


vim 설치하기 (Vi IMproved)

- \$ sudo apt install vim 실행

```
ubuntu-virtual-machine:~$ sudo apt install vim
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  vim-runtime
Suggested packages:
  ctags vim-doc vim-scripts
The following NEW packages will be installed:
  vim vim-runtime
0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.
Need to get 7,111 kB of archives.
After this operation, 34.6 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://kr.archive.ubuntu.com/ubuntu focal/main amd64 vim-runtime all 2:8.1.2269-1ubuntu5 [5,873 kB]
Get:2 http://kr.archive.ubuntu.com/ubuntu focal/main amd64 vim amd64 2:8.1.2269-1ubuntu5 [1,238 kB]
Fetched 7,111 kB in 32s (224 kB/s)
Selecting previously unselected package vim-runtime.
(Reading database ... 151153 files and directories currently installed.)
Preparing to unpack .../vim-runtime_2%3a8.1.2269-1ubuntu5_all.deb ...
Adding 'diversion of /usr/share/vim/vim81/doc/help.txt to /usr/share/vim/vim81/doc/help.txt.vim-tiny by vim-runtime'
Adding 'diversion of /usr/share/vim/vim81/doc/tags to /usr/share/vim/vim81/doc/tags.vim-tiny by vim-runtime'
Unpacking vim-runtime (2:8.1.2269-1ubuntu5) ...
Selecting previously unselected package vim.
Preparing to unpack .../vim_2%3a8.1.2269-1ubuntu5_amd64.deb ...
Unpacking vim (2:8.1.2269-1ubuntu5) ...
Setting up vim-runtime (2:8.1.2269-1ubuntu5) ...
Setting up vim (2:8.1.2269-1ubuntu5) ...
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/vim (vim) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/vimdiff (vimdiff) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/rvim (rvim) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/rview (rview) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/vi (vi) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/view (view) in auto mode
update-alternatives: using /usr/bin/vim.basic to provide /usr/bin/ex (ex) in auto mode
Processing triggers for man-db (2.9.1-1) ...
```

vim 설정하기

- `/etc/vim/vimrc` 파일을 자신의 폴더에 `.vimrc`파일로 복사하기
 - vim 설정 파일

`$ cp /etc/vim/vimrc ~/.vimrc`

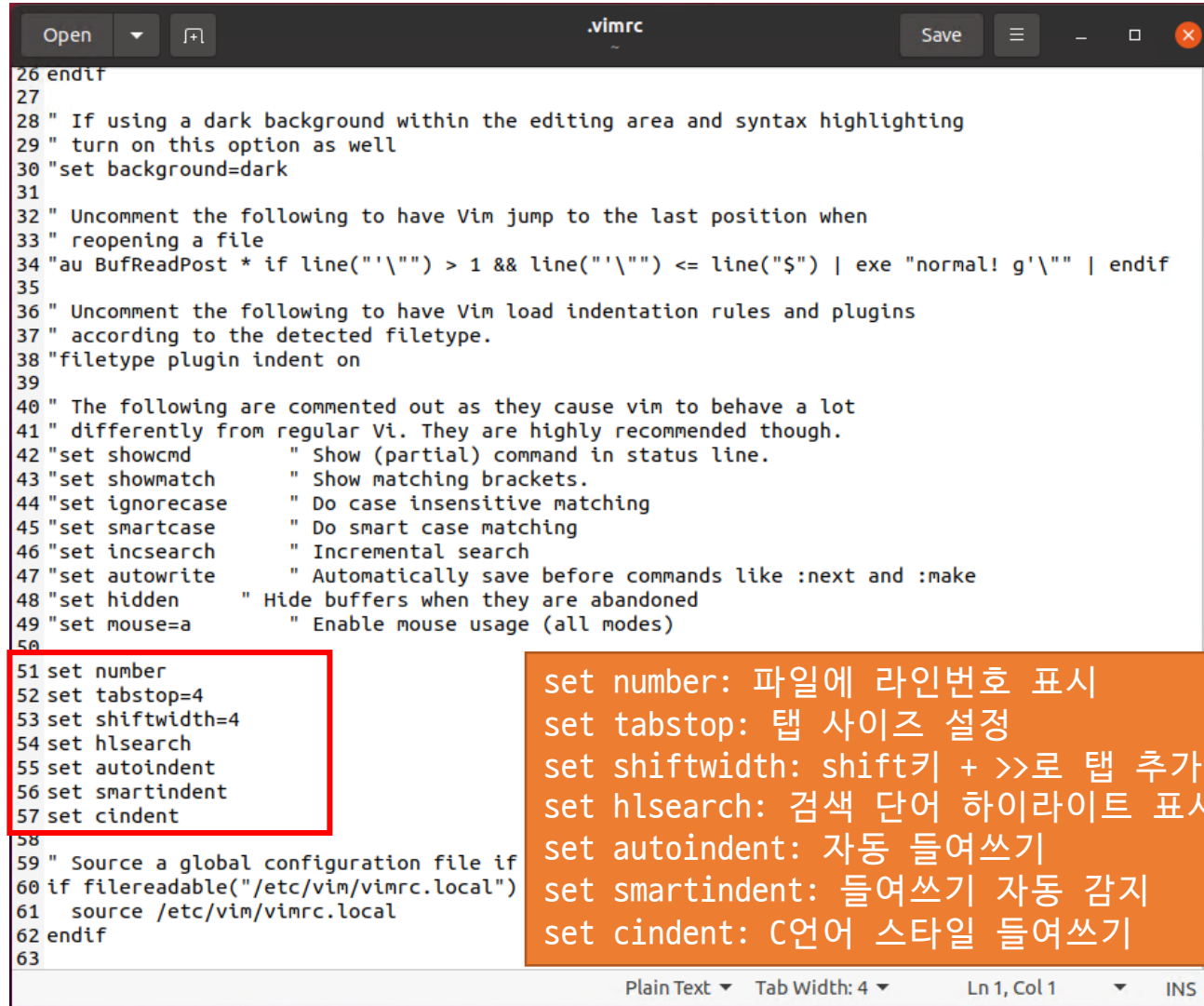
→ 내 홈 디렉토리.

