

18CSC302J (Computer Networks Lab)

Lab session 3 - UDP ECHO CLIENT SERVER COMMUNICATION

Name :- Puneet Sharma

Reg. No. :- RA1911003010331

Class :-CSE F1

UDP SERVER: SERVER CODE:

```
#include<stdio.h>
#include<netinet/in.h>
#include<sys/types.h>
#include<sys/socket.h>
#include<netdb.h>
#include<string.h>
#include <unistd.h>
#include<stdlib.h>
#define MAX 80
#define PORT 43456
#define SA struct sockaddr
void func(int sockfd)
{
char buff[MAX];
int n,clen;
struct sockaddr_in cli;
clen=sizeof(cli);
for(;;)
{
bzero(buff,MAX);
recvfrom(sockfd,buff,sizeof(buff),0,(SA *)&cli,&clen);
printf("From client %s To client : ",buff);
bzero(buff,MAX);
n=0;
while((buff[n++]=getchar())!='\n');
sendto(sockfd,buff,sizeof(buff),0,(SA *)&cli,clen);
if(strncmp("exit",buff,4)==0)
{
printf("Server Exit...\n");
break;
}
}
}
int main()
{
int sockfd;
struct sockaddr_in servaddr;
sockfd=socket(AF_INET,SOCK_DGRAM,0);
if(sockfd==-1)
{
printf("socket creation failed...\n");
```

```

exit(0);
}
else
printf("Socket successfully created..\n");
bzero(&servaddr,sizeof(servaddr));
servaddr.sin_family=AF_INET;
servaddr.sin_addr.s_addr=htonl(INADDR_ANY);
servaddr.sin_port=htons(PORT);
if((bind(sockfd,(SA *)&servaddr,sizeof(servaddr)))!=0)
{
printf("socket bind failed...\n");
exit(0);
}
else
printf("Socket successfully binded..\n");
func(sockfd);
close(sockfd);
}

```

```

1 #include<stdio.h>
2 #include<netinet/in.h>
3 #include<sys/types.h>
4 #include<sys/socket.h>
5 #include<netdb.h>
6 #include<string.h>
7 #include<unistd.h>
8 #include<stdlib.h>
9 #define MAX 80
10 #define PORT 43456
11 #define SA struct sockaddr
12 void func(int sockfd)
13 {
14     char buff[MAX];
15     int n,clen;
16     struct sockaddr_in cli;
17     clen=sizeof(cli);
18     for(;;)
19     {
20         bzero(buff,MAX);
21         recvfrom(sockfd,buff,sizeof(buff),0,(SA *)&cli,&clen);
22         printf("From client %s To client : ",buff);
23         bzero(buff,MAX);
24         n=0;
25         while((buff[n++]=getchar())!='\n');
26         sendto(sockfd,buff,sizeof(buff),0,(SA *)&cli,&clen);
27         if(strcasecmp("exit",buff,4)==0)
28         {
29             printf("Server Exit...\n");
30             break;
31         }
32     }
33 }
34 int main()
35 {

```

331/udp_server_331.c - Run

Socket successfully created..
Socket successfully binded..
From client Puneet
To client : |

10:19 C and C++ Spaces: 4

AWVS (not connected)

UDP CLIENT: CLIENT CODE

```

#include<sys/socket.h>
#include<netdb.h>
#include<string.h>
#include<stdlib.h>
#include<unistd.h>
#include<stdio.h>
#define MAX 80
#define PORT 43456
#define SA struct sockaddr
int main()
{
char buff[MAX];

```

```

int sockfd,len,n;
struct sockaddr_in servaddr;
sockfd=socket(AF_INET,SOCK_DGRAM,0);
if(sockfd==-1)
{
printf("socket creation failed...\n");
exit(0);
}
else
printf("Socket successfully created..\n");
bzero(&servaddr,sizeof(len));
servaddr.sin_family=AF_INET;
servaddr.sin_addr.s_addr=inet_addr("127.0.0.2");
servaddr.sin_port=htons(PORT);
len=sizeof(servaddr);
for(;;)
{
printf("\nEnter string : ");
n=0;
while((buff[n++]=getchar())!='\n');
sendto(sockfd,buff,sizeof(buff),0,(SA *)&servaddr,len);
bzero(buff,sizeof(buff));
recvfrom(sockfd,buff,sizeof(buff),0,(SA *)&servaddr,&len);
printf("From Server : %s\n",buff);
if(strncmp("exit",buff,4)==0)
{
printf("Client Exit...\n");
break;
}
}
close(sockfd);
}

```

```

1 #include <sys/socket.h>
2 #include <netdb.h>
3 #include <string.h>
4 #include <stdlib.h>
5 #include <unistd.h>
6 #include <stdio.h>
7 #define MAX 80
8 #define PORT 43456
9 #define SA struct sockaddr
10 int main()
11 {
12     char buff[MAX];
13     int sockfd,len,n;
14     struct sockaddr_in servaddr;
15     sockfd=socket(AF_INET,SOCK_DGRAM,0);
16     if(sockfd==-1)
17     {
18         printf("socket creation failed...\n");
19         exit(0);
20     }
21     else
22     printf("Socket successfully created..\n");
23     bzero(&servaddr,sizeof(len));
24     servaddr.sin_family=AF_INET;
25     servaddr.sin_addr.s_addr=inet_addr("127.0.0.2");
26     servaddr.sin_port=htons(PORT);
27     len=sizeof(servaddr);
28     for(;;)
29     {
30         printf("\nEnter string : ");
31         n=0;
32         while((buff[n++]=getchar())!='\n');
33         sendto(sockfd,buff,sizeof(buff),0,(SA *)&servaddr,len);
34         bzero(buff,sizeof(buff));
35         recvfrom(sockfd,buff,sizeof(buff),0,(SA *)&servaddr,&len);

```

331/udp_server_331.c 331/udp_client_331.c - Run

Stop Command: 331/udp_client_331.c Runner: C CWD: ENV

Socket successfully created..
Enter string : Puneet

AWS (not connected)