

Users > moshika > Desktop > c server.c

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  #include <unistd.h>
5  #include <arpa/inet.h>
6
7  #define PORT 8080
8  #define BUF_SIZE 1024
9
10 int main() {
11     int server_fd, client_fd;
12     char buffer[BUF_SIZE];
13     struct sockaddr_in server, client;
14     socklen_t client_len = sizeof(client);
15
16     // Create TCP socket
17     server_fd = socket(AF_INET, SOCK_STREAM, 0);
18
19     server.sin_family = AF_INET;
20     server.sin_addr.s_addr = INADDR_ANY;
21     server.sin_port = htons(PORT);
22
23     bind(server_fd, (struct sockaddr*)&server, sizeof(server));
24     listen(server_fd, 1);
25     printf("Server started, waiting for client...\n");
26
27     client_fd = accept(server_fd, (struct sockaddr*)&client, &client_len);
28     printf("Client connected!\n");
29
30     // Create a file on server side
31     FILE *fp = fopen("server_file.txt", "w");
32     if (!fp) {
33         perror("File creation error");
34         close(client_fd);
35         close(server_fd);
36         return 1;
37     }
38     fprintf(fp, "Hello! This file is sent from server to client.\n");
39     fprintf(fp, "This is a sample TCP file transfer.\n");
40     fclose(fp);
41
```

```
42 // Open file for reading and send to client
43 fp = fopen("server_file.txt", "rb");
44 if (!fp) {
45     perror("File open error");
46     close(client_fd);
47     close(server_fd);
48     return 1;
49 }
50
51 int n;
52 while ((n = fread(buffer, sizeof(char), BUF_SIZE, fp)) > 0) {
53     write(client_fd, buffer, n);
54 }
55
56 printf("File sent successfully!\n");
57
58 fclose(fp);
59 close(client_fd);
60 close(server_fd);
61
62 return 0;
63 }
```

C server.c

C client.c

Users > moshika > Desktop > C client.c

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  #include <unistd.h>
5  #include <arpa/inet.h>
6
7  #define PORT 8080
8  #define BUF_SIZE 1024
9
10 int main() {
11     int sockfd;
12     struct sockaddr_in server;
13     char buffer[BUF_SIZE];
14
15     sockfd = socket(AF_INET, SOCK_STREAM, 0);
16
17     server.sin_family = AF_INET;
18     server.sin_port = htons(PORT);
19     server.sin_addr.s_addr = inet_addr("127.0.0.1");
20
21     connect(sockfd, (struct sockaddr*)&server, sizeof(server));
22     printf("Connected to server. Receiving file...\n");
23
24     FILE *fp = fopen("received_file.txt", "wb");
25     if (!fp) {
26         perror("File creation error");
27         close(sockfd);
28         return 1;
29     }
30
31     int n;
32     while ((n = read(sockfd, buffer, BUF_SIZE)) > 0) {
33         fwrite(buffer, sizeof(char), n, fp);
34     }
35
36     printf("File received successfully!\n");
37
38     fclose(fp);
39     close(sockfd);
40
41     return 0;
42 }
```

Last login: Sun Dec 7 01:11:23 on ttys000

[moshika@Moshikas-MacBook-Air ~ % cd desktop

moshika@Moshikas-MacBook-Air desktop % gcc server.c -o server
gcc client.c -o client

[moshika@Moshikas-MacBook-Air desktop % ./server

Server started, waiting for client...

Client connected!

File sent successfully!

moshika@Moshikas-MacBook-Air desktop % █

```
Last login: Sun Dec  7 01:11:28 on ttys000
moshika@Moshikas-MacBook-Air ~ % cd desktop
moshika@Moshikas-MacBook-Air desktop % ./client
Connected to server. Receiving file...
File received successfully!
moshika@Moshikas-MacBook-Air desktop %
```