

# 4. Linux Shell 사용하기

Shell

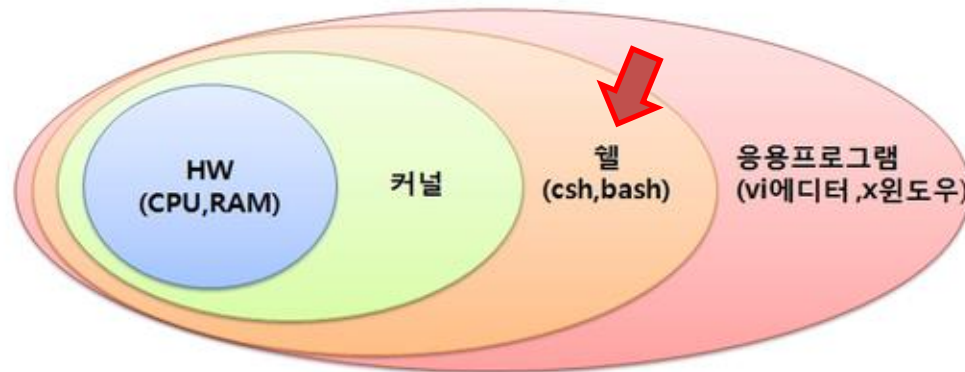
## 목 차

1. Shell 기능
2. Shell 종류
3. Bash Shell 기능
4. Shell 설정파일
5. Shell 정보 파일
6. Shell 환경변수

# 1. Shall 기능

## ● Shall 기능 - 터미널에서 해석기의 역할

- 1) 명령어 해석 기능-사용자와 커널 사이 중계
- 2) 프로그래밍 기능 - 셸 스크립트
  - ✓ GNOME 또는 '터미널'에서 명령어를 입력하는 환경
- 3) 사용자 환경 설정 기능 - 초기화 파일 기능 제공
  - ✓ 검색 경로, 터미널 종류, 환경 변수 등 설정
  - ✓ 로그인 되는 시점에서 실행시키고 싶은 명령
  - ✓ 시스템에 대한 정보를 제공하는 명령 수행



## 2. Shall 종류

### ● Shall 종류

- 1) Bourn Shall : 유닉스 V7에 처음 등장한 쉘(sh)
- 2) C Shall : Bourn Shall 기능 확장(csh)
- 3) Korn Shall : 벨연구소의 데이비드 콘 개발
- 4) **Bash Shall : CentOS 기본셸(bash)**
  - ✓ .bash\_profile : 로그인 할 때 읽어 들이는 설정 파일
- 5) Dash Shall – Bourn Shall 기반, 작은 크기, 속도 빠름
  - ✓ 우분투 6.10부터 Bourn Shall 대신 Dash Shall 사용

# 3. Bash Shell 기능

## ● Bash Shell 기능

1) Alias 기능 : 명령어 단축 기능

예) alias ls2 = "ls -al"

1) History 기능(↑, ↓ 방향키)

2) 자동 이름 완성 기능(Tab 키)

3) Shell 스크립트를 이용한 프로그래밍

✓ 설정 파일을 이용하여 시스템 제어

✓ 부팅 시 자동으로 어플리케이션 및 환경변수 실행

예) 실행파일 path 설정

지금까지  
해운 것

# 4. Shell 설정파일

환경변수는 어디에 설정할까?

- Shell 스크립트를 이용하여 시스템의 환경을 설정한 시스템 설정 파일

| 설정파일                   | 내용   |
|------------------------|--|
| <b>/etc/profile</b>    | <b>전역파일</b> , 관리자 모드의 환경변수와 bash 수행 시 실행되는 시스템 설정 파일 (Windows의 autoexec.bat 역할)                              |
| /etc/bashrc            | 전역파일, 별칭과 bash가 수행될 때 실행되는 함수 제어, 사용할 shell 지정(SHELL=/bin/bash)  |
| <b>~/.bash_profile</b> | <b>지역파일</b> , 일반사용자의 환경변수와 bash 수행 시 실행되는 시스템 설정파일, /etc/profile 수행된 후 바로 실행<br>실행시점 : 일반사용자 로그인(login) 시 실행 |
| ~/.bashrc              | 지역파일, 별칭과 bash가 수행될 때 실행되는 함수 제어<br>실행시점 : /etc/bashrc 수행된 후 바로 실행   |

/etc/전역파일(부모 셸) – 모든 사용자에게 영향 미치는 관리자 설정파일

~/지역파일(자식 셸) – 해당 사용자에게만 영향을 미치는 사용자 설정파일

# 5. Shall 환경변수

➤ 환경변수 : 환경설정에서 사용되는 변수(\$HOME, \$PATH 등)

→ 프로그램이 실행될 때 동작 방식이나 위치 등을 알려주는 시스템 변수

## ● 환경 변수의 특징

- ✓ 타입 없음
- ✓ 대소문자 구분(주로 대문자 사용)
- ✓ 환경변수에 값 할당 : 환경변수 = 값 (예: PATH = \$HOME/test)
- ✓ 변수 사용 : \$환경변수 (예: cd \$PATH)

## ● 환경변수 확인 : ~]# echo \$환경변수 (예: echo \$PATH)

환경변수가 없으면 시스템은 어디서 프로그램을 찾아야 하는지,  
기본 경로는 어디인지, 어떤 셸을 써야 하는지 등을 알 수 없어요.

