<h1>Speech color changer</h1>

<p>Tap/click then say a color to change the background color of the app.</p>

<div>

<p class="output"><em>...diagnostic messages</em></p>

</div>

const SpeechRecognition = window.SpeechRecognition || webkitSpeechRecognition;

const SpeechGrammarList = window.SpeechGrammarList || webkitSpeechGrammarList;

const SpeechRecognitionEvent = window.SpeechRecognitionEvent || webkitSpeechRecognitionEvent;

const colors = [ 'aqua' , 'azure' , 'beige', 'bisque', 'black', 'blue', 'brown', 'chocolate', 'coral' ... ];

const grammar = '#JSGF V1.0; grammar colors; public <color> = ' + colors.join(' | ') + ' ;'

const recognition = new SpeechRecognition();

const speechRecognitionList = new SpeechGrammarList();

speechRecognitionList.addFromString(grammar, 1);

recognition.grammars = speechRecognitionList;

recognition.continuous = false;

recognition.lang = 'ar-KSA';

recognition.interimResults = false;

recognition.maxAlternatives = 1;

const diagnostic = document.querySelector('.output');

const bg = document.querySelector('html');

const hints = document.querySelector('.hints');

let colorHTML= '';

colors.forEach(function(v, i, a){

console.log(v, i);

colorHTML += '<span style="background-color:' + v + ';"> ' + v + ' </span>';

});

hints.innerHTML = 'Tap/click then say a color to change the background color of the app. Try ' + colorHTML + '.';

document.body.onclick = function() {

recognition.start();

console.log('Ready to receive a color command.');

}

recognition.onresult = function(event) {

let color = event.results[0][0].transcript;

diagnostic.textContent = 'Result received: ' + color + '.';

bg.style.backgroundColor = color;

console.log('Confidence: ' + event.results[0][0].confidence);

}

recognition.onspeechend = function() {

recognition.stop();

}

recognition.onnomatch = function(event) {

diagnostic.textContent = 'I didn\'t recognize that color.';

}

recognition.onerror = function(event) {

diagnostic.textContent = 'Error occurred in recognition: ' + event.error;

}

<h1>Speech synthesizer</h1>

<p>Enter some text in the input below and press return to hear it. change voices using the dropdown menu.</p>

<form>

<input type="text" class="txt">

<div>

<label for="rate">Rate</label><input type="range" min="0.5" max="2" value="1" step="0.1" id="rate">

<div class="rate-value">1</div>

<div class="clearfix"></div>

</div>

<div>

<label for="pitch">Pitch</label><input type="range" min="0" max="2" value="1" step="0.1" id="pitch">

<div class="pitch-value">1</div>

<div class="clearfix"></div>

</div>

<select>

</select>

</form>

let synth = window.speechSynthesis;

const inputForm = document.querySelector('form');

const inputTxt = document.querySelector('.txt');

const voiceSelect = document.querySelector('select');

const pitch = document.querySelector('#pitch');

const pitchValue = document.querySelector('.pitch-value');

const rate = document.querySelector('#rate');

const rateValue = document.querySelector('.rate-value');

const voices = [];

function populateVoiceList() {

voices = synth.getVoices();

for(i = 0; i < voices.length ; i++) {

const option = document.createElement('option');

option.textContent = voices[i].name + ' (' + voices[i].lang + ')';

if(voices[i].default) {

option.textContent += ' — DEFAULT';

}

option.setAttribute('data-lang', voices[i].lang);

option.setAttribute('data-name', voices[i].name);

voiceSelect.appendChild(option);

}

}

populateVoiceList();

if (speechSynthesis.onvoiceschanged !== undefined) {

speechSynthesis.onvoiceschanged = populateVoiceList;

}

inputForm.onsubmit = function(event) {

event.preventDefault();

const utterThis = new SpeechSynthesisUtterance(inputTxt.value);

const selectedOption = voiceSelect.selectedOptions[0].getAttribute('data-name');

for(i = 0; i < voices.length ; i++) {

if(voices[i].name === selectedOption) {

utterThis.voice = voices[i];

}

}

utterThis.pitch = pitch.value;

utterThis.rate = rate.value;

synth.speak(utterThis);

utterThis.onpause = function(event) {

const char = event.utterance.text.charAt(event.charIndex);

console.log('Speech paused at character ' + event.charIndex + ' of "' +

event.utterance.text + '", which is "' + char + '".');

}

inputTxt.blur();

}

pitch.onchange = function() {

pitchValue.textContent = pitch.value;

}

rate.onchange = function() {

rateValue.textContent = rate.value;

}

**By:Rahaf Mohmmad Almohammadi**