

## Pipes con nombre

/\* escritor.c \*/

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

main(int argc, char *argv[]) {

    char buff[80];
    int ret, num, fd_pipe;

    fd_pipe = open("mipipe", O_WRONLY);
    if ((fd_pipe < 0) && (errno == ENOENT)) {
        mknod("mipipe", S_IFIFO|0660,0);
        fd_pipe = open("mipipe", O_WRONLY);
    }

    while ( (ret = read(0,buff,sizeof(buff)))> 0) {
        buff[ret]='\0';
        num = atoi(buff);
        write(fd_pipe,&num,sizeof(num));
    }

    close(fd_pipe);
}
```

/\* lector.c \*/

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

main(int argc, char *argv[]) {

    char buff[80];
    int ret, num, fd_pipe,fd;

    fd_pipe = open("mipipe", O_RDONLY);
    fd = open (argv[1], O_WRONLY|O_CREAT|O_TRUNC, 0600);

    while ( (ret = read(fd_pipe, &num,sizeof(num)))> 0) {
        sprintf(buff,"%d\n",num);
        write(fd,buff,strlen(buff));
    }

}
```

## Pipes sin nombre

```
/*launch.c*/
```

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

main(int argc, char * argv[] ){
    int ret;
    int fd_pipe[2];

    pipe(fd_pipe);

    ret = fork();

    if (ret == 0) {
        dup2(fd_pipe[1],1);
        close(fd_pipe[1]);
        close(fd_pipe[0]);
        execlp("./escritor_v2", "escritor_v2",(char *)0);
        perror("Mutando a escritor");
        exit(1);
    }

    close(fd_pipe[1]);
    ret = fork();
    if (ret == 0) {
        dup2(fd_pipe[0],0);
        close(fd_pipe[0]);
        execlp("./lector_v2", "lector_v2",argv[1], (char *)0);
        perror("Mutando a lector");
        exit(1);
    }
    close(fd_pipe[0]);

    while (waitpid(-1,NULL,0)>0);

}
```