- 전체 파일 확인하기: `$ ls -al`

```
changsu@changsu-virtual-machine:~$ ls -al
total 80
drwxr-xr-x 16 changsu changsu 4096 8월 30 15:34 .
drwxr-xr-x  3 root    root    4096 8월 30 14:39 ..
-rw-r--r--  1 changsu changsu  220 8월 30 14:39 .bash_logout
-rw-r--r--  1 changsu changsu 3771 8월 30 14:39 .bashrc
drwx----- 10 changsu changsu 4096 8월 30 15:03 .cache
drwx----- 12 changsu changsu 4096 8월 30 15:32 .config
drwxr-xr-x  2 changsu changsu 4096 8월 30 14:59 Desktop
drwxr-xr-x  2 changsu changsu 4096 8월 30 14:59 Documents
drwxr-xr-x  2 changsu changsu 4096 8월 30 14:59 Downloads
drwx-----  3 changsu changsu 4096 8월 30 15:15 .gnupg
drwxr-xr-x  3 changsu changsu 4096 8월 30 14:59 .local
drwxr-xr-x  2 changsu changsu 4096 8월 30 14:59 Music
drwxr-xr-x  2 changsu changsu 4096 8월 30 14:59 Pictures
-rw-r--r--  1 changsu changsu   807 8월 30 14:39 .profile
drwxr-xr-x  2 changsu changsu 4096 8월 30 14:59 Public
drwx-----  2 changsu changsu 4096 8월 30 15:15 .ssh
-rw-r--r--  1 changsu changsu     0 8월 30 15:22 .sudo_as_admin_successful
drwxr-xr-x  2 changsu changsu 4096 8월 30 14:59 Templates
drwxr-xr-x  2 changsu changsu 4096 8월 30 14:59 Videos
-rw-r--r--  1 changsu changsu 2266 8월 30 15:34 .vimrc
drwxrwxr-x  2 changsu changsu 4096 8월 30 15:31 workspace
```

