

용어 정리

1. ftp, xinetd

- 1) ftp : The File Transfer Protocol 파일전송 프로토콜의 약자로 인터넷을 통해 한 컴퓨터에서 다른 컴퓨터로 파일을 전송할 수 있도록 하는 방법과 그 관련 프로그램 모두를 일컫는다.
(ftp port : 21번)
- 2) xinetd : extended Internet daemon. 모든 서비스 서버에 데몬은 물론 유닉스 계열 시스템에서 돌아가며 인터넷에 관한 연결을 관리. 오래된 Inetd에 비해 강력한 보안을 제공하며 다양한 소켓의 리스닝 바인딩이 많이 사용된다.
※ daemon : 사람이 직접 관리하지 않고, 백그라운드에서 돌면서 여러 작업을 하는 프로그램

2. CR, LF

- 1) CR (Carriage Return) : 한 칸 앞으로 ↓ } window의 엔터 (↵) 기능
- 2) LF (Line Feed) : 처음으로 돌아가기 ←

LINUX와 UNIX에서는 Line Feed만 쓰인다. window에서 만든 텍스트 파일을 Linux에서 열면 개행의 테더가 2개임

ex) Linux window

1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10

WC : 6 WC : 8

3. UTF-8, 한글 2byte

- 1) UTF-8 : Universal Coded Character Set + Transformation Format - 8 bit 유니코드를 위한 가변 길이 인코딩 방법 중 하나로, 변 통하는 8 비트짜리가 만들어진다.
- 2) 한글 2byte : 영문의 경우 ASCII 코드값만 만들면서 알파벳당 1byte (8bit)를 할당해 줬다. 8bit는 255까지 표현이 가능하지만, 알파벳만 표현하면 되려면 영문만 할당 할당을 포함한 다른 언어들은 8bit로는 불가능하다. 2byte짜리 (16bit = 65,536만큼 표현 가능) 용량이 만들어진다. 한글은 2byte ASCII 코드를 쓰는 영문은 1byte, UNICODE를 쓰는 한글은 2byte이다.

정리하기

1) vi 편집기 기본 명령어

실행 vi filename (여러개 가능)
vi *.txt vi 편집모드 실행하기

④ 실행 후 명령어

yy 복사하기 → yN줄짜리 : 숫자만큼 복사가 여러번됨

p paste 붙여넣기

dd 해당 line 전체 삭제

D 커서가 있는 곳 뒤부터 해당 라인 끝까지 삭제

\$ 커서 line의 맨 앞글자부터 이동

* 이 외에도 h, l, k, j 는 ←, →, ↑, ↓ 는 위, 아래로 커서를 이동하는 명령어이다. 방향을 지정 page up, down key의 사용으로 자주 쓰이지 않는다.

:wq 저장 후 나가기

:w 저장

:q! 저장하지 않고 나가기

a 커서위치 앞에서 입력모드로 전환

i 커서위치 뒤에서부터 입력모드로 전환

:e 입력모드로 전환

x(X) 커서 위치의 글자 삭제 (커서 위치 글자 삭제)

r 커서 위치의 문자 삭제 후 입력모드로 전환 (변경시 사용)

o(O) 현재 라인에서 아래로 빈줄을 하나 남고 입력 (위에 줄 삽입 후 실행)

:q (저장하는 상태에서는) 종료

A 커서 위치 라인의 끝에서 입력시작

:x 문서 저장 후 종료

② 자주 쓰이는 명령어.

yy @ p : copy & paste (yN @ p)

a, i : 커서 앞쪽, 뒤쪽 insert 명령어.

:qw : 저장 후 종료 (= :x)

:q! : 저장하지 않고 종료

o : 위에 한 줄 공백 후 insert

2-1) 마지막행 명령모드 (: 명령어)에서 실행

/word : 찾기 시작한 부분부터 아래로 내려가며 검색 → 저장함 : n" (검색해서 ↓)

?word : 찾기 시작한 부분부터 위로 올라가며 검색 → 개발함 : N" (검색해서 ↑)