예:

ls 같은 명령어는 사실 /bin/ls에 위치해 있어요.  
시스템은 PATH를 보고 /bin에 ls가 있는 걸 찾아 실행합니다.

## ● Shell 스크립트에서 환경변수 사용 예

```
hadoop@localhost:~  
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)  
[hadoop@localhost ~]$ cat .bash  
.bash_history .bash_logout .bash_profile .bashrc  
[hadoop@localhost ~]$ cat .bash_profile  
# .bash_profile  
  
# Get the aliases and functions  
if [ -f ~/.bashrc ]; then  
    . ~/.bashrc  
fi  
  
# User specific environment and startup programs  
  
PATH=$PATH:$HOME/.local/bin:$HOME/bin  
  
export PATH  
[hadoop@localhost ~]$
```

출력

환경변수

환경변수 내보내기 : \$PATH 환경변수 수정 반영

홈 디렉토리에 있는 .bash\_profile 파일의 내용을 출력

.bash\_profile: 시작할 때 자동으로 실행되는 명령어하고 환경변수 path를 보여줌

PATH와 HOME 환경변수를  
이용하여 PATH 환경변수  
수정

# 주요 환경변수

## ● 의미가 이미 정해진 환경변수

- 1) HOME : 사용자의 홈디렉토리
- 2) PATH : 실행파일을 찾는 경로
- 3) LANG : 프로그램 사용시 기본 지원되는 언어
- 4) PWD : 사용자의 현재 작업하는 디렉토리
- 5) TERM : 로긴 터미널 타입
- 6) SHELL : 로그인해서 사용하는 셸
- 7) USER : 사용자의 이름
- 8) DISPLAY : X 디스플레이 이름
- 9) VISUAL : visual 편집기의 이름
- 10) EDITOR : 기본 편집기의 이름
- 11) COLUMNS : 현재 터미널이나 컬럼수
- 12) PS1 : 명령프롬프트변수
- 13) PS2 : 2차 명령프롬프트(₩ 사용 명령행 연장)
- 14) BASH : 사용하는 bash 셸의 경로
- 15) BASH\_VERSION : bash의 버전

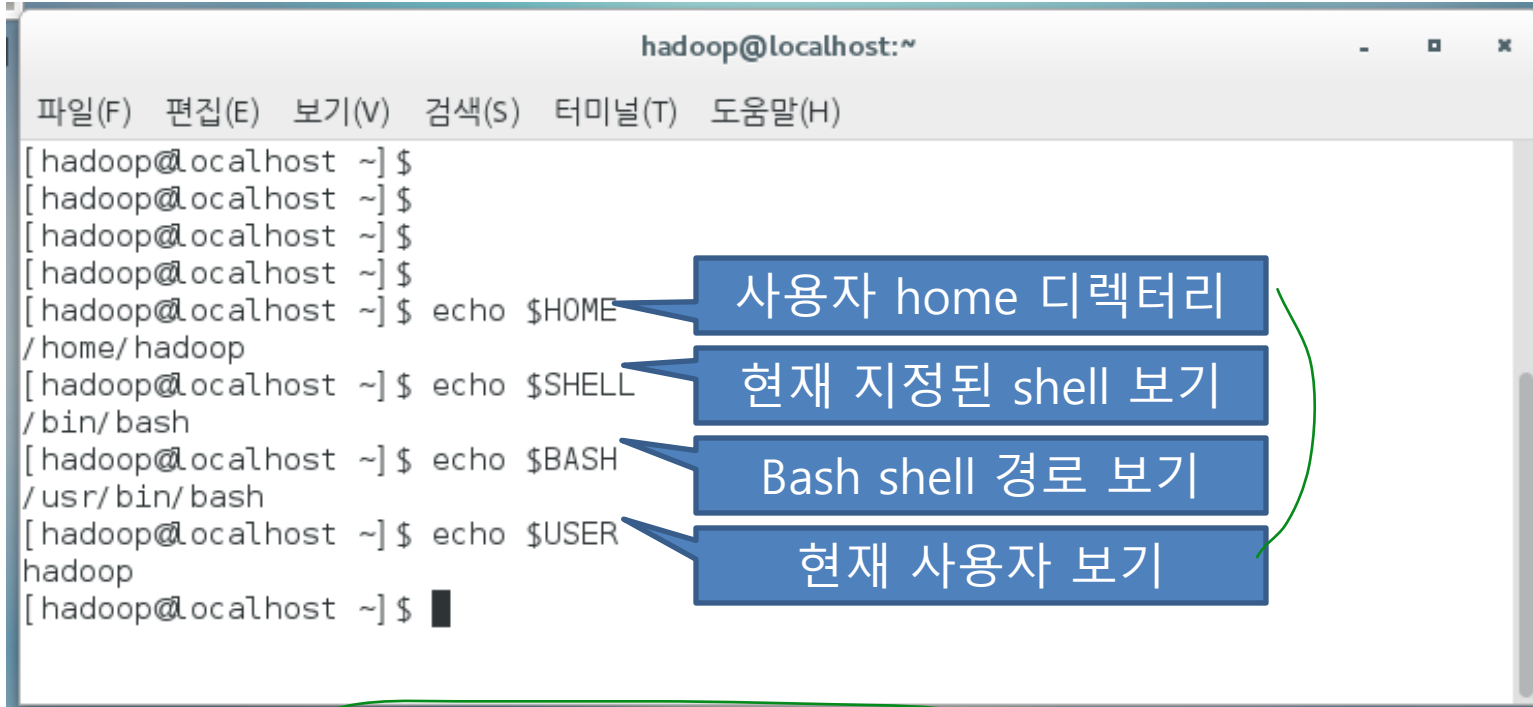
- 1) HISTFILE : history 파일의 경로
- 2) HISTFILESIZE : history 파일의 크기
- 3) HISTSIZE : history에 저장되는 갯수
- 4) HISTCONTROL : 중복 명령에 대한 기록 유무 지정
- 5) HOSTNAME : 호스트의 이름
- 6) LINES : 터미널의 라인 수
- 7) LOGNAME : 로그인이름
- 8) LS\_COLORS : ls 명령의 색상관련 옵션
- 9) MAIL : 메일을 보관하는 경로
- 10) MAILCHECK : 메일확인시간
- 11) OSTYPE : 운영체제 타입
- 12) SHLVL : 셸의 레벨
- 13) TERM : 터미널종류
- 14) UID : 사용자의 UID
- 15) USERNAME : 사용자이름



