

## ARDUINO CODE

```
#include
<Servo.h>

Servo gripper;
Servo wrist;
Servo elbow;
Servo shoulder;
Servo base;
double base_angle=90;
double shoulder_angle=90;
double elbow_angle=90;
double wrist_angle=90;
void setup() {
  Serial.begin(115200); //Baud Rate 115200
  base.attach(8);
  shoulder.attach(9);
  elbow.attach(10);
  wrist.attach(11);
  gripper.attach(12);
  base.write(base_angle);
  shoulder.write(shoulder_angle);
  elbow.write(elbow_angle);
  wrist.write(wrist_angle);
}
String getValue(String values, char sep, int i1)
{
  int found = 0;
  int stIndex[] = {0, -1};
  int largIndex = values.length()-1;
  for(int i=0; i<=largIndex && found<=i1; i++){
    if(values.charAt(i)==sep || i==largIndex){
      found++;
      stIndex[0] = stIndex[1]+1;
      stIndex[1] = (i == largIndex) ? i+1 : i;
    }
  }
  return found>i1 ? values.substring(stIndex[0], stIndex[1]) : "";
}
void loop(){

  String computerT = Serial.readStringUntil('@');
  computerT.trim();
```

```

if (computerT.length() == 0) {
    return;
}
String command = getValue(computerT, ' ',0);
if (command == "right" || command == "يمين" || command == "Right") {
    base.write(base_angle -= 30);
}
if (command == "left" || command == "Left" || command == "يسار") {
    base.write(base_angle += 30);
}
if (command == "top" || command == "فوق" || command == "Top") {
    shoulder.write(shoulder_angle -= 30);
}
if (command == "bottom" || command == "تحت" || command == "Bottom") {
    shoulder.write(shoulder_angle += 30);
}
Serial.println(command);
Serial.println("running");
delay(1000);
}

```

## Web serial api

```

let
isConnecttd
= false;

let port;
let writer;
var target_id;
const enc = new TextEncoder();
async function onChangespeech() {
    if (!isConnecttd) {
        alert("connect to the usb in order to use this.");
        return; }

    try {
        const comlist = content;
        const comSplit = comlist.split(" ")
        const command = comSplit.slice(-1);
        const computerT = `${command}@`;
        await writer.write(enc.encode(computerT));
    } catch (e) {
        console.log(e);
    }
}

```

```

    }
  }

  async function onConnectUsb() {
    try {
      const requestOptions = {
        // Filter on devices with the Arduino USB vendor ID.
        filters: [{ usbVendorId: 0x2341 }],
      };
      // Request an Arduino from the user.
      port = await navigator.serial.requestPort(requestOptions);
      await port.open({ baudRate: 115200 });
      writer = port.writable.getWriter();
      isConnected = true;
    } catch (e) {
      console.log("error", e);
    }
  }
}

```

## Html code

```

<!DOCTYPE
E html>

<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Speech to text</title>
  <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.mi
n.css">
</head>
<body background="image1.jpg">
  <div class="container">
    <h1 class="text-center mt-5">
      speech to text in Arabic language
    </h1>
    <div class="form-group">
      <textarea id="textbox" rows="6" class="form-
control"></textarea>
    </div>
    <div class="form-group">

```

```

        <button id="start-btn" class="btn btn-danger btn-block"
style="background-color:black ;">
            Start Recognition
        </button>
        <button id="pause-record-btn" class="btn btn-danger btn-
block" style="background-color:black;">
            Pause Recognition
        </button>
        <p id="instructions">
            Press the Start button </p>
    </div>
</div>
</body>
<script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"><
/script>
<script src="script.js"></script>
</html>

```

## Script code

```

var SpeechRecognition =
window.webkitSpeechRecognition

var recognition = new SpeechRecognition()
var textbox = $("#textbox")
var instructions = $("#instructions")
var content = ''
recognition.continuous = true
recognition.lang='ar';
recognition.onstart = function () {
instructions.text("Voive Recognition is on")
}
recognition.onspeechend = function () {
instructions.text("No Activity")
}
recognition.onerror = function (){
instructions.text("Try Again")
}
recognition.onresult= function (event){
var current = event.resultIndex;
var transcript=
event.results[current][0].transcript

```

```
content += transcript
textbox.val(content)
}
$("#start-btn").click(function(event) {
if (content.length){
content += ' '
}
recognition.start()
})
$('#pause-record-btn').click(function(event) {
recognition.stop();
instructions.text('Voice recognition paused');
});
textbox.on('input', function (){
content = $(this).val()
})
```