.vimrc 파일 수정

- TextEdit에서 51라인부터 57라인까지 추가



```
26 endif
27
28 " If using a dark background within the editing area and syntax highlighting
29 " turn on this option as well
30 "set background=dark
31
32 " Uncomment the following to have Vim jump to the last position when
33 " reopening a file
34 "au BufReadPost * if line("'\"") > 1 && line("'\"") <= line("$") | exe "normal! g'\"" | endif
35
36 " Uncomment the following to have Vim load indentation rules and plugins
37 " according to the detected filetype.
38 "filetype plugin indent on
39
40 " The following are commented out as they cause vim to behave a lot
41 " differently from regular Vi. They are highly recommended though.
42 "set showcmd      " Show (partial) command in status line.
43 "set showmatch    " Show matching brackets.
44 "set ignorecase   " Do case insensitive matching
45 "set smartcase    " Do smart case matching
46 "set incsearch    " Incremental search
47 "set autowrite    " Automatically save before commands like :next and :make
48 "set hidden       " Hide buffers when they are abandoned
49 "set mouse=a      " Enable mouse usage (all modes)
50
51 set number
52 set tabstop=4
53 set shiftwidth=4
54 set hlsearch
55 set autoindent
56 set smartindent
57 set cindent
58
59 " Source a global configuration file if
60 if filereadable("/etc/vim/vimrc.local")
61   source /etc/vim/vimrc.local
62 endif
63
```

Handwritten red text: `:wq =>` 저장 후 종료

Orange box text:

- set number: 파일에 라인번호 표시
- set tabstop: 탭 사이즈 설정
- set shiftwidth: shift키 + >>로 탭 추가할 때
- set hlsearch: 검색 단어 하이라이트 표시
- set autoindent: 자동 들여쓰기
- set smartindent: 들여쓰기 자동 감지
- set cindent: C언어 스타일 들여쓰기

.bashrc 파일에 vim 연결

i: ↓

- vi 명령어로 vim 프로그램 실행
 - alias vi='vim' 내용 추가

91: alias ll='ls -aLF'
=wq

```
70 ;;
71 *)
72 ;;
73 esac
74
75 # enable color support of ls and also add handy aliases
76 if [ -x /usr/bin/dircolors ]; then
77     test -r ~/.dircolors && eval "$(dircolors -b ~/.dircolors)" || eval "$(dircolors -b)"
78     alias ls='ls --color=auto'
79     #alias dir='dir --color=auto'
80     #alias vdir='vdir --color=auto'
81
82     alias grep='grep --color=auto'
83     alias fgrep='fgrep --color=auto'
84     alias egrep='egrep --color=auto'
85 fi
86
87 # colored GCC warnings and errors
88 #export GCC_COLORS='error=01;31:warning=01;35:note=01;36:caret=01;32:locus=01:quote=01'
89
90 # some more ls aliases
91 alias ll='ls -aLF'
92 alias la='ls -A'
93 alias l='ls -CF'
94
95 alias vi='vim'
96 # Add an "alert" alias for long running commands. Use like so:
97 # sleep 10; alert
98 alias alert='notify-send --urgency=low -i "${[ $? = 0 ] && echo terminal || echo error}" "$(his
    tory|tail -n1|sed -e '\''s/^\s*[0-9]\+\s*//;s/[;&|]\s*alert$//'\`)"'
99
100 # Alias definitions.
101 # You may want to put all your additions into a separate file like
102 # ~/.bash_aliases, instead of adding them here directly.
103 # See /usr/share/doc/bash-doc/examples in the bash-doc package.
104
105 if [ -f ~/.bash_aliases ]; then
```