변수 안의 값을 쓰려면 \$를 붙여야 함  
echo HOME하면 그냥 'HOME' 텍스트 출력

## ● 환경변수 실행 : echo \$환경변수

환경변수 보기 : \$환경변수



```
hadoop@localhost:~  
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)  
[hadoop@localhost ~]$  
[hadoop@localhost ~]$  
[hadoop@localhost ~]$  
[hadoop@localhost ~]$  
[hadoop@localhost ~]$ echo $HOME  
/home/hadoop  
[hadoop@localhost ~]$ echo $SHELL  
/bin/bash  
[hadoop@localhost ~]$ echo $BASH  
/usr/bin/bash  
[hadoop@localhost ~]$ echo $USER  
hadoop  
[hadoop@localhost ~]$
```

사용자 home 디렉터리

현재 지정된 shell 보기

Bash shell 경로 보기

현재 사용자 보기

"환경변수 값"을 보려면 echo를 써야 하고  
"어디서 어떻게 설정되어 있는지"를 보려면 cat을 써야 함

# [실습] 사용자 환경변수 만들기

- 디렉터리 만들기

```
[hadoop@localhost ~]$ mkdir hadoop123
```

```
[hadoop@localhost ~]$
```

```
[hadoop@localhost ~]$ vi .bash_profile
```

```
[hadoop@localhost ~]$
```

- 환경변수 추가

```
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)

# User specific environment and startup programs
PATH=$PATH: $HOME/. local/bin: $HOME/bin
export PATH
export HADOOP_HOME=$HOME/hadoop123
~
~
~
:wq
```

환경변수

만들게하는 명령어

만든 환경변수

이름

현재

놓 dir

환경변수

## • 환경설정 반영 & 환경변수 실행

```
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[hadoop@localhost ~]$ vi .bash_profile
[hadoop@localhost ~]$
[hadoop@localhost ~]$
[hadoop@localhost ~]$ source .bash_profile
[hadoop@localhost ~]$
[hadoop@localhost ~]$
[hadoop@localhost ~]$ cd $HADOOP_HOME
[hadoop@localhost hadoop123]$
[hadoop@localhost hadoop123]$
[hadoop@localhost hadoop123]$
```

환경설정 반영 *구글*

환경변수 이용  
디렉터리 변경

hadoop@localhost:~/hadoop123

## • 환경변수 보기

```
파일(F) 편집(E) 보기(V) 검색(S) 터미널(T) 도움말(H)
[hadoop@localhost hadoop123]$
[hadoop@localhost hadoop123]$ echo $PATH
/usr/local/bin:/usr/local/sbin:/usr/bin:/usr/sbin:/bin:/sbin:/home/hadoop/.l
ocal/bin:/home/hadoop/bin:/home/hadoop/.local/bin:/home/hadoop/bin
[hadoop@localhost hadoop123]$
[hadoop@localhost hadoop123]$
[hadoop@localhost hadoop123]$ echo $HADOOP_HOME
/home/hadoop/hadoop123
[hadoop@localhost hadoop123]$
```