2-2) 문단별 변경은 마지막행 명령모드에서 다음과 같은 명령어로 실행 가능

:s/찾으려는 단어/바꿀려는 단어/ → 커서가 있는 줄부터 첫번째로 찾는 단어 변경

:s/찾으려는 단어/바꿀려는 단어/g → 커서가 있는 줄의 모든 단어 변경

:N1,N2 s/찾으려는 단어/바꿀려는 단어/ → N1~N2 줄 사이 첫번째 변경

:N1,N2 s/찾으려는 단어/바꿀려는 단어/g → N1~N2 줄 사이 해당하는 모든 단어 변경

:N.\$ s/찾으려는 단어/바꿀려는 단어/g → N번째 줄부터 끝까지 모든 단어 변경

* 첫 번째 단어를 삭제하고 싶을 시, 바꿀려는 단어를 공백으로 두어 "/s/찾/단/g"의 글로 명령어를 입력

2-3) vi 편집기 사용 중 여러 파일을 한 번에 편집할 수 있는 파일 남김

:n → 다음 파일로 이동

:bn → 다음 파일로 이동

:e# → 직접 편집할 파일로 이동 (행의 뒤까지 같은 기능)

:e filename → 편집파일 추가

:w filename → 파일명 변경 * :N1,N2 w filename → N1~N2 줄의 해당 파일명으로 저장

:w %>book → 현재 파일 백업 (파일명에 ".book" 붙음)

! 명령어 → 잠깐 shell 모드로 명령어를 수행하는 vi 모드로 돌림

3. notepad++ 설치·사용 방법

1. google.com에 notepad++를 검색하여 PC에 맞는 버전을 다운로드 후 설치
2. 메뉴 → 플러그인 → NppFTP 설치버전
- 2-1. 설치된 메뉴 → 플러그인 관리 → NppFTP 찾아 설치
3. 메뉴 → 플러그인 → NppFTP → Show NppFTP window
4. NppFTP window에서 (setting) → profile setting → Add new
 - ① Host IP 입력
 - ② Connection type: SFTP
 - ③ Port: 22
 - ④ Username & Password 입력 (root면 User-name root로!)
 - ⑤ Timeout: 0
5. ④ 모양 클릭하여 연결

* root는 비로 연결되지 않을 수 있으니 리눅스 서버에서 따로 설치

sudo vi /etc/ssh/ssh-config

↓

* Permit Root Login 찾기

↓

Permit Root Login yes로 변경

↓

저장 (:wq)

↓

재부팅 (shutdown -r now)

● 2. 생각해볼 문제 및 용어

● 학습전 생각해볼 문제

- 개행 문자인 CR(Carriage Return)과 LF(Line Feed)에 대하여 검색 등을 통하여 알아봅니다.
- 윈도우의 TEXT파일에서의 개행 방법과 유닉스 리눅스에서 개행 방법에 대하여 알아봅니다.

● 용어 (강의 정리 시 필기 할 것)

- ftp, xinetd
- CR, LF
- UTF-8, 한글2byte

CR (Carriage Return) 한칸 밑으로 ↓
LF (Line Feed) 처음으로 돌아가기 ←
→ Line Feed만 씀 LINUX . UNIX
예)

0	0	1	2
0	0	3	4
0	0	5	6
0	0	7	8

 - window 8
- LINUX 6.

● 3. 이해하기

1) 기본 명령어

vi명령어는 편집 작업과 명령어실행을 일종의 단축키와 같은 방식으로 실행하기 때문에 많이 사용되는 명령어는 외워서 사용 하여야 한다.

① vi실행

· vi는 리눅스 셸(터미널)환경에서 사용되는 파일의 조회, 수정, 작성을 위한 프로그램, 윈도우에서 메모장과 같은 기능

· vi filename : filename이라는 파일을 편집함

· vi filename1 filename2 ...filenameN : 파일 filename1,...,filenameN을 전부 열어서 편집함. 처음 보이는 파일은 filename1이며 편집기 안에서 필요한 파일로 전환하여 편집수행
파일은 이동하며 편집가능.