96,1

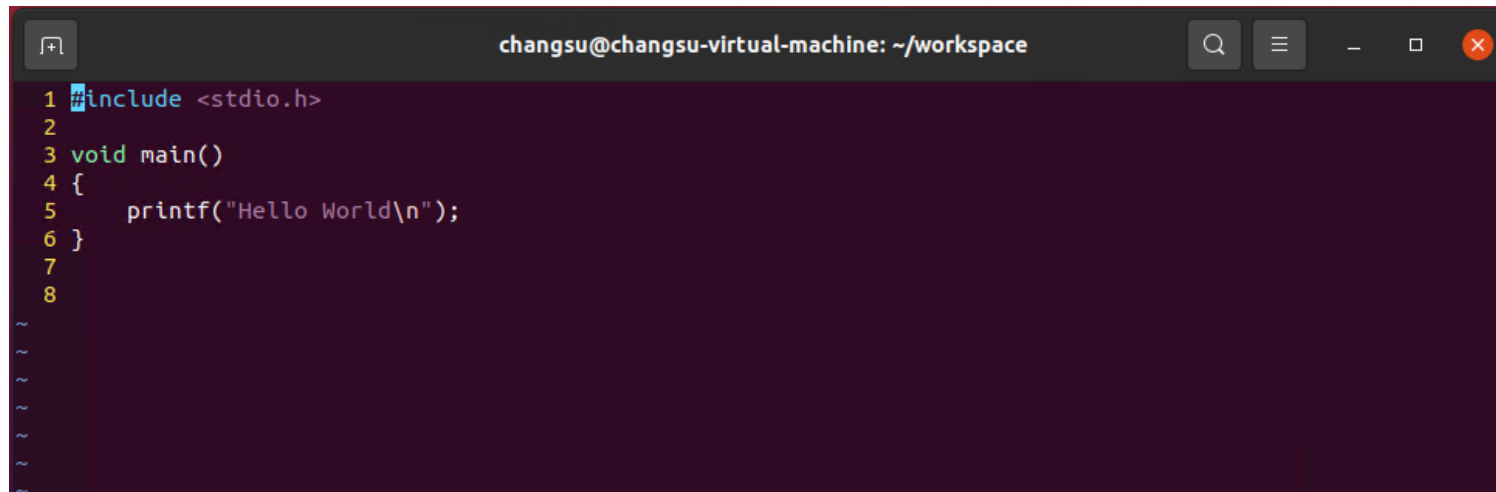
70%

수정된 .bashrc 파일 적용하기

- 파일 수정 이후 수정된 파일을 시스템에 적용
 - \$ source ~/.bashrc
 - source: 수정된 파일 적용
- 변경 내용
 - C 문법에 따라 다른 색상 표시
 - 라인 번호 표시
 - 자동 들여쓰기 등

바로 적용이 안되네

source ~/.bashrc
를 통해 적용

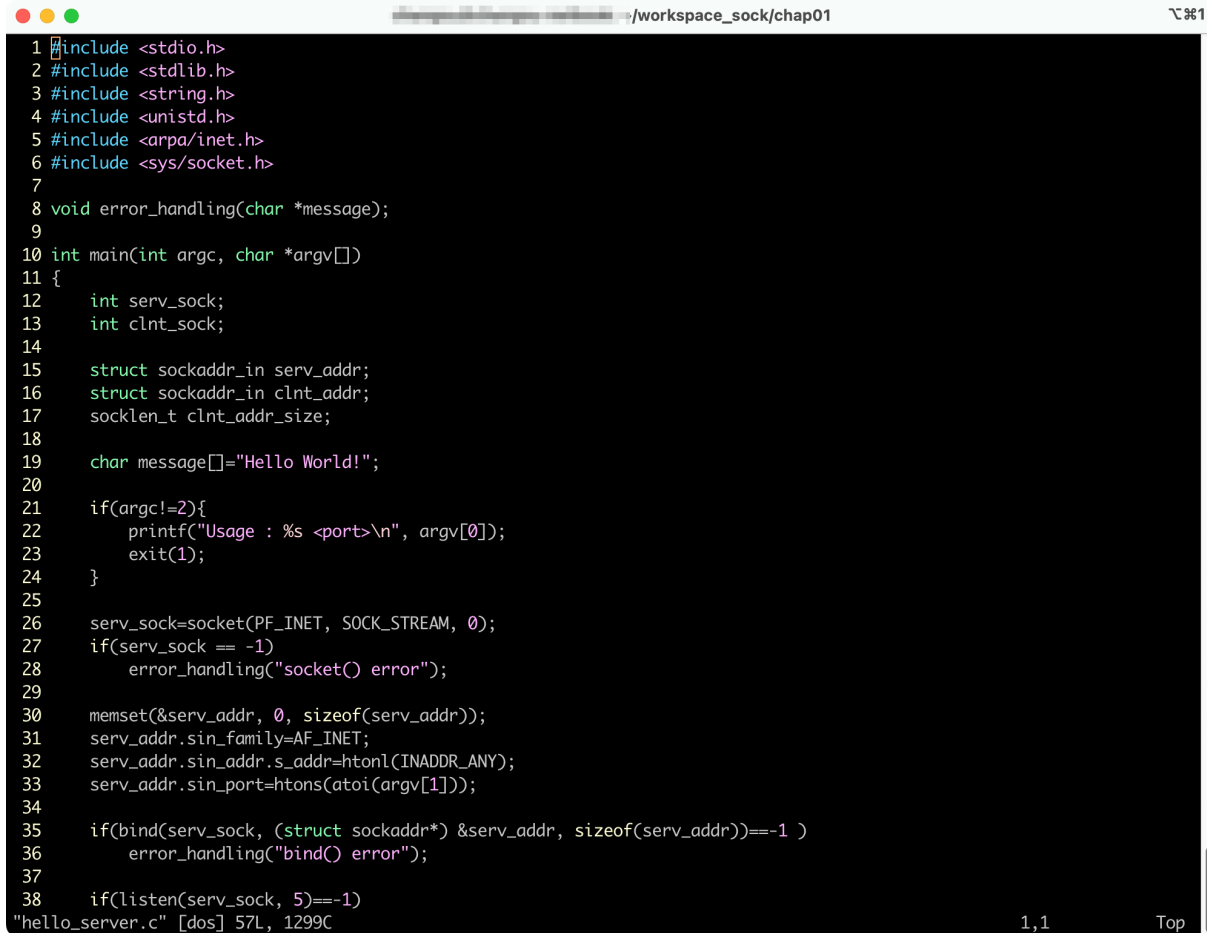


```
changsu@changsu-virtual-machine: ~/workspace
1 #include <stdio.h>
2
3 void main()
4 {
5     printf("Hello World\n");
6 }
7
8
~
~
~
~
~
```

