ADVANCE

AI MUSIC WEB APP- 4

INSTRUCTIONS:

Goal of the Project:

In class 129, you have achieved to control the volume of the song using your left wrist.

In this project you have to add JS code for fetching the score of the left wrist, drawing a circle on x and y coordinates of the left wrist, and start playing a song and stop the older song if the left wrist is detected.

**** This is a continuation of the project we did for Class 126, 127, and C128. Please complete those projects before attempting this project ****

Story:

Party Master is an event hosting company. This time they have got an offer to host an event at an old age home. So they are planning to have one of the old grandpa to be the DJ. Since an old grandpa will be playing the DJ they want it to be simple, where a grandpa can play music just by raising his hands. And they want someone to build this awesome web app for them.
This project is all about fetching the score of the left wrist, drawing a circle on x and y coordinates of the left wrist, and start playing a song and stop the older song if the left wrist is detected.

Getting Started:

Start JS coding in **main.js** file. You will find this file in the **AI-Music-Web-App** folder which you created in the project no. 126.

Specific Tasks to complete the Project:

1. Define a variable to store the score of the left wrist at the beginning of the main.js file, and set the value as 0.
2. Define a variable with an empty string value at the beginning of the main.js file. This variable will be used to hold the status of the song files which are supposed to be played when the left wrist is in front of the webcam. Status of the song file refers to whether or not the song is playing.
3. Inside **gotPoses()** function:
    - Write code for fetching the score of the left wrist, and assign the score of the left wrist to its respective variable which you have defined in point 1.

4. Inside p5.js **draw()** function:
    - **Add** code for setting the color and border color for the circle.
    - Remember you have loaded 2 songs in p5.js **preload()** function, so let's consider - playing song1 when the left wrist is shown in front of the webcam and playing song2 when the right wrist is shown in front of the webcam.
        - So first add code for getting the status of song1 (song1 will be played when left wrist is shown). To get the status of any song p5.js has a function - **isPlaying()** function.
            - Syntax of **isPlaying()** function - **song_variable.isPlaying()** - this will give either **true**(song is playing) or **false**(song is stopped).

        - **Add** code for getting the status of song1(song1 will be played when left wrist is shown) and store in a variable which you have defined in point 2.
        - **Add** a "if condition" to check that if the score of the left wrist is greater than 0.2 then.
        - Inside this "if condition":
            - **Add** code for drawing a circle on x and y coordinates of the left wrist.
            - Stop song2 playing(song2 will be played when the right wrist is shown). To stop a song p5.js has a function **stop()** function. Syntax of **song_variable.stop().**
        - **Add** a "if condition" to check that the variable which is holding the status of song1 is equal to false(song is stopped).
            - If yes then write code for playing song1(song1 will be played when left wrist is shown).
            - Remember in project no.126 you had defined a heading tag with an id as "song", and the purpose of this heading tag was to hold the song name. So **Add** code for updating this heading tag with the name of song1 (song1 will be played when left wrist is shown).

Note: For a better undertaking of how the code will look inside the **draw()** function, an image is provided in the **HINT** section.

Submitting the Project:

IF you have not created a repository in the previous project and this is the first time you are hosting an AI Music Web App project on GitHub then:

1. Upload all the files which you have created in the **AI-Music-Web-App** folder on GitHub.
2. You can get the steps to do this by clicking on this link.

3. Copy the hosted link which you will get after uploading all your files on GitHub and submit it in the Student Dashboard Projects panel against the correct class number.

OR

IF you have already hosted AI Music Web App project on GitHub then:

1. Upload the 2 music files which you have downloaded on GitHub.
2. Reupload **main.js** file on GitHub, this is done because you have made changes in this file.
3. You can get the steps to do this by clicking on this link.
4. Copy the hosted link which you will get after uploading all your files on GitHub and submit it in the Student Dashboard Projects panel against the correct class number.

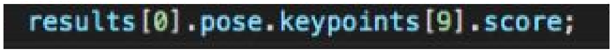
DO NOT TRY TO HOST THIS ON DRIVE TO WEB BECAUSE WE WILL BE USING AUDIO FILES IN THIS PROJECT AND DRIVE TO WEB WON'T SUPPORT