· vi *.txt : 확장파일명으로 vi 실행시 관련된 모든 파일이 편집됨

② vi명령어 모드

· vi프로그램 안에서는 “명령모드”, “입력모드”, “마지막 행 명령모드”의 세 가지 모드 상태가 있으며 각각의 상황에 맞도록 사용함

3. 이해하기

모드	상태	주요 명령어 수행방법
명령모드	간단한 키 명령을 입력하는 상태	·방향이동 : 화살표키 ·페이지 이동 : ctrl-u, ctrl-f ·[a] : (append) 현재문자 뒤부터 입력 > 입력모드로 전환 ·[i] : (insert) 현재문자 앞에서부터 입력>입력모드로 전환 ·[o] : 현재 줄 아래 한 줄 삽입>입력 모드로 전환 ·[: : 마지막행 모드로 전환
입력모드	간단한 키 명령을 입력하는 상태	·(ESC)나 (ctf)(): 명령모드로 전환
마지막행 명령모드	화면의 마지막 행에서 명령을 입력할 수 있는 모드 하단에[:] 표시가 나타남	·:q! 내용을 저장 없이 종료 ·:wq 내용을 저장 하고 종료 ·:e 입력모드로 전환

<표 II-3> vi 명령어 모드

```
i am agirl
i am agirl
i am agirl i am agirl
~
~
~
:q!
```

<그림 II-17> 마지막 행 명령모드

3. 이해하기

2) vi 명령어 및 주요 키

앞에서 학습한 vi 의 주요 명령어 모드에서 많이 사용되는 키나 명령어에 대하여 학습해 본다

① 명령 모드에서 사용되는 명령어

- 명령 모드에서 사용되는 주요 명령어는 다음과 같음
- 자주 사용되는 명령어에 대하여 반드시 숙지 필요

Tip 아래 명령어는 글로 설명하는 것보다 직접 현재 명령어를 실습을 통해 익히는 것이 중요함

자주 사용하는 명령		간혹 사용하는 명령	
키	내용	키	내용
h	<- (방향키 왼쪽)	k	방향키 위쪽
l	-> (방향키 오른쪽)	j	방향키 아래쪽
^b	한 페이지 위로 이동	e	한 단어 뒤로 이동
^f	한 페이지 아래로 이동	b	한 단어 앞으로 이동
\$	해당 라인 맨 끝으로 이동	^u	반 페이지 위로 이동
x	현재 커서 문자 삭제(del)	^d	반 페이지 아래로 이동
dd	커서가 위치한 라인 삭제	0	해당 라인 맨 앞으로 이동
dd	커서가 위치한 라인 삭제	X	현재 커서 앞 문자 삭제 (Backspace)
yy	현재 라인을 yank(copy)함.	dw	커서가 위치한 곳부터 단어 끝까지 삭제
p	커피한 라인을 커서가 있는 라인 아래에 붙임 (Paste)	r	커서가 위치한 글자를 바꿈.

3. 이해하기

(2) vi 편집기 활용

vi 편집기에 대한 기본적인 명령어를 익혔다면, 이제는 직접 문서를 작성하거나 편집할 때 유용한 단어의 검색, 문자열 바 꾸기, 기타 유용한 명령에 대하여 배워보도록 한다.

1) 파일 내 단어 검색

파일의 내용이 많은 경우 우리가 원하는 단어를 찾아 그 위치에서 작업하여야 하는 경우가 많다. 원하는 단어를 주고 검색을 하여야 하는데 이때 유용한 명령어를 알아본다.

① 필요한 단어 검색

- 마지막 행 명령모드에서 실행 함
- /word: word라는 단어를 위에서 아래방향으로 찾을 <그림 II-18>
- ?word: word라는 단어를 아래에서 위 방향으로 찾을

```
i am agirl
i am agirl i am agirl i am agirl
~
~
:/girl
```

<그림 II-18> 필요한 단어 검색

② 검색 이후 계속해서 다음 단어 검색

- 명령모드에서 실행함
- n: 한번 검색 후 아래로 동일한 검색어를 다시 찾을 *아래단에 있을지 :/ word로 내용*
- N: 한번 검색 후 위로 동일한 검색어를 다시 찾을

3. 이해하기

2) 문자열 변경

파일의 내용 중 단어를 변경하여야 하는 경우가 많은데, 이때 찾을 문자와 교체할 문자를 지정하고 바꿀 수 있다. 이때 유용한 명령어를 알아본다.

