

Name: Preyash

Date: 20-01-2022

Registration Number: 20BPS1022

LAB-02

Socket Programming

AIM: To print a message from the server to the client utilizing the socket functions, and vice versa

ALGORITHM

Server-side Algorithm

1. Start.
2. Declare variables.
3. Take the port number as input from the user.
4. Using socket(), create TCP socket for server.
5. Using bind(), bind the socket to server address.
6. Using listen(), put the server socket in a passive mode, where it waits for the client to approach the server to make a connection
7. Using accept(), at this point, connection is established between client and server, and they are ready to transfer data.
8. Sending a message to client side.
9. Printing the message received from client side.

Client-Side Algorithm

1. Start.
2. Declare variables.
3. Take port as input from user.
4. Using `socket()`, create TCP socket for client.
5. Using `connect()`, connect the client to server to exchange information.
6. Using `recv()`, receive the data from server side.
7. If the connection will be successful data will be revived.
8. The message from server side is printed at client side.
9. The client will acknowledge by sending a message.

Server Program Source Code:

```
day_server_modified.c
~/Netcom1022/SocketProgramming

1 #include<stdio.h>
2 #include<sys/types.h>
3 #include<netinet/in.h>
4 #include<string.h>
5 #include<time.h>
6 int main(){
7     int sd,sd2,nsd,clilen,sport,len;
8     int port;
9     time_t ticks;
10    char sendmsg[100],rcvmsg[100];
11    struct sockaddr_in servaddr,cliaddr;
12    printf("Enter the server port:\n");
13    scanf("%d",&sport);
14    printf("%d",sport);
15    sd=socket(AF_INET,SOCK_STREAM,0);
16    ticks=time(NULL);
17    strcpy(sendmsg,ctime(&ticks));
18    strcpy(sendmsg,"Namaste, this is Preyash! \n");
19    if(sd<0)
20        printf("Can't create \n");
21    else
22        printf("Socket is created\n");
23    servaddr.sin_family=AF_INET;
24    servaddr.sin_addr.s_addr=htonl(INADDR_ANY);
25    servaddr.sin_port=htons(sport);
26    sd2=bind(sd,(struct sockaddr*) &servaddr,sizeof(servaddr));
27    if(sd2<0)
28        printf("Can't bind\n");
29    else
30        printf(" Binded \n");
31    listen(sd,5);
32    clilen=sizeof(cliaddr);
33    nsd=accept(sd,(struct sockaddr *)&cliaddr,&clilen);
34    if(nsd<0)
35        printf("Can't accept\n");
36    else
37        printf("Accepted\n");
38    send(nsd,sendmsg,100,0);
39    recv(nsd,rcvmsg,100,0);
40    printf("Message read from the client: %s\n,rcvmsg);
41 }
```

Output:

```
preyash-20bps1022@Preyash-20BPS1022: ~/Net... x preyash-20bps1022@Preyash-20BPS1022: ~/Net... x
preyash-20bps1022@Preyash-20BPS1022:~/Netcom1022/SocketPrograming$ gcc day_server_modified.c
preyash-20bps1022@Preyash-20BPS1022:~/Netcom1022/SocketPrograming$ ./a.out
Enter the server port:
2906
2906Socket is created
Binded
Accepted
Message read from the client: Message succesfully recieved at client side!

preyash-20bps1022@Preyash-20BPS1022:~/Netcom1022/SocketPrograming$ █
```

Client Program Source Code:

```
Open ▼ [icon] day_client_modified.c Save [icon] [icon] [icon]
~/Netcom1022/SocketPrograming
1 #include<stdio.h>
2 #include<sys/types.h>
3 #include<netinet/in.h>
4 #include<string.h>
5 int main(){
6     int csd,cport,len;
7     char sendmsg[100],revmsg[100];
8     struct sockaddr_in servaddr;
9     printf("Enter the port \n");
10    scanf("%d",&cport);
11    printf("Port: ");
12    printf("%d",cport);
13    csd=socket(AF_INET,SOCK_STREAM,0);
14    if(csd<0)
15        printf("Can't create\n");
16    else
17        printf("Socket is created\n");
18    servaddr.sin_family=AF_INET;
19    servaddr.sin_addr.s_addr=htonl(INADDR_ANY);
20    servaddr.sin_port=htons(cport);
21    if(connect(csd,(struct sockaddr *)&servaddr,sizeof(servaddr))<0)
22        printf("Can't connect\n");
23    else
24        printf("Connected sucessfully\n");
25    recv(csd,revmsg,100,0);
26    printf("Message read from the server side: %s\n",revmsg);
27    strcpy(sendmsg,"Message succesfully recieved at client side!\n");
28    send(csd,sendmsg,100,0);
29 }
30
31 █
```

Output:

```
preyash-20bps1022@Preyash-20BPS1022: ~/Net... ×  preyash-20bps1022@Preyash-20BPS1022: ~/Net... ×
preyash-20bps1022@Preyash-20BPS1022:~/Netcom1022/SocketPrograming$ gcc day_client_modified.c
preyash-20bps1022@Preyash-20BPS1022:~/Netcom1022/SocketPrograming$ ./a.out
Enter the port
2906
Port: 2906Socket is created
Connected sucessfully
Message read from the server side:  Namaste, this is Preyash!

preyash-20bps1022@Preyash-20BPS1022:~/Netcom1022/SocketPrograming$
```

Result: A successful connection between the client and server is established.