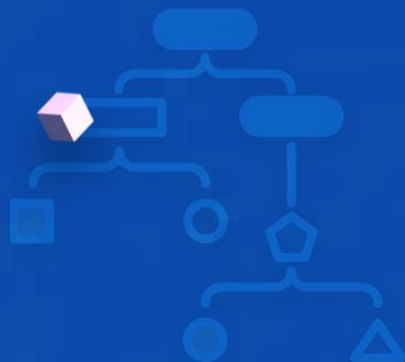


강원혁신플랫폼

리눅스프로그래밍

텍스트 에디터





학습
내용

1 vi 에디터의 명령어

학습
목표

vi 에디터의 명령어를 파악할 수 있다.



vi 에디터의 명령어



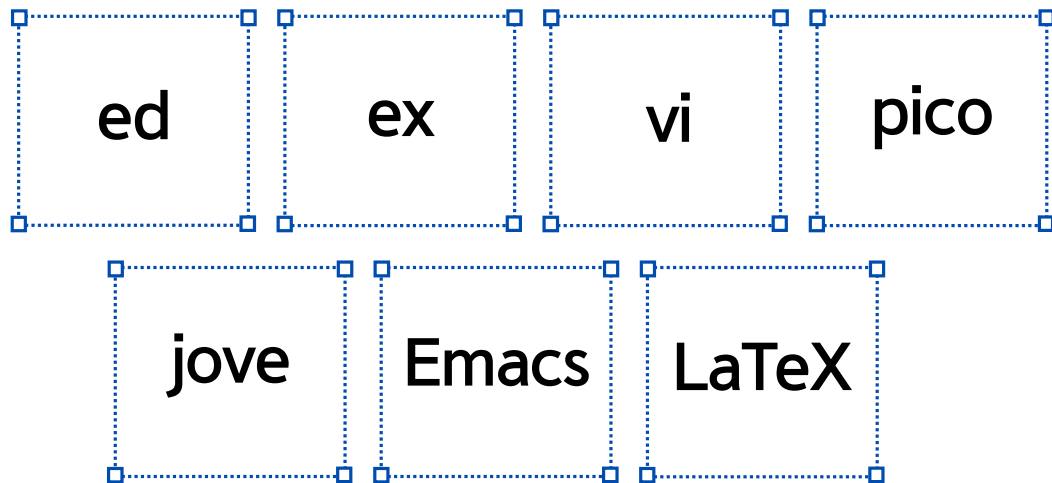


유닉스의 텍스트 에디터

프로그램	내용
ed/ex	적은 메모리를 사용하고 속도는 빠르지만, 텍스트 파일을 줄(라인) 단위로 편집하기 때문에 다른 텍스트 에디터에 비해 불편하다.
vi	대부분의 유닉스에서 지원하고 있으며, 강력한 기능을 제공하지만 사용할 수 있는 기능이 너무 많아서 완벽히 배우기 어렵다.
GNU nano	문자의 색상 표시, 정규표현식을 이용한 검색 등을 지원하는 유닉스를 위한 텍스트 에디터로 vi보다 편리한 편집 기능을 제공한다. 라즈베리 파이도 nano를 지원하며 vi를 대신하여 사용할 수 있다.
Emacs	GNU를 만든 리처드 스톨먼이 만든 텍스트 에디터로 라이선스가 자유롭고 강력한 기능을 제공하지만, 대부분의 유닉스에서 기본으로 제공되지 않고 별도의 설치 과정이 필요하다.
LaTeX	알고리즘과 프로그래밍 기법의 선구자 도널드 커누스(Donald Ervin Knuth)가 만든 TeX 기반의 문서 조판용 프로그램이다. 다양한 매크로와 변환 기능을 제공한다.



