

```

/*****
/* lily.c
/* fifo 이름과 open 하는 순서에 유의할 것 !
/*****
#include <stdio.h>
#include <stdlib.h>
#include <errno.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <string.h>

#define MSG_SIZE 1024

int main(int argc, char *argv[])
{
    char *fifo1 = "lily2cara";
    char *fifo2 = "cara2lily";

    char snd_buf[MSG_SIZE];
    char rcv_buf[MSG_SIZE];
    char msg_header[MSG_SIZE];
    int hd_length;

    int rfd;
    int sfd;

    fd_set master_set, read_set;
    int rd_cnt;

    if (argc != 2) {
        printf("usage: %s user_name \n", argv[0]);
        exit(1);
    }

    memset(msg_header, 0, MSG_SIZE);
    sprintf(msg_header, "%s : ", argv[1]);
    hd_length = strlen(msg_header);

    if (mkfifo(fifo1, 0666) == -1) {
        if (errno != EEXIST) {
            fprintf(stderr, "Error in mkfifo %s\n", fifo1);
            exit(2);
        }
    }
}

```

```

if (mkfifo(fifo2, 0666) == -1) {
    if (errno != EEXIST) {
        fprintf(stderr, "Error in mkfifo %s\n", fifo2);
        exit(3);
    }
}

if ((rfd = open(fifo2, O_RDONLY)) < 0) {
    fprintf(stderr, "Error in open %s\n", fifo2);
    exit(4);
}

if ((sfd = open(fifo1, O_WRONLY)) < 0) { // blocking until open for reading
    fprintf(stderr, "Error in open %s\n", fifo1);
    exit(5);
}

FD_ZERO(&master_set);
FD_SET(0, &master_set);
FD_SET(rfd, &master_set);

while (read_set = master_set, select(sfd+1, &read_set, NULL, NULL, NULL) > 0) {
    // sfd + 1

    if (FD_ISSET(0, &read_set)) {
        if ((rd_cnt = read(0, snd_buf, MSG_SIZE)) <= 0) {
            fprintf(stderr, "Error in read from stdin\n");
            exit(6);
        }
        else {
            if (write(sfd, msg_header, hd_length) != hd_length) {
                fprintf(stderr, "Error in write to fifo\n");
                exit(7);
            }
            if (write(sfd, snd_buf, rd_cnt) != rd_cnt) {
                fprintf(stderr, "Error in write to fifo\n");
                exit(7);
            }
        }
    }

    if (FD_ISSET(rfd, &read_set)) {
        if ((rd_cnt = read(rfd, rcv_buf, MSG_SIZE)) <= 0) {
            fprintf(stderr, "Error in read from fifo\n");
            exit(8);
        }
    }
}

```

```
    }
    else {
        if (write(1, rcv_buf, rd_cnt) != rd_cnt) {
            fprintf(stderr, "Error in write to stdout\n");
            exit(9);
        }
    }
}
}
```