vi 에디터 실행

■ vi 에디터 실행

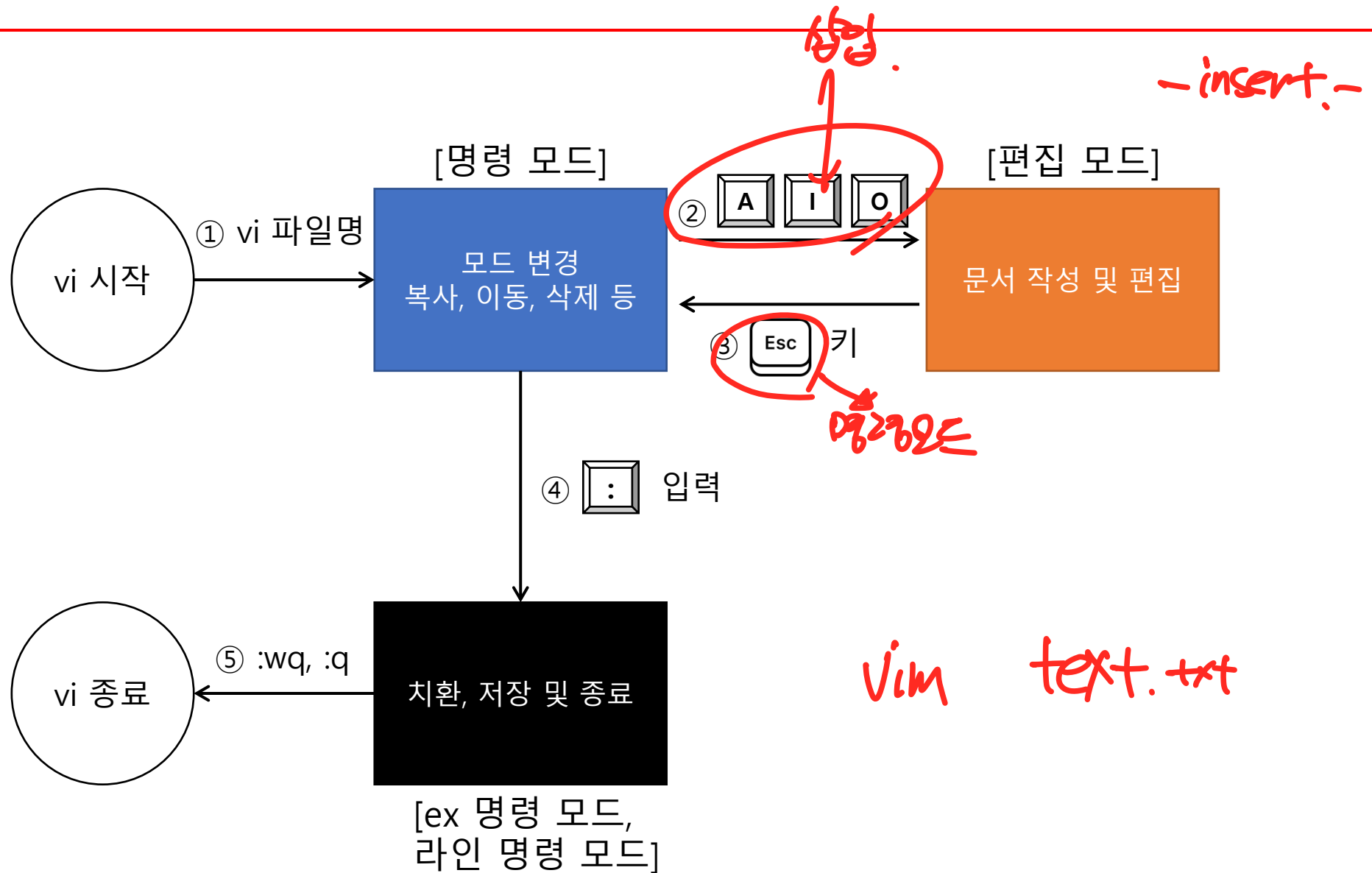
\$ vi 파일이름



```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <string.h>
4 #include <unistd.h>
5 #include <arpa/inet.h>
6 #include <sys/socket.h>
7
8 void error_handling(char *message);
9
10 int main(int argc, char *argv[])
11 {
12     int serv_sock;
13     int clnt_sock;
14
15     struct sockaddr_in serv_addr;
16     struct sockaddr_in clnt_addr;
17     socklen_t clnt_addr_size;
18
19     char message[]="Hello World!";
20
21     if(argc!=2){
22         printf("Usage : %s <port>\n", argv[0]);
23         exit(1);
24     }
25
26     serv_sock=socket(PF_INET, SOCK_STREAM, 0);
27     if(serv_sock == -1)
28         error_handling("socket() error");
29
30     memset(&serv_addr, 0, sizeof(serv_addr));
31     serv_addr.sin_family=AF_INET;
32     serv_addr.sin_addr.s_addr=htonl(INADDR_ANY);
33     serv_addr.sin_port=htons(atoi(argv[1]));
34
35     if(bind(serv_sock, (struct sockaddr*) &serv_addr, sizeof(serv_addr))==-1 )
36         error_handling("bind() error");
37
38     if(listen(serv_sock, 5)==-1)
```

