Java Server

import java.net.\*;

import java.io.\*;

import java.util.\*;

class DateServer

{

public static void main(String args[]) throws Exception

{

ServerSocket s=new ServerSocket(43454);

while(true)

{

System.out.println("Waiting For Connection ...");

Socket soc=s.accept();

DataOutputStream out=new DataOutputStream(soc.getOutputStream());

out.writeBytes("Server Date: " + (new Date()).toString() + "\n");

out.close();

soc.close();

}

}

}

C client

#include <sys/socket.h>

#include <sys/types.h>

#include <netinet/in.h>

#include <netdb.h>

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#include <unistd.h>

#include <errno.h>

#include <arpa/inet.h>

int main()

{

int CreateSocket = 0,n = 0;

char dataReceived[1024];

struct sockaddr\_in ipOfServer;

memset(dataReceived, '0' ,sizeof(dataReceived));

if((CreateSocket = socket(AF\_INET, SOCK\_STREAM, 0))< 0)

{

printf("Socket not created \n");

return 1;

}

ipOfServer.sin\_family = AF\_INET;

ipOfServer.sin\_port = htons(43454);

ipOfServer.sin\_addr.s\_addr = inet\_addr("192.168.48.128");

if(connect(CreateSocket, (struct sockaddr \*)&ipOfServer, sizeof(ipOfServer))<0)

{

printf("Connection failed due to port and ip problems\n");

return 1;

}

while((n = read(CreateSocket, dataReceived, sizeof(dataReceived)-1)) > 0)

{

dataReceived[n] = 0;

if(fputs(dataReceived, stdout) == EOF)

{

printf("\nStandard output error");

}

printf("\n");

}

if( n < 0)

{

printf("Standard input error \n");

}

return 0;

}