유닉스의 텍스트 에디터

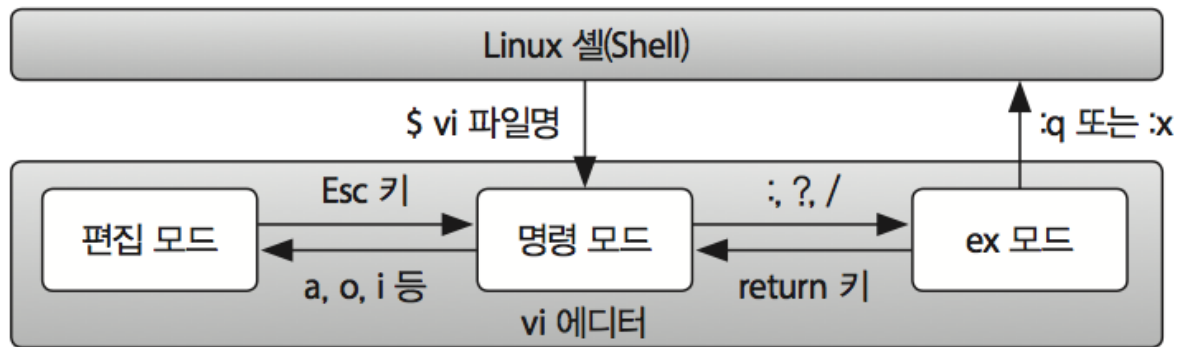


다양한 텍스트 에디터를 제공



vi 에디터의 특징

- ◆ vi 에디터는 UC 버클리의 BSD 유닉스용으로 개발
- ◆ 거의 모든 유닉스에서 기본적으로 제공
- ◆ 리눅스에서는 기존의 vi를 보다 편하게 사용하도록 성능을 개선한 vim(vi improved)을 사용





vi 에디터의 특징

모드	내용
명령모드	기본 명령 실행 모드
편집모드	문자 입력과 수정 가능 모드
ex 모드	명령어를 입력할 수 있는 모드

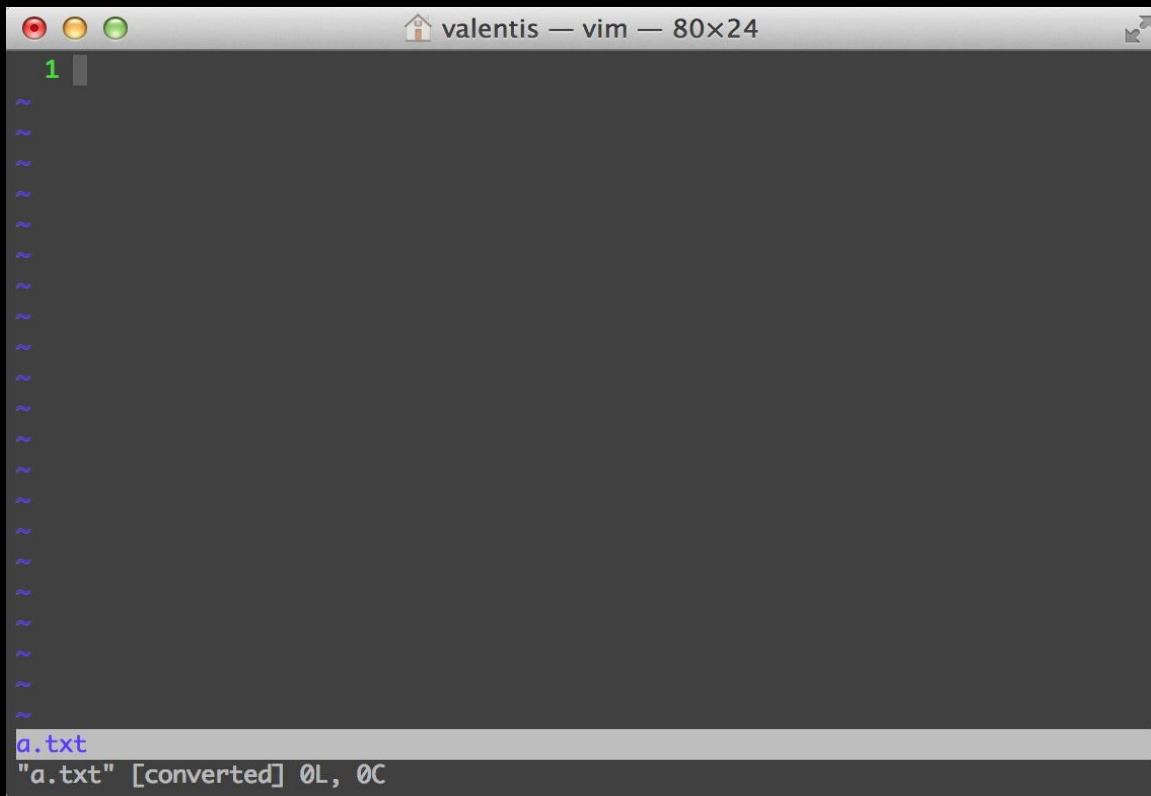


vi 에디터의 입력/편집 모드 키

명령	내용
i	현재 커서의 앞에서부터 문자를 입력한다.
I	현재 커서가 위치하고 있는 줄의 제일 앞부분에서부터 문자를 입력한다.
a	현재 커서의 뒤에서부터 문자가 입력된다.
A	현재 커서가 위치하고 있는 줄의 제일 뒷부분에서부터 문자를 입력한다.
o	현재 커서의 위치에서 다음 줄을 삽입하고 문자를 입력한다.
O	현재 커서의 위치에서 앞 줄을 삽입하고 문자를 입력한다.
R	현재 커서가 있는 위치부터 기존의 문자를 새로운 문자로 덮어쓴다.
r	현재 커서가 있는 위치의 한 문자를 새로운 문자로 덮어쓴다.



