#include <SPI.h>//inposta la libreria per la comunicazione tra arduino e scheal

#include <Ethernet.h>//importa la libreria per la creazione di server

#include <Servo.h>//include la libreria per il pilotaggio del servo

byte mac[]={0x7B,0xF8,0xA9,0xF9,0x20,0x0A};//crea un array per l'indirizzo mac

IPAddress ip (10, 0, 10, 210);//crea una variabile contenuta l'indirizzo ip

EthernetServer server(80);// crea un server e specifica la porta di comunicazione

int posizione;//crea la variabile comunicazione

String buff=String();//crea la variabile biff è la svuota

Servo servo;//da un nome specifico ad un servo

void setup()//setup dati

{

Serial.begin(9600);//velocità della comunicazione tramite monitor seriale

Ethernet.begin(mac,ip);//inserite l'ip e l'indirizzo mac della scheda di rete

server.begin();//fa partire il server

servo.attach(6);//specifica la porta dove è attaccato li servo

servo.write(90);//specifica la velocità di scrittura del servo

}

void loop()//loop

{

EthernetClient client= server.available();//crea client è setta il server available

if(client){//controlla se client e vera

boolean vuota=true;//crea la variabile vuota

String line= String();//la variabile line viene svuotata

while( client.connected() ){//inizia un ciclo affinché il client è connesso

if(client.available()){//controlla se li client e disponibile

char c= client.read();//creazione variabile che contiene la lettura di co che a in viato li client

line.concat(c);//line viene connessa con la variabile c

if((c=='\n')&& vuota){//controlla se c è uguale a capo e vela variabile vuota e vera

client.println("HTTP/1.1 200 OK");//versione html

client.println("Content-Type: text/html");//tipo della pagina

client.println("connection:close");//specifica che la connessione è chiusa

client.println("");

client.println("<!DOCTYPE HTML>");//specifica la protezione del codice

client.println("<html>");//apre html

client.println("<style>");//apre lo stile per css

client.println("html {background-color: powderblue;}");//setta a html li colore di fondo a powder blue

client.println("html {text-align: center;}");//setta tutto all'interno di html al centro della pagina

client.println("h1 {color: blue;}");//setta il colore di h1 a blue

client.println("</style>");//chiusura stile css

client.println("<h1>inserie la posizione in gradi</h1>");//scrittura in maiuscolo di una frase

client.println("<form action='/' method='get'> <lable for='valore'>valore</label> <input type='text' id='valore' name='valore'> <imput type='submit' value='invio'> </form>");//crea un casella per inserire il valore poi lo inserisce nel url

client.println("</html>");//chiude html

break;//esce dal if

}

if(c=='\n'){//controlla se la variabile c è uguale a capo

vuota=true;//mette la variabile vuota a vero

Serial.println(line);//stampa la variabile string

if(line.indexOf("=")==35)//controlla se nella stringa line esiste li carretera =

{

Serial.println("HIT");//stampa hit

buff=line.substring(line.indexOf("=")+1, line.indexOf("\r"));//metta nella variabile buff i dati della variabile line dopo la = e prima della fina della variabile

Serial.println(buff);//stampa la variabile buff

posizione=buff.toInt() ;//mette nella variabile posizione la conversione in int della variabile buff

Serial.println(posizione);//stampa la variabile posizione

}

line=String();//svuota la variabile linea

}

else{//percorso se l'if e falso

if(c!='\r'){//controlla se la variabile c è diversa dal /r

vuota=false;//setta la variabile vuota a falso

}

}

}

}

}

servo.write(posizione);//manda la variabile posizione al serov per farlo spostare

client.stop();//ferma li client

Ethernet.maintain();//mantiene la connessione

}