113年智慧行動商務系統開發班第1期

網頁前端程式設計與開發學習評量

|  |  |  |  |
| --- | --- | --- | --- |
| **評量編號** | 2-4 | | |
| **評量主題** | 前端程式設計綜合應用 | **實施時間** | 2025/5/21 |
| **評量形式** | 作業紀錄 | **評量日期** | 2025/6/13 |
| **評量目的** | 評定學員是否具備網頁前端程式設計能力，包含撰寫HTML、CSS與JavaScript程式碼並使其相互搭配運用，並可以Function架構進行程式基本解耦之實作。 | | |
| **評量說明** | **任務一**：請利用前端(Front-end)程式設計方法包含HTML、CSS與JavaScript或其他您熟知的方法，撰寫題目所指定的網頁軟體，具備題目所示的功能及適切的介面，並可於直接瀏覽器執行。  **任務二**：程式碼必須有基本的結構化設計，包括將HTML、CSS與JavaScript分開、函數化、參數化等降低耦合力的撰寫技巧。 | | |
| **提交要求** | 繳交時請以FTP上傳**相關程式檔案**與**作業文件檔**各一份，**相關程式檔案**請另放置在「完成結果檔」資料夾內，最外層資料夾名稱請用**兩碼學號與姓名**，例如1號王小明，資料夾請命名為「01王小明」。 | | |

座號：05 姓名：鄭冠怡

**請利用前端(Front-end)程式設計方法包含HTML、CSS與JavaScript，撰寫一網頁音樂播放器軟體，並可於直接瀏覽器執行，須達到以下要求：**

1. 須有可按鈕執行播放、停止與暫停的功能。

2. 須有可按鈕執行上一首的下一首的功能。

3. 須有直接選擇曲目的功能。

4. 須有可按鈕執行快轉及倒轉的功能。

5. 須有可按鈕執行單曲循環、全部循環及隨機選曲播放的功能。

6. 須有音量控制的功能。

7. 須有資訊看版可顯示目前播放狀態及目前播放曲目的功能。

8. 須有播放進度條，並可由進度條調整曲目快轉或倒轉的功能。

10. 播放器介面上須放置本分署Logo，位置不限。

11. 介面不限，可自行設計，但須符合一般操作原則(UI/UX)。

12. CSS及JavaScript檔必須獨立在HTML檔之外，再引入使用。

13. 程式碼必須有基本的結構化設計，包括函數化、參數化等技巧。

進度條

介面上要有Logo

直接選曲功能

各功能按鈕





音量控制

音量控制

資訊看版

播放狀態呈現

**完成參考範例**

程式原始碼請貼於此處下方(若有好幾個檔案請註明檔名)：

HTML：

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>05\_MusicPlayer</title>

<link rel="stylesheet" href="player.CSS" />

</head>

<body>

<audio src="music/tabun.mp3" title="【たぶん-YOASOBI】" controls id="myMusic" onended="musicStatus()"></audio>

<hr>

<div id="controlPanel">

<select id="musicList" onchange="changeMusic(0)">

<option value="music/tabun.mp3">♦ たぶん-YOASOBI</option>

<option value="music/hakujitsu.mp3">♦ 白日-King Gnu</option>

<option value="music/sayonaramataistuka.mp3">♦ さよーならまたいつか！-米津玄師</option>

<option value="music/akumanoko.mp3">♦ 悪魔の子-ヒグチアイ</option>

<option value="music/basanka.mp3">♦ 晩餐歌ばさんかん-tuki.</option>

</select>

<img src="logo.png" id="logoImage" alt="分署logo" title="分署logo">

<div id="functionButtons">

<span onclick="playMusic()" title="播放/暫停">4</span>

<!-- <span onclick="pauseMusic()">;</span> -->

<span onclick="stopMusic()" title="停止">&lt;</span> <!-- 【&lt;】代換"<""-->

<span onclick="changeMusic(-1)" title="上一首">9</span>

<span onclick="changeTime(-5)" title="倒轉">7</span>

<span onclick="changeTime(+5)" title="快轉">8</span>

<span onclick="changeMusic(1)" title="下一首">:</span>

<span onclick="setMute()" title="靜音/取消靜音">U</span>

<span onclick="setMode('單曲循環')" title="單曲循環">q</span>

<span onclick="setMode('隨機播放')" title="隨機播放">s</span>

<span onclick="setMode('全曲循環')" title="全曲循環">`</span>

</div>

<div id="volumeControl">

<input type="range" oninput="setVolumeByRangeBar()" value="50"><!-- 預設音量為50-->

<button onclick="changeVolume(-1)">-</button>

<button onclick="changeVolume(1)">+</button>

<input type="text" value="100" readonly>

</div>

<input type="range" id="progressBar" min="0" value="0" oninput="setProgress()">

<div id="information">

<div>00:00 / 00:00</div>

<div>目前狀態</div>

<div>播放模式</div>

</div>

</div>

<script src="player.js"></script>

</body>

</html>

CSS：

:root {

font-family: 'Courier New ', Courier, monospace;