① 하나의 문자열 변경

- 마지막 행 명령모드에서 실행
- :s/찾을 문자열/변경할 문자열/로 명령 *동일한 라인에 문자열, 조건에 다 변경*
- 정확한 표현은 :s/찾을 문자열/변경할 문자열/g로 명령
- 현재 커서가 위치한 행에 대하여 문자열 교체

Tip

:s/girl/boy/로 명령하는 경우 문서에서 제일 처음 나오는 girl이라는 단어가 boy로 변경 됨

```
i am agirl
i am agirl
i am agirl i am agirl
~
~
:s/girl/boy/
```

<그림 II-19> 단어 한 개 변경

② 모든 문자열 변경

- 마지막 행 명령모드에서 실행

3. 이해하기

- :1,\$s\$찾을 문자열/변경할 문자열/g로 명령
- 1,\$ 는 첫줄부터 끝줄까지를 의미함 → 전체 파일내 모두변경
- 1,5 라고 표기하면 첫줄부터 다섯째 줄까지임

Tip

1,\$ s\$girl/boy/g 로 명령하는 경우 문서에 모든 girl이라는 단어가 boy로 변경됨

```
i am agirl
i am agirl
i am agirl i am agirl
~
~
:1,$s$girl/boy/g
```

<그림 II-20> 단어 모두 변경

3. 이해하기

3) 기타 유용한 명령어

vi편집기를 사용할 때 많이 사용되는 유용한 명령어를 알아본다.

① 여러 개 파일을 동시에 편집

- 여러 문서를 동시에 편집할 때 원하는 파일명을 나열하거나 범위표현으로 사용
- vi filename1 filename2 filename3
- vi *.dat
- vi편집기 사용 중 마지막행 명령모드에서 :n을 누르는 경우 다음 파일로 이동
- vi편집기 사용 중 마지막행 명령모드에서 :bn을 누르는 경우 다음 파일로 이동
- vi편집기 사용 중 마지막행 명령모드에서 :c#을 실행하면 이전 편집하던 파일로 이동 *순서 상관없이 직전에 편집한 파일로*

```
i am agirl
i am agirl
i am agirl i am agirl
~
~
:c#
```

<그림 II-21> 이전 작성하던 파일로 이동

Tip

다수의 파일을 편집할 때 변경된 파일을 저장하였는지를 고려하고 작업 필요

② 마지막행 명령모드에서 기타 명령

- :c filename 현재파일에서 다른 파일을 오픈. 명령모드에서 n을 눌러 다시 돌아감

● 3. 이해하기

- 개행문자 (CR, LF)의 처리

vi 마지막행명령보드에서 :%!xxd

원래대로 되돌릴때 :%!xxd -r

man ascii

```
1 00000000: 6162 630a 6465 660a 616b 730a 0d: carriage return
   a b c d e f a k s 0a: line feed
```

	8진법	10진법	16진법	
000 0	00	NUL	'\0' (null character)	100 64 40 @
001 1	01	SOH	(start of heading)	101 65 41 A
002 2	02	STX	(start of text)	102 66 42 B
003 3	03	ETX	(end of text)	103 67 43 C
004 4	04	EOT	(end of transmission)	104 68 44 D
005 5	05	ENQ	(enquiry)	105 69 45 E
006 6	06	ACK	(acknowledge)	106 70 46 F
007 7	07	BEL	'\a' (bell)	107 71 47 G
010 8	08	BS	'\b' (backspace)	110 72 48 H
011 9	09	HT	'\t' (horizontal tab)	111 73 49 I
012 10	0A	LF	'\n' (new line)	112 74 4A J
013 11	0B	VT	'\v' (vertical tab)	113 75 4B K
014 12	0C	FF	'\f' (form feed)	114 76 4C L
015 13	0D	CR	'\r' (carriage ret)	115 77 4D M
016 14	0E	SO	(shift out)	116 78 4E N
017 15	0F	SI	(shift in)	117 79 4F O
020 16	10	DLE	(data link escape)	120 80 50 P
021 17	11	DC1	(device control 1)	121 81 51 Q
022 18	12	DC2	(device control 2)	122 82 52 R
023 19	13	DC3	(device control 3)	123 83 53 S

Manual page ascii(7) line 18/136 28% (press h for help or q to quit)