LAB ASSIGNMENT – 7

Grihit Budhiraja

19BCE2141

Code –

Server Side

```
#include <stdio.h>
#include <netdb.h>
#include <netinet/in.h>
#include <stdlib.h>
#include <string.h>
#include <sys/socket.h>
#include <sys/types.h>
#define MAX 80
#define PORT 8080
#define SA struct sockaddr

void func(int sockfd)
{
	char buff[MAX];
	int n;
	for (;;)
	{
		bzero(buff, MAX);
		read(sockfd, buff, sizeof(buff));
```

```
            printf("From client: %s\t To client: ", buff);
            bzero(buff, MAX);
            n = 0;
            while ((buff[n++] = getchar()) != '\n')
                ;
            write(sockfd, buff, sizeof(buff));
            if (strncmp("exit", buff, 4) == 0) {
                printf("Server Exit\n");
                break;
            }
        }
}
int main()
{
        int sockfd, connfd, len;
        struct sockaddr_in servaddr, cli;
        sockfd = socket(AF_INET, SOCK_STREAM, 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 binded\n");

if ((listen(sockfd, 5)) != 0) {
        printf("Listen failed\n");
        exit(0);
}
else
        printf("Server listening\n");
len = sizeof(cli);

connfd = accept(sockfd, (SA*)&cli, &len);
if (connfd < 0) {
        printf("Server acccept failed\n");
        exit(0);
}
else
        printf("Server acccept the client\n");
```

```
        func(connfd);
        close(sockfd);
}
```

Client Side

```
#include <netdb.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/socket.h>
#define MAX 80
#define PORT 8080
#define SA struct sockaddr
void func(int sockfd)
{
        char buff[MAX];
        int n;
        for (;;) {
                bzero(buff, sizeof(buff));
                printf("Enter the string : ");
                n = 0;
                while ((buff[n++] = getchar()) != '\n')
                        ;
                write(sockfd, buff, sizeof(buff));
                bzero(buff, sizeof(buff));
                read(sockfd, buff, sizeof(buff));
                printf("From Server : %s", buff);
```

```
            if ((strncmp(buff, "exit", 4)) == 0) {
                printf("Client Exit...\n");
                break;
            }
        }
}

int main()
{
        int sockfd, connfd;
        struct sockaddr_in servaddr, cli;

        sockfd = socket(AF_INET, SOCK_STREAM, 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 = inet_addr("127.0.0.1");
        servaddr.sin_port = htons(PORT);

        if (connect(sockfd, (SA*)&servaddr, sizeof(servaddr)) != 0) {
```

```
        printf("connection with the server failed...\n");
        exit(0);
    }
    else
        printf("connected to the server..\n");

    func(sockfd);
    close(sockfd);
}
```

Output –

Server Side

```
grihit@DESKTOP-J4LEGSE:/mnt/d/Study Material/SEM 4/NETCOM/LAB/Socket TCP$ ./server
Socket successfully created
Socket binded
Server listening
Server acccept the client
From client: Hello
        To client: Hi Client
From client: It is nice chatting to you
        To client: Thank you. How are you?
From client: I am good
        To client: exit
Server Exit
grihit@DESKTOP-J4LEGSE:/mnt/d/Study Material/SEM 4/NETCOM/LAB/Socket TCP$
```

Client Side

```
grihit@DESKTOP-J4LEGSE:/mnt/d/Study Material/SEM 4/NETCOM/LAB/Socket TCP$ ./client
Socket successfully created..
connected to the server..
Enter the string : Hello
From Server : Hi Client
Enter the string : It is nice chatting to you
From Server : Thank you. How are you?
Enter the string : I am good
From Server : exit
Client Exit...
grihit@DESKTOP-J4LEGSE:/mnt/d/Study Material/SEM 4/NETCOM/LAB/Socket TCP$
```