"hello_server.c" [dos] 57L, 1299C 1,1 Top

vi 기본 동작



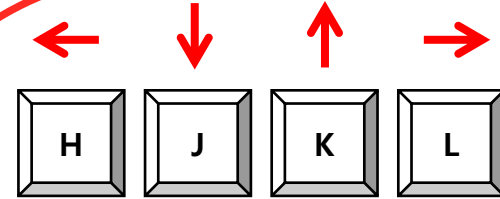
vi 명령 모드

'a' : 현재 커서 뒤에 추가
'o' : 다음 줄

■ 명령 모드: ESC

• 커서 이동

:숫자 + Enter: 해당 라인으로 이동



• 삭제, 복사, 붙여 넣기

한번
재무력

명령어	의미
x	현재 위치에서 한 글자 삭제 (del)
dw	단어 단위 삭제
dd	현재 커서 위치에서 행 삭제 (1줄 삭제)
숫자dd	커서 위치에서 숫자개의 라인 삭제
yy	현재 커서 위치에서 행 복사 (1줄 복사)
숫자yy	현재 커서 위치에서 숫자만큼의 행을 복사 (예: 3yy)
p	복사한 내용을 현재 행 이후에 붙여 넣기
P	복사한 내용을 현재 행 이전에 붙여 넣기
:5,7d	5~7번째 행 삭제
u	직전 명령 취소 (undo)

vi 입력 모드

- 입력 모드 (편집 모드)
 - i: 현재 커서의 위치부터 입력
 - a: 현재 커서의 위치 다음부터 입력
 - o: 현재 커서의 다음 줄에 입력
- ex 명령 모드 (라인 명령 모드)

명령어	의미
:w	변경 사항 저장
:w 파일명	변경 사항을 입력한 파일명에 저장
:wq	저장 후 종료
:q!	강제 종료 (변경 사항 저장하지 않음)
/키워드	해당 문서에서 키워드를 검색
n	검색된 키워드가 있는 경우 다음으로 이동
N	검색된 키워드가 있는 경우 이전으로 이동
:%s/str/rep/g	파일 전체에서 str을 찾아서 rep로 전부 교체

