1. Server에서 Client에게 stop fifo 전달
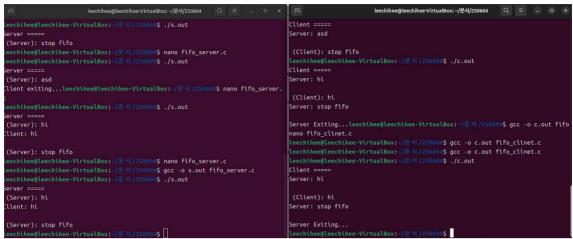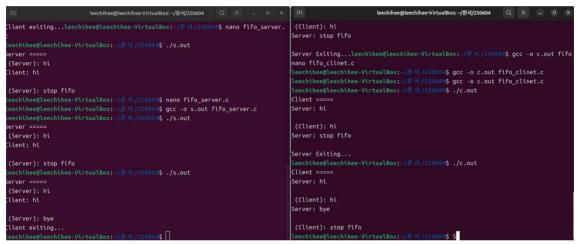

2. Client에서 Server로 stop fifo 전달

```c
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>

int main() {
    int to_client_fd, to_server_fd;
    char send_buf[100], recv_buf[100];
        const char stop[100] = "stop fifo\n";
    int n;


    printf("Server =====\n");
    // FIFO 파일 생성
    mkfifo("to_client_fifo", 0666);
    mkfifo("to_server_fifo", 0666);


    // FIFO 열기
    to_client_fd = open("to_client_fifo", O_WRONLY);
    to_server_fd = open("to_server_fifo", O_RDONLY);

    if (to_client_fd == -1 || to_server_fd == -1) {
        perror("open");
        exit(1);
    }

  while (1) {
        // 메시지 보내기
        printf(" (Server): ");
        fflush(stdout); //표준 출력 버퍼를 비움
```

```c
        fgets(send_buf, sizeof(send_buf), stdin);
        write(to_client_fd, send_buf, strlen(send_buf));
        // 메시지 받기
        n = read(to_server_fd, recv_buf, sizeof(recv_buf));
        if(strcmp(recv_buf,"stop fifo\n")==0)
        {
                printf("Client exiting...\n");
                break;
        }

        if(strcmp(send_buf,"stop fifo\n") == 0)
                break;

        if (n > 0) {
                recv_buf[n] = '\0';
                printf("Client: %s\n", recv_buf);
        }

    }

    close(to_client_fd);
    close(to_server_fd);
    return 0;
}
```

```c
#include <fcntl.h>
#include <unistd.h>
#include <stdlib.h>
#include <stdio.h>
#include <string.h>

int main() {
    int to_client_fd, to_server_fd;
    char send_buf[100], recv_buf[100];
    int n, flag = 0;

    printf("Client =====\n");

    // FIFO 열기
    to_client_fd = open("to_client_fifo", O_RDONLY);
    to_server_fd = open("to_server_fifo", O_WRONLY);

    if (to_client_fd == -1 || to_server_fd == -1) {
        perror("open");
        exit(1);
    }

while (1) {
        // 메시지 받기
        n = read(to_client_fd, recv_buf, sizeof(recv_buf));

        if (n > 0) {
            recv_buf[n] = '\0';
            printf("Server: %s\n", recv_buf);
        }
        // 메시지 보내기
        if(strcmp(recv_buf,"stop fifo\n")==0)
```

```c
        {
            printf("Server Exiting...\n");
            break;
        }
        printf(" (Client): ");
        fflush(stdout);
        fgets(send_buf, sizeof(send_buf), stdin);
        write(to_server_fd, send_buf, strlen(send_buf));

        if(strcmp(send_buf,"stop fifo\n")==0)
        {
            break;
        }


    }

    close(to_client_fd);
    close(to_server_fd);
    return 0;
}
```