

▶ 리눅스와 텍스트 파일

- 리눅스 시스템에서 중요한 설정 정보나 셸 스크립트가 텍스트 파일로 존재함
- 텍스트 파일의 장점
 - 공유가 쉽고 텍스트 편집기를 이용하여 간단하게 편집할 수 있음

▶ 리눅스 텍스트 편집기의 종류

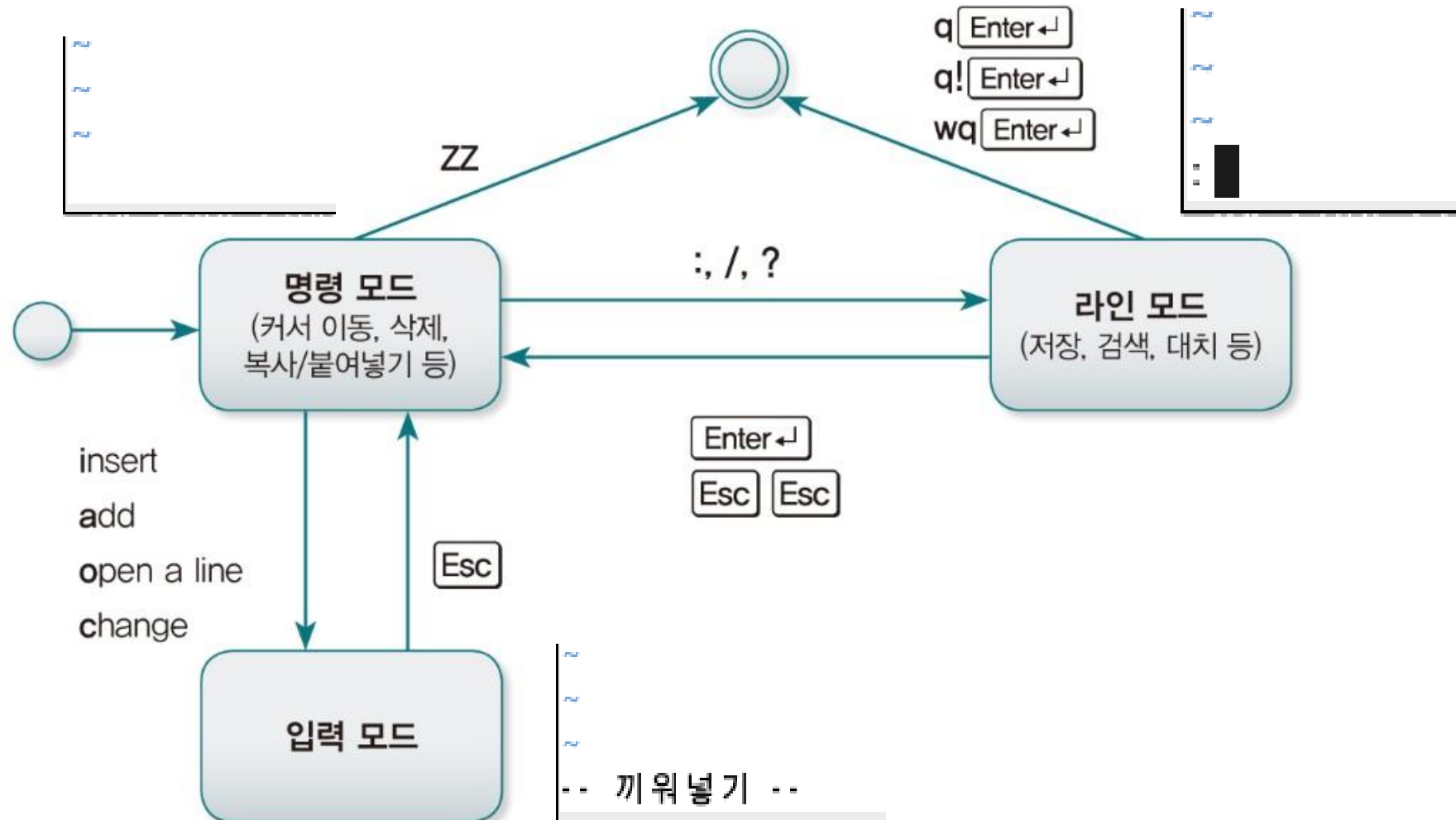
- gedit
 - GNOME 데스크톱에서 제공하는 작고 가벼운 편집기
- emacs
 - 화면 단위 편집기로 다양한 기능을 제공하나 사용법이 어려움
- vi
 - vi는 유닉스 계열 운영체제에서 가장 보편적인 화면 편집기
 - vi는 'vim'으로 에일리어스 되어 있음
 - vim(Vi Improved)이 vi와 다른 점
 - 파일의 구조를 표시하기 위해 컬러를 사용
 - 마우스 지원, 다중 undo, 다중 탭(또는 화면 분할)을 지원
 - 블록을 선택할 때 비주얼 모드를 지원