📦 복사와 붙여넣기(yy, p)



📦 복사와 붙여넣기(yy, p)

```

1 Hello World
2 Hello World
3 Hello World
4 Hello World
5 Hello World

```

a.txt

"a.txt" [converted] 0L, 0C



복사와 붙여넣기

명령	내용
yy	현재 라인을 메모리로 복사
x	현재 문자를 메모리로 복사하고 지우기(자르기)
p	메모리에 복사된 내용을 현재 위치에 붙여넣는다.



📦 커서의 이동

키	내용
'↑'(위 방향키)나 k 또는 -	현재의 커서를 한 줄 위로 이동한다.
'↓'(아래 방향키)나 j 또는 +	현재의 커서를 한 줄 아래로 이동한다.
'→'(오른쪽 방향키) 또는 l	현재의 커서를 한 문자 오른쪽으로 이동한다.
'←'(왼쪽 방향키) 또는 h	현재의 커서를 한 문자 왼쪽으로 이동한다.
^ 또는 0	커서를 현재 줄의 시작 위치로 이동한다.
\$	커서를 현재 줄의 마지막 위치로 이동한다.
b	현재의 커서를 한 단어 앞으로 이동한다.
w	현재의 커서를 한 단어 뒤로 이동한다.



📦 커서의 이동

키	내용
Control-D	반 화면 아래로 이동한다.
Control-F	반 화면 위로 이동한다.
Control-U	한 화면 아래로 이동한다.
Control-B	한 화면 위로 이동한다.
:nn↓ 또는 nnG↓	커서를 nn번째의 줄로 이동한다.



❏ 삭제, 범위, 치환, 검색

명령	내용
[숫자]x	현재의 커서의 문자를 삭제하거나 변경한다.
[숫자]dw	현재의 커서에서 단위를 삭제하거나 변경한다.
[숫자]dd	현재의 커서가 위치하고 있는 줄을 삭제하거나 변경한다.
D	현재 커서의 위치에서 마지막까지 삭제한다.
:<범위>d	특정 범위의 줄을 삭제한다.



범위 지정 및 치환

명령	내용
숫자	해당 숫자의 줄 번호를 의미한다.
.	현재 커서가 있는 줄 번호를 의미한다.
\$	파일의 마지막 줄 번호를 의미한다.
:<범위>s/원본/변경/g	문서 내에서 특정 범위의 원본 문자열을 변경



검색

명령	내용
/문자열	현재의 위치에서 문서의 끝 방향까지 검색한다.
?문자열	현재의 위치에서 문서의 앞 방향까지 검색한다.
n 이나 /	이전에 검색했던 문자열로 문서의 끝 방향까지 다음의 내용을 찾는다.
N 이나 ?	이전에 검색했던 문자열로 문서의 앞 방향까지 다음의 내용을 찾는다.



설정

명령	내용
:se nu	문서에서 줄 번호를 표시
:se ai	자동 들여쓰기 기능을 설정
:se cindent	C 언어 스타일의 자동 들여쓰기 기능을 설정
:se all	모든 vi 변수와 설정 값을 표시



vi 에디터의 명령어(6)

vi의 저장과 종료

강제 저장 및 종료에는 명령 뒤에 !를 사용

명령	내용
:w [파일명]	vi에서 편집한 내용을 파일에 저장한다.
:e 파일명	새로운 파일을 엽니다.
:r 파일명	현재의 내용 뒤에 파일의 내용을 덧붙인다.
:q	vi 에디터를 종료한다.
:x	내용을 파일에 저장하고 vi 에디터를 종료한다.



- 비교적 가볍고 사용하기 쉬운 텍스트 에디터
- vi 에디터에 비해 직관적이며 사용하기 쉬움



VI 에디터의 주요 모드가 아닌 것은?

- 1 편집모드 2 명령모드 3 ex모드 4 set모드

4

모드	내용
명령모드	기본 명령 실행 모드
편집모드	문자 입력과 수정 가능 모드
ex 모드	명령어를 입력할 수 있는 모드



01 • vi 에디터의 명령어