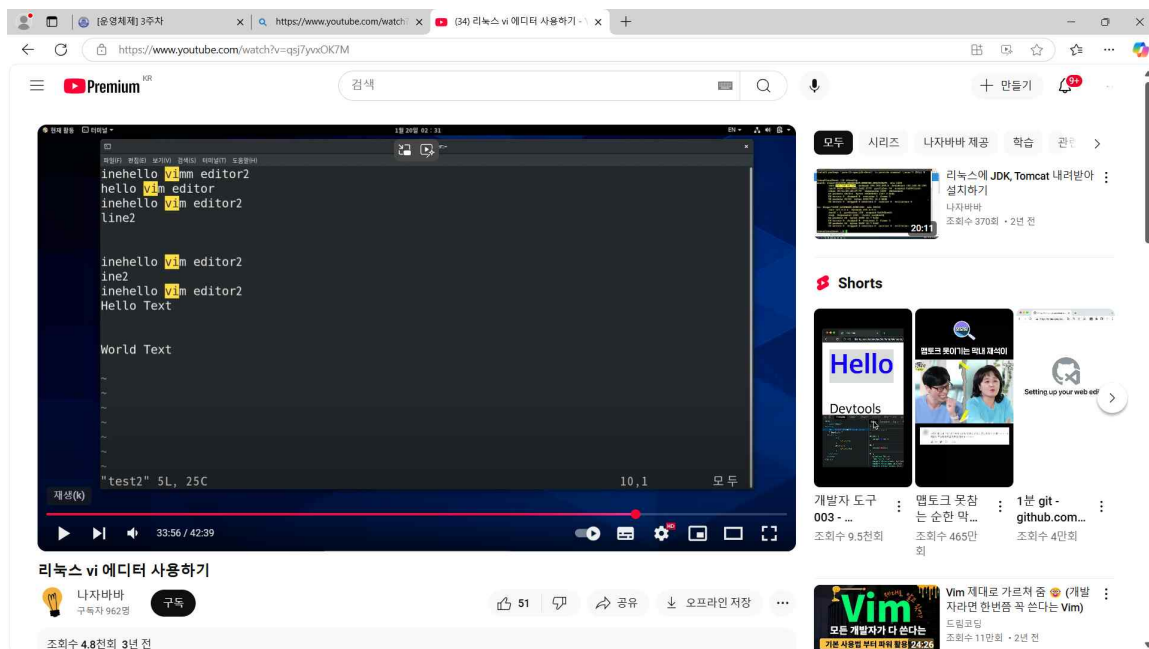


컴퓨터운영체제 3주차 과제

20220552 이서준

1) vi 사용법 학습하기



2) 복합문제 6번 풀이

2-1) fork() 후 어느 부분이 부모 프로세스의 코드와 자식 프로세스의 코드인가?

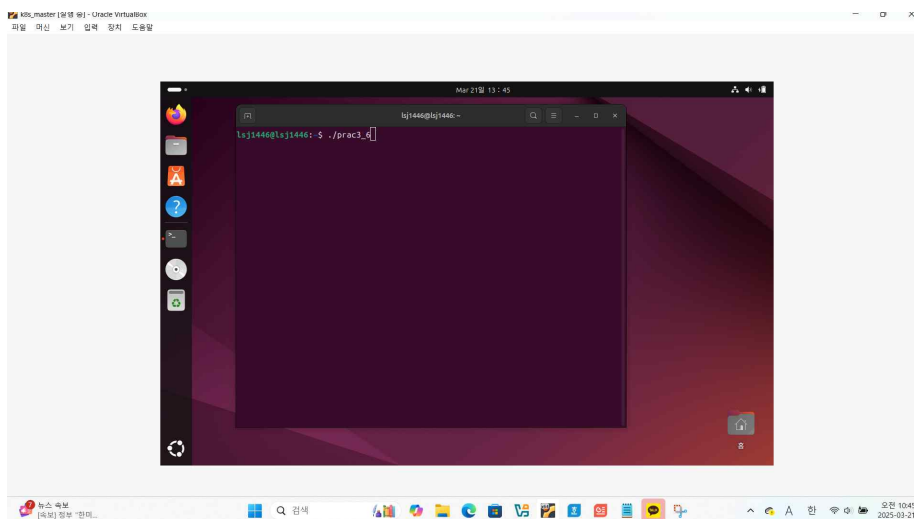
-> 부모 프로세스의 코드는 if(pid > 0) 블록 내의 코드이고,
자식 프로세스의 코드는 else if(pid == 0) 블록 내의 코드이다.

2-2) 고아 프로세스가 발생하는가?

-> 이 코드에서 고아 프로세스는 발생하지 않는다. 왜냐하면 고아

프로세스는 부모 프로세스가 자식 프로세스보다 먼저 종료되고, 자식 프로세스가 종료되지 않으면 발생되기 때문이다.

2-3) 이 프로그램의 실행 결과는 무엇이라고 예측되는가? 프로그램을 실행시켜 결과를 확인하고 실행 결과에 대한 이유를 설명하라.]



-> 이 프로그램의 실행 결과는

부모 프로세스의 pid = <부모 프로세스의 PID>

이다. fork() 함수 호출 후 부모 프로세스와 자식 프로세스가 분기하고, 부모 프로세스는 sleep(1)을 호출하여 1초 동안 대기하고 종료되며, 자식 프로세스는 sleep(2)을 호출하여 2초 동안 대기하다가 이후 부모 프로세스의 PID를 출력하므로, 실제 실행 중인 부모 프로세스의 PID 값이 출력된다.

3) 복합문제 7번 풀이

<1에서 10까지 더한 값을 종료 코드로 리턴하는 프로그램>

```
// sum.c
#include <stdio.h>
#include <stdlib.h>

int main(void) {
    int sum = 0;

    for (int i = 1; i <= 10; i++) {
        sum += i;
    }

    exit(sum);
}
```

<prac3_7.c 프로그램>

```
// prac3_7.c
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <unistd.h>

int main(void) {
    pid_t pid;
    int status;

    pid = fork();
```

```
if (pid > 0) {
    // 부모 프로세스
    waitpid(pid, &status, 0);

    if (WIFEXITED(status)) {
        int sum = WEXITSTATUS(status);
        printf("1에서 10까지 합친 결과는 %d\n", sum);
    }
}
else if (pid == 0) {
    execlp("./sum", "./sum", NULL);
    perror("execlp 실패");
    exit(1);
}
else {
    perror("fork 실패");
    exit(1);
}

return 0;
}
```