줄

. - 현재 커서가 있는 줄

0 - 문서의 가장 처음 줄 (1 과 사실상 같지만, 계산은 다름에 주의)

\$ - 문서의 가장 마지막 줄

줄번호는 “+ n”과 조합하여 상대적일 수 있음

+ - 아래로 (뒤로)

- - 위로 (앞으로)

x+n - x 번째 줄에서 아래로 n 번째 줄

x-n - x 번째 줄에서 위로 n 번째 줄

.+n - 현재 커서가 있는 줄에서 아래로 n 번째 줄

.-n - 현재 커서가 있는 줄에서 위로 n 번째 줄

0+n - 문서의 처음부터 n 번째 줄

\$-n - 문서의 마지막부터 n 번째 줄

(0-n, \$+n 은 생각해보면 상식적으로 말이 안됨)

ex1)

:1,3+7d

[블록범위] : 1 ~ 3+7 번째 줄

[시작] : 1 번째 줄

[끝] : 3+7 = 10 번째 줄

[op] : d - 잘라내기 (여러 글자)

ex2)

:4+7y

[블록범위] : 4+7 번째 줄

[시작=끝] : 4+7 = 11 번째 줄

[op] : y - 복사

ex3) (현재 커서가 77 번째 줄에 있다 가정)

:0,-55c

[블록범위] : 문서의 가장 처음 ~ 현재 커서가 있는 줄 - 55 번째 줄

[시작] : 문서의 가장 처음 줄 = 1 번째 줄

[끝] : 현재 커서 - 55 번째 줄 = 22 번째 줄

[op] : c - 대체 (여러 글자)

...다만 실 사용시 비주얼모드가 훨씬 편할 것임

Vim Cheat Sheet for Beginners

/ 모든 명령어는 대문자와 소문자를 구분함 */**

13. ~/.vimrc

홈 디렉터리에 .vimrc 파일을 만들어서, 해당 사용자의 vim 기본 옵션을 설정

~/.vimrc 에

```
set number " 줄번호 표시, number = nu
set ruler " 현재 커서가 있는 위치 표시
set showmatch " 짝을 이루는 괄호 강조
set scrolloff=2 " 2 줄 남았을 때 스크롤
set laststatus=2 " 상태바 항상 표시

set smartindent " c 포함 다양한 언어에서 똑똑하게 들여쓰기
set noexpandtab " /t 를 /sp 4 개로 대체하는 것을 방지
set tabstop=4 " /t 를 몇 개의 /sp 처럼 보이게 할지
set shiftwidth=4 " 자동/수동 들여쓰기 시 /sp 4 개만큼

set hlsearch " 검색 시 검색어 하이라이트
set incsearch " 점진적 검색 - 앞글자부터 일치하는거 줄여나감
set smartcase " 검색 시 대소문자 구분

set fileencoding=utf-8 " 파일 쓸 때 utf-8 인코딩
set encoding=utf-8 " 파일 읽을 때 utf-8 인코딩

set wildmenu " 명령모드에서 탭 시 명령어 힌트
set wildmode=longest:list,full

colorscheme default
```

와 같이 작성하고 저장

참고로 expandtab 옵션은 softtabstop 와 세트로 쓰임
Makefile 작성 시 /t 를 /sp 4 번으로 대체 할 경우 오류가 생기기에
위의 vimrc 에서는 이를 원천 차단함

복붙해야 하는데 줄번호가 표시되서 끄고싶으면

```
:set nonu
```

해주면 줄번호가 잠깐 사라짐, 다시 켜려면

```
:set nu
```