ADVANCE

AI MUSIC WEB APP- 4

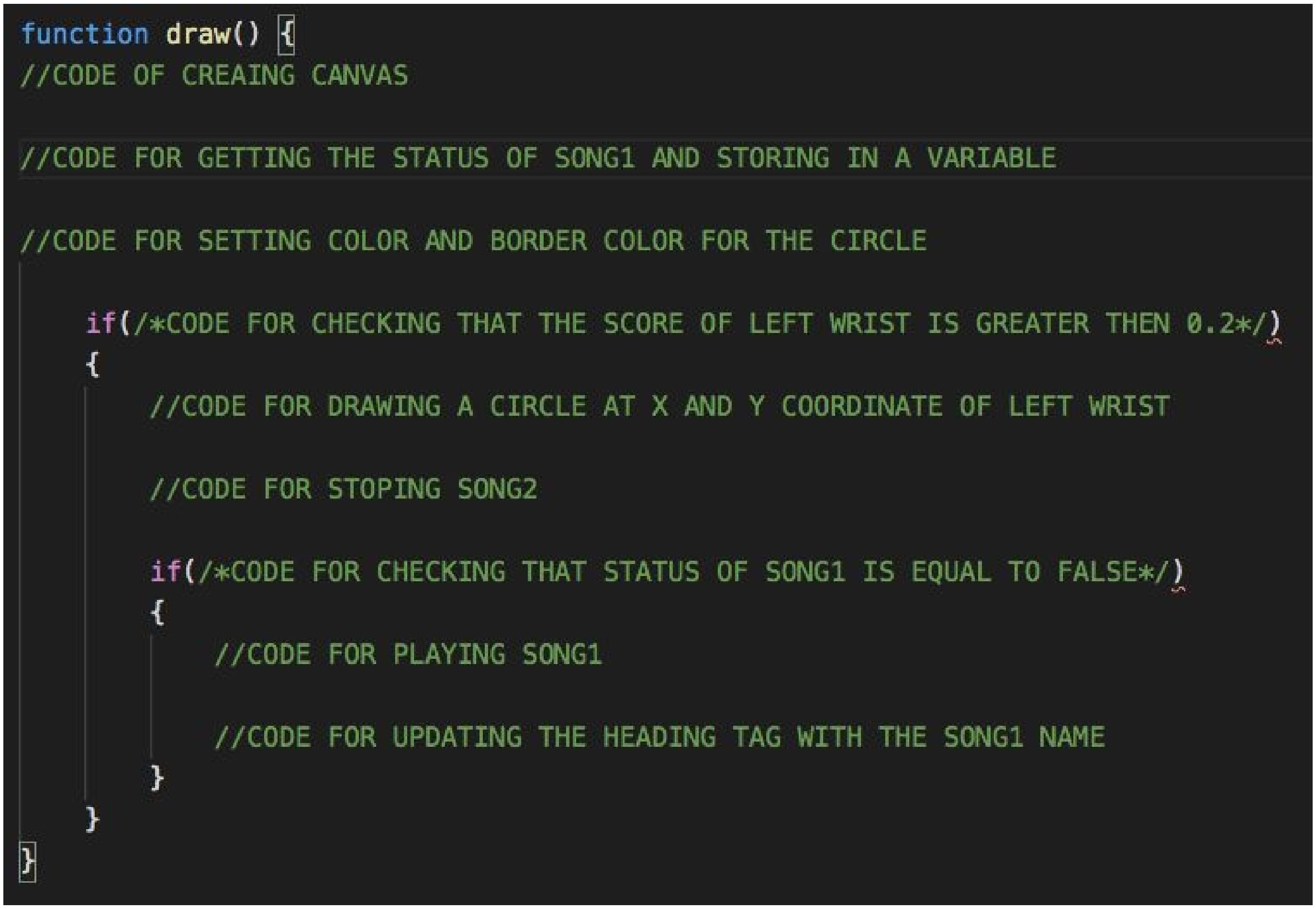
Hints:

1. Code for fetching the score of the left wrist.



```
results[0].pose.keypoints[9].score;
```

2. For a better understanding of how the code will look like inside **draw()** function, an image is provided below.



```
function draw() {
//CODE OF CREAING CANVAS

//CODE FOR GETTING THE STATUS OF SONG1 AND STORING IN A VARIABLE

//CODE FOR SETTING COLOR AND BORDER COLOR FOR THE CIRCLE

    if(/*CODE FOR CHECKING THAT THE SCORE OF LEFT WRIST IS GREATER THEN 0.2*/)
    {
        //CODE FOR DRAWING A CIRCLE AT X AND Y COORDINATE OF LEFT WRIST

        //CODE FOR STOPING SONG2

        if(/*CODE FOR CHECKING THAT STATUS OF SONG1 IS EQUAL TO FALSE*/)
        {
            //CODE FOR PLAYING SONG1

            //CODE FOR UPDATING THE HEADING TAG WITH THE SONG1 NAME
        }
    }
}
```

REMEMBER... Try your best, that's more important than being correct.
After submitting your project your teacher will send you feedback on your work.

———— xxx ———— xxx ———— xxx ———— xxx ———— xxx ————