▶ vi 시작하기

- 터미널 창의 셸 프롬프트에서 다음을 수행함
 - 모달 편집기로 다른 작업은 수행할 수 없음
- *vi [options] [filename]*



▶ vi 편집기와 세 가지 모드



▶ vi 종료하기

➤ 명령 모드에서 키보드 명령을 입력

- ZZ(Shift+zz) : 변경 내용을 저장하고 종료

➤ 라인 모드에서 종료하는 명령

- :q
 - 종료(변경된 내용이 없는 경우)
- :q!
 - 저장하지 않고 강제 종료
- :wq [*filename*]
 - 저장하고 종료
 - :w [*filename*] 또는 :w! [*filename*] 은 저장만 하는 명령
- :x [*filename*]
 - 변경이 있었다면 저장하고 종료
- :x! [*filename*]
 - 위와 같으나 같은 이름의 파일이 있으면 덮어쓰

학습목차

1 편집기

2 vi로 편집하기

3 파일 찾기와 문자열 검색









▶ 텍스트 입력

- 초기 명령 모드에서 **입력 모드로 이동 후 입력**이 가능함
- 입력 모드일 때는 화면의 하단에 '-- 끼워 넣기 --'가 표시됨
- 입력 모드에서 **[Esc]**를 누르면 명령 모드로 돌아감

명령 키	설명
a	현재 커서 위치의 오른쪽부터 텍스트를 추가(add)할 수 있음
A	현재 라인의 끝에 텍스트를 추가할 수 있음
i	현재 커서의 위치부터 텍스트를 삽입(insert)할 수 있음
I	현재 라인의 처음에 텍스트를 삽입할 수 있음
o	현재 라인의 아래에 라인이 열리고(open) 텍스트를 추가할 수 있음
O	현재 라인의 위에 라인이 열리고 텍스트를 추가할 수 있음

▶ 커서 이동 명령

- 명령 모드에서 화살표 키와 ,  키를 사용할 수 있음
- 그 외에도 다양한 키가 존재함

명령 키	설명
h j k l	   
+f +b	 
w	다음 단어의 첫 자로 이동
b	앞 단어의 첫 자로 이동
0	현재 라인의 첫 칸으로 이동
\$	현재 라인의 마지막 칸으로 이동
20G	20번째 라인으로 이동
G	마지막 라인으로 이동
5j	현재 라인에서 아래로 5라인을 이동

▶ 텍스트 삭제 명령

➤ 명령 모드에서 텍스트를 삭제하거나 변경함

- 적당한 위치로 커서를 옮긴 후 삭제함

명령 키	설명
x	커서 위치의 한 문자를 삭제
dw	커서의 위치부터 다음 단어 전까지 삭제
d\$ 또는 D	커서의 위치부터 현재 라인의 끝까지 삭제
d0	커서의 왼쪽부터 현재 라인의 처음까지 삭제
dd	현재 라인을 삭제
dG	현재 라인부터 파일의 끝까지 삭제
d20G	현재 라인부터 라인 번호 20까지 삭제
3dw	커서의 위치부터 세 단어를 삭제
5dd	현재 라인부터 다섯 라인을 삭제

▶ 텍스트 변경(또는 치환) 명령

- 대개 일정 부분의 텍스트를 삭제하고 입력 모드로 전환됨
- 삭제, 변경 또는 붙여 넣기를 한 후 커서를 이동하고 **도트(.) 명령**을 수행하면 앞서 수행한 작업을 반복할 수 있음

명령 키	설명
r	커서 위치의 한 문자를 대치
R	커서의 위치부터 계속 대치할 수 있음. 입력 모드로 전환됨(수정)
s 또는 ci	커서 위치의 문자가 삭제되고 입력 모드로 전환됨(삽입)
S 또는 cc	현재 라인에 존재하는 문자가 삭제되고 현재 라인에서 입력 모드로 전환됨
cw	커서 위치부터 단어의 끝까지 삭제되고 입력 모드로 전환됨
cb	커서 왼쪽부터 단어의 앞까지 삭제되고 입력 모드로 전환됨
c)	커서 위치부터 문장의 끝까지 삭제되고 입력 모드로 전환됨
c}	커서 위치부터 문단의 끝까지 삭제되고 입력 모드로 전환됨