vi 단축키

version 1.1
April 1st, 06

Esc

명령 모드

vi / vim

단축키 모음

~ 대문자 전환	! 외부 명령	@ 매크로 실행	# 이전 검색	\$ 줄 끝으로 이동	% 일치하는 줄로 찾기	^ 줄의 첫 글자	& :s 반복	* 다음 검색	(문장 시작) 문장 끝	아래줄로 이동	+ 다음 줄
· 마크로 이동	1	2	3	4	5	6	7	8	9	0 줄의 처음	- 이전 줄	= 자동 들여쓰기
Q 실행 모드	W 다음 WORD	E 끝 WORD	R 수정 모드	T 뒤로 검색	Y 줄단위 복사	U 줄 단위 실행취소	I 줄 시작에서 삽입	O 행 위에 삽입	P 커서 이전에 붙여넣기	{ 문단 시작	}	문단 끝
q 매크로 기록	w 다음 단어	e 단어 끝	r 한 문자 교체	t 한 문자 검색	y 복사	u 실행취소	i 편집 모드	o 행 아래에 삽입	p 커서 이후에 붙여넣기	[기타]	기타
A 줄 끝에 덧붙이기	S 삭제 후 편집 모드	D 줄 끝까지 삭제	F 뒤로 검색	G 파일 끝으로 이동	H 화면 상단	J 줄 합치기	K 다음 말	L 화면 하단	: 명령줄	ex 레지스터 지정	" 열 이동	.
a 덧붙이기	s 단어 삭제 후 편집 모드	d 삭제	f 한 문자 찾기	g 확장 명령	h ←	j ↓	k ↑	l →	; 명령 반복	/T/t/f/F 마크로 이동	\ 사용 안함	.
Z 종료	X 백스페이스 바꾸기	C 줄 끝까지 바꾸기	V 줄단위 비주얼 모드	B 이전 WORD	N 이전 (찾기)	M 화면 가운데	< 내어쓰기	> 들여쓰기	? 찾기 (뒤로)	.	.	.
Z 확장명	x 글자 삭제	c 바꾸기	v 비주얼 모드	b 이전 단어	n 다음 (찾기)	m 마크 설정	, 역순 검색	. 명령 반복	/ 찾기	.	.	.

동작

명령

연산자

확장

커서를 이동하거나, 연산자가 동작할 범위를 지정합니다.

바로 동작하는 명령, 빨간색은 편집 모드로 변경됩니다.

이동 관련 문자(숫자나 커서 이동)와 함께 사용하여야 하며, 커서의 위치부터 목적지까지 연산합니다.

특별한 키 합수로, 추가적인 키 입력이 필요합니다.

입력 후 (숫자를 제외한 .으로 끝낼 수 있는) 글자를 입력하여야 합니다.

words: 구분자로 공백, 특수기호 모두 사용
WORDS: 구분자로 공백 문자만 사용

words: `quux(foo, baz, baz)`
WORDS: `quux(foo, baz, baz)`

주요 명령행 명령 ('ex'):

:w (저장), :q (종료), :q! (저장하지 않고 종료)
:e f (파일 f 열기),
:%s/x/y/g (파일 전체에서 'x'를 'y'로 교체),
:h (vim 도움말), :new (새 파일)

그외 중요한 명령들:

CTRL-R: 재실행 (vim),
CTRL-F/-B: 페이지 위로/아래로,
CTRL-E/-Y: 줄 스크롤 위로/아래로,
CTRL-V: 블록-비주얼 모드 (vim 전용)

비주얼 모드:

커서를 움직여 지정된 범위에 연산자를 적용합니다. (vim 전용)

참고:

(1) 복사/붙여넣기/지우기 명령어를 사용하기 전에 "x"를 입력하여 레지스터(클립보드)를 지정하세요. (x는 a에서 z 또는 *을 사용할 수 있음) (예: "ay\$를 입력하면 현재 커서에서 라인 끝까지의 내용을 레지스터 'a'에 저장합니다.)

(2) 어떤 명령을 입력하기 전에 횡수를 지정하면, 횡수만큼 반복하게 됩니다. (예: 2p, d2w, 5i, d4j)

(3) 연속으로 입력하는 명령은 현재의 라인에 반영됩니다. 예시: dd(현재 라인 지우기), >>(들여쓰기)

(4) ZZ는 저장후 종료, ZQ는 저장하지 않고 종료.

(5) zt: 커서가 위치한 곳을 제일위로 올리기, zb: 바닥으로, zz: 가운데로

(6) gg: 파일의 처음으로(Vim 전용), gf: 커서가 위치한 곳의 파일 열기(Vim 전용)