#include<stdio.h>

#include<string.h>

#include<stdlib.h>

#include<sys/socket.h>

#include<arpa/inet.h>

#include<unistd.h>

#include<pthread.h>

void \* todo\_client(void \* socket\_descriptor){

int socket\_desc = \* (int\*) socket\_descriptor;

int read\_size;

char \*message,client\_message[2000];

// send some message

message= "Greeting ! I am Mohit kumar's connection handler\n";

write(socket\_desc,message,strlen(message));

message="Type Message \n";

write(socket\_desc,message,strlen(message));

while((read\_size= recv(socket\_desc,client\_message,2000,0))>0){

client\_message[read\_size]='\0';

write(socket\_desc,client\_message,strlen(client\_message));

memset(client\_message,0,2000);

}

if( read\_size ==0 ){

puts("Client disconnected\n");

fflush(stdout);

}

return 0;

}

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

int socket\_server,socket\_client,sizeaddr;

struct sockaddr\_in server\_addr, client\_addr;

// socket creation

socket\_server=socket(AF\_INET,SOCK\_STREAM,0);

if(socket\_server==-1){

printf("Could Not Created");

exit(1);

}

puts("Socket Created");

server\_addr.sin\_family=AF\_INET;

server\_addr.sin\_addr.s\_addr=INADDR\_ANY;

server\_addr.sin\_port=htons(8888);

if( bind(socket\_server,(struct sockaddr\*)&server\_addr,sizeof(server\_addr))<0){

perror("Error in bind");

return 1;

}

puts("Binding has been completed");

// listen

listen(socket\_server,3);

// accept the request

puts("Waiting for clients....\n");

sizeaddr=sizeof(struct sockaddr\_in);

pthread\_t thread\_id;

while((socket\_client=accept(socket\_server,(struct sockaddr\*)&client\_addr,(socklen\_t\*)&sizeaddr))){

puts("Connection accepted");

if(pthread\_create(&thread\_id,NULL,todo\_client,(void\*)& socket\_client)<0){

perror("thread could not be created\n");

return 1;

}

// now join

pthread\_join(thread\_id,NULL);

puts("Headers assigned");

}

if( socket\_client<0){

perror("Accept failed");

return 1;

}

return 0;

}

#include<sys/socket.h>

#include<netinet/in.h>

#include<arpa/inet.h>

#include<stdio.h>

#include<stdlib.h>

#include<unistd.h>

#include<errno.h>

#include<string.h>

#include<unistd.h>

#include<sys/types.h>

#include<fcntl.h>

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

// define client socket

int client\_socket;

client\_socket=socket(AF\_INET,SOCK\_STREAM,0);

// define client address

struct sockaddr\_in server\_address;

server\_address.sin\_family= AF\_INET;

server\_address.sin\_addr.s\_addr= INADDR\_ANY;

server\_address.sin\_port= htons(8888);

int connection\_status=connect(client\_socket,(struct sockaddr\*)&server\_address,sizeof(server\_address));

if( connection\_status ==-1){

printf("Error happens\n");

exit(1);

}

char buff[256];

read(client\_socket,buff,sizeof(buff));

printf("Message received: %s\n",buff);

char \*msg;

msg=(char\*)malloc(sizeof(char)\*50);

printf("Enter message for server:");

fgets(msg,50,stdin);

write(client\_socket,msg,strlen(msg));

close(client\_socket);

return 0;

}