▶ 텍스트 복사

- 삭제나 변경을 위한 명령 키를 사용할 때 텍스트가 복사됨
- 또는 커서를 적당한 위치로 이동한 후 **y** 명령을 사용하여 복사함

▶ 텍스트 붙여 넣기

- **P** 또는 **p** 명령을 이용하여
(커서의 왼편/위 라인) 또는 (오른 편/아래 라인)에 붙여넣기를 할 수 있음

명령 키	설명
yy	현재 라인을 복사
5yy	현재 라인부터 다섯 라인을 복사
yw	커서의 위치부터 다음 단어의 앞까지 복사
y\$	커서의 위치부터 라인의 끝까지 복사
y0	커서 왼쪽부터 라인의 앞까지 복사
y^	위와 같으나 라인 앞의 빈칸은 복사하지 않음
yG	현재 라인부터 파일의 끝까지 복사
y)	커서의 위치부터 한 문장을 복사
y}	커서의 위치부터 한 문단을 복사

▶ 블록 지정하기

➤ v 또는 V 명령을 사용하여 문자 또는 라인 단위로 지정함

- 블록을 지정하고 싶은 곳에 커서를 이동하고 화살표를 사용하여 블록을 지정하고 삭제(x) 또는 복사(y)함
- 커서를 이동한 후 p 또는 P 명령을 사용하여 붙여 넣기를 할 수 있음

명령 키	설명
v	비주얼 모드에서 한 번에 한 문자씩 선택
V	비주얼 모드에서 한 번에 한 라인씩 선택

▶ 명령의 취소와 기타 명령

➤ 명령 모드에서 다음과 같은 명령을 수행할 수 있음

명령 키	설명
u	이전 변경 작업을 취소(undo)
Ctrl+r	앞서 취소했던 작업을 다시 하는 것으로 취소를 취소
:e!	마지막으로 저장한 후의 작업을 취소하고 다시 작업
:n	n번째 라인으로 이동
J 또는 :j	현재 라인과 아래 라인을 합침
!:command	vi를 사용하다 쉘 명령 command를 수행
:sh	vi를 잠시 빠져나가 쉘 명령을 수행함. 쉘 명령으로 exit을 실행하면 다시 vi로 돌아옴
:se nu	화면 좌측에 라인 번호를 표시
Ctrl+g	화면 하단에 파일의 이름, 전체 라인 수, 현재 라인의 상대적 위치를 퍼센티지로 표시
:r filename	지정된 파일의 내용을 읽어 현재 커서의 아래 줄에 추가

▶ 검색 명령(1)

- 한 라인에서 검색할 때 명령 모드에서 f를 사용
 - 예를 들어 fa 명령은 커서 다음 위치부터 a를 찾음
 - 세미콜론(;)을 사용하여 검색을 반복
- 파일 전체를 대상으로 검색할 때 다음 명령을 사용

명령 키	설명
<i>/pattern</i>	순방향으로 pattern을 검색하여 커서를 이동
<i>/pattern/+ num</i>	pattern을 검색하고 그 뒤(+) num번째 라인으로 이동
<i>?pattern</i>	역방향으로 pattern을 검색하여 커서를 이동
<i>?pattern?- num</i>	pattern을 검색하고 그 앞(-) num번째 라인으로 이동
n	이전 검색을 같은 방향으로 반복
N	이전 검색을 반대 방향으로 반복

▶ 검색 명령(2)

➤ 문자열을 검색할 때 정규식을 사용할 수 있음

명령 키	설명
/hello	순방향으로 단어 hello를 검색하여 커서를 이동
?goodbye	역방향으로 단어 goodbye를 검색하여 커서를 이동
/The.*does	단어 The를 포함하고 그 이후에 단어 does를 포함하는 라인을 검색함. 점(.)은 임의의 한 문자와 매칭되며, 애스터리스크(*)는 앞의 문자(여기서는 임의의 한 문자)가 0개 이상 나타날 수 있음을 의미
?[pP]rint	단어 print나 Print를 역방향으로 검색함 []는 안에 포함하는 문자 중 하나와 매칭됨

▶ 검색과 대치

➤ 문자열을 검색하여 대치하는 기능

➤ **`:[g][address]s/pattern/replacement [/option]`**

- `[g][address]`는 검색 범위로, %는 전체이며, 예를 들어 `/pine/`은 커서 다음에 나오는 pine을 포함하는 한 라인, `g/pine/`은 pine을 포함하는 모든 라인을 대상으로 함
- `s`는 `pattern`을 찾아 `replacement`로 대치하라는 연산
- 마지막 옵션이 생략되면 검색된 라인에서 첫 번째 패턴만 대치하며, `/g`가 사용되면 모든 패턴을 대치

명령 키	설명
<code>:s/pattern/replacement</code>	커서가 위치한 라인에서 첫 번째 등장하는 pattern을 replacement로 대치
<code>:%s/pattern/replacement</code>	전체 범위의 라인에서 위와 같은 것을 수행. % 대신에 1,\$를 넣어도 됨
<code>:%s/pattern/replacement/g</code>	위와 같으나 등장하는 모든 pattern을 대치
<code>:1,10s/pattern/replacement/g</code>	위와 같으나 1~10 라인에서 수행됨