

CLI Shell 제작

챕터의 포인트

- Shell 제작을 위한 준비
- 도전, CLI Shell 제작하기

Shell 제작을 위한 준비

System 함수로 shell 명령어를 수행 가능

- 실제로 bash를 실행시킨 후 shell 명령어를 수행하는 원리
- stdlib.h 인클루드 필요
- 아래 코드를 수행하면 ls 명령어가 그대로 출력 됨

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main()
5 {
6     system("echo HI");
7     system("ls");
8
9     return 0;
10 }
```

```
inho@inho-VirtualBox:~$ gcc ./test.c -o ./test
inho@inho-VirtualBox:~$ ./test
HI
a.c      bts.txt      meminfo  test.tc  공개      비디오
a.out    dino.jpg    snap     ti.c     다운로드 사진
abc      dinosaur-5995333_960_720.png test      win     문서      음악
abc.txt  gogo        test.c   work    바탕화면 템플릿
inho@inho-VirtualBox:~$
```

1. 셸 스크립트 명령어 수행
2. 셸 명령어 수행

dmesg : 커널 메시지를 출력하는 셸 명령어

- system diagnostic message, 시스템 진단 메시지의 줄임말
- 시스템이 시작된 후 부터, 어떤 작업을 했는지 Log Message를 볼 수 있다.

```
lnho@lnho-VirtualBox:~$ dmesg
[ 0.000000] Linux version 5.8.0-44-generic (buildd@lgw01-amd64-054) (gcc (Ubuntu 9.3.0-17ubuntu1~20.04) 9.3.0, G
buntu) 2.34) #50~20.04.1-Ubuntu SMP Wed Feb 10 21:07:30 UTC 2021 (Ubuntu 5.8.0-44.50~20.04.1-generic 5.8.18)
[ 0.000000] Command line: BOOT_IMAGE=/boot/vmlinuz-5.8.0-44-generic root=UUID=71bc22d6-82a0-4e56-9899-045cfd7ad6
[ 0.000000] KERNEL supported cpus:
[ 0.000000]   Intel GenuineIntel
[ 0.000000]   AMD AuthenticAMD
[ 0.000000]   Hygon HygonGenuine
[ 0.000000]   Centaur CentaurHauls
[ 0.000000]   zhaoxin   Shanghai
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x001: 'x87 floating point registers'
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x002: 'SSE registers'
[ 0.000000] x86/fpu: Supporting XSAVE feature 0x004: 'AVX registers'
[ 0.000000] x86/fpu: xstate_offset[2]: 576, xstate_sizes[2]: 256
[ 0.000000] x86/fpu: Enabled xstate features 0x7, context size is 832 bytes, using 'standard' format.
[ 0.000000] BIOS-provided physical RAM map:
[ 0.000000] BIOS-e820: [mem 0x0000000000000000-0x000000000009fbff] usable
[ 0.000000] BIOS-e820: [mem 0x000000000009fc00-0x000000000009ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000000f0000-0x00000000000fffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000000100000-0x0000000000dfffff] usable
[ 0.000000] BIOS-e820: [mem 0x000000000dff0000-0x000000000dfffffff] ACPI data
[ 0.000000] BIOS-e820: [mem 0x00000000fec00000-0x00000000fec0ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fee00000-0x00000000fee0ffff] reserved
[ 0.000000] BIOS-e820: [mem 0x00000000fffc0000-0x00000000ffffffff] reserved
[ 0.000000] BIOS-e820: [mem 0x0000000100000000-0x000000011fffffff] usable
[ 0.000000] NX (Execute Disable) protection: active
```

date 명령어

- 현재 시간을 확인할 수 있다. date

```
inho@inho-VirtualBox:~$ date
2021. 03. 05. (금) 16:26:35 KST
inho@inho-VirtualBox:~$ █
```

uptime 명령어

- 현재시간, 시스템 시간과 커널이 부팅 된 후(up) 지난 시간 을 나타낸다.
- 그 외, 접속자 수와 CPU 시스템 부하율 (1분, 5분, 15분 평균치) 나타낸다.

```
inho@inho-VirtualBox:~$ uptime
16:25:46 up 15:38, 1 user, load average: 0.00, 0.00, 0.00
inho@inho-VirtualBox:~$
inho@inho-VirtualBox:~$
```

도전, CLI Shell 제작하기

SSAFY 셸을 제작하자

- 프롬프트 : **SSAFY > _**
- 1 단계, 명령어 입력
 - date 입력시 : date 출력
 - uptime 입력 : uptime 출력
 - ls 입력시 : ls -al 결과 출력
 - log 입력시 : dmesg 출력
 - exit 입력시 : 셸 종료
 - 정확하지 않는 명령어 입력시 : "ERROR";
- 2 단계, history 입력
 - 명령어 입력한 내역 모두 출력하기 (번호와 함께 입력)
 - ! 번호 입력 시 해당 명령어 입력하기
 - "hclear" 입력시 history 모두 삭제

```
nojin@nojin-VirtualBox:~$ ./ssafy
SSAFY > date
2022. 03. 22. (화) 21:57:08 KST
SSAFY > uptime
 21:57:10 up  1:35,  1 user,  load average: 0.09
SSAFY > log
[ 0.000000] Linux version 5.13.0-35-generic (
Ubuntu) 2.34) #40~20.04.1-Ubuntu SMP Mon Mar 7 6
[ 0.000000] Command line: BOOT_IMAGE=/boot/vr
ash
[ 0.000000] KERNEL supported cpus:
[ 0.000000]   Intel GenuineIntel
[ 0.000000]   AMD AuthenticAMD
[ 0.000000]   Hygon HygonGenuine
[ 0.000000]   Centaur CentaurHauls
[ 0.000000]   zhaoxin   Shanghai
[ 0.000000] x86/fpu: Supporting XSAVE feature
[ 0.000000] x86/fpu: Supporting XSAVE feature
[ 0.000000] x86/fpu: Supporting XSAVE feature
[ 0.000000] x86/fpu: xstate_offset[2]: 576,
[ 0.000000] x86/fpu: Enabled xstate features
[ 0.000000] BIOS-provided physical RAM map:
nojin@nojin-VirtualBox:~$ ./ssafy
SSAFY > date
2022. 03. 22. (화) 22:01:42 KST
SSAFY > update
ERROR
SSAFY > uptime
 22:01:51 up  1:40,  1 user,  load average: 0.08, 0.04, 0.04
SSAFY > sl
ERROR
SSAFY > history
1 date
2 update
3 uptime
4 sl
5 history
SSAFY > |
```