--vol-thumb-appearance: none;

--vol-thumb-width: 12px;

--vol-thumb-height: 16px;

--vol-thumb-background-color: #b5724d;

--vol-thumb-border-radius: 3px;

--vol-thumb-border: 2px solid #4b3d36;

--progress-appearance: none;

--progress-width: 12px;

--progress-height: 12px;

--progress-background-color: #4b3d36;

--progress-border: 1px solid #bfa76a;

--progress-border-radius: 0;

--progress-transition: 0.3s;

--progress-hover-width: 14px;

--progress-hover-height: 14px;

--progress-hover-background-color: #bfa76a;

}

body {

background-color: #808074;

color: #d9c8b4;

}

#musicList {

width: 400px;

height: 45px;

appearance: none;

background-color: #f2f2f2;

color: #2e2e2e;

border: 3px solid #a89373;

border-radius: 12px;

font-size: 14px;

padding-left: 10px;

cursor: pointer;

margin-bottom: 10px;

background-image: url('https://cdn-icons-png.flaticon.com/128/563/563541.png');

/\* 會出現GET http://127.0.0.1:5501/favicon.ico 404 (Not Found)的'提示' ,因為瀏覽器在資料夾找不到檔案

不影響功能\*/

background-repeat: no-repeat;

background-position: right 10px center;

background-size: 18px 18px;

}

#musicList:hover {

background-color: #e0d4c1;

}

#logoImage {

width: 300px;

height: 300px;

margin: 5px auto;

margin-bottom: 30px;

display: flex;

filter: blur(3px);

transition: filter 0.3s;

}

@keyframes rotateLogo {

100% {

transform: rotate(360deg);

}

}

#logoImage.playing {

animation: rotateLogo 5s linear infinite;

filter: blur(0);

}

#logoImage.playing:hover {

animation-play-state: paused;

}

#logoImage:hover {

filter: blur(0);

}

#controlPanel .setMute {

text-decoration: line-through;

background-color: #3a2f2b;

}

#controlPanel {

border: 4px solid #6a5a4d;

background-color: #3b3732;

border-radius: 10px;

width: 600px;

padding: 40px;

margin: 10px auto;

box-shadow: 0px 0px 20px #1a1a1a;

justify-content: center

}

#controlPanel span {

min-width: 10px;

background-color: #8d7d71;

border-radius: 6px;

border-bottom: 4px solid #40362c;

text-align: center;

box-shadow: 0px 4px 10px #1e1e1e;

transition: all 0.2s;

padding: 7px 15px;

cursor: pointer;

color: #f2e9da;

font-family: webdings;

margin: 10px 2px;

}

#controlPanel span:hover {

box-shadow: 0px 12px 18px -5px #aa8453;

transform: scale(1.02);

}

#controlPanel span:active {

box-shadow: 0px 4px 8px #eee6dd;

transform: scale(0.96);

}

#volumeControl {

display: flex;

align-items: center;

gap: 10px;

margin-top: 20px;

}

#volumeControl>input[type="text"] {

width: 50px;

height: 25px;

border: 2px solid #917a5a;

background-color: #1f1e1c;

color: #f2e9da;

text-align: center;

font-weight: bold;

border-radius: 6px

}

#volumeControl>input[type="range"] {

width: 300px;

height: 8px;

appearance: none;

background-color: #4d4439;

border: 2px solid #7a6857;

border-radius: 6px;

}

#volumeControl>input[type="range"]::-webkit-slider-thumb {

appearance: var(--vol-thumb-appearance);

width: var(--vol-thumb-width);

height: var(--vol-thumb-height);

background-color: var(--vol-thumb-background-color);

border-radius: var(--vol-thumb-border-radius);

border: var(--vol-thumb-border);

}

#volumeControl>input[type="range"]::-moz-range-thumb {

appearance: var(--vol-thumb-appearance);

width: var(--vol-thumb-width);

height: var(--vol-thumb-height);

background-color: var(--vol-thumb-background-color);

border-radius: var(--vol-thumb-border-radius);

border: var(--vol-thumb-border);

}

#volumeControl>button {

border-radius: 6px;

border: 2px solid #715b47;

background-color: #3c3228;

color: #e2d2bd;

width: 30px;

height: 30px;

}

#volumeControl>button:hover {

background-color: #715b47;

color: #ffffff;

transform: scale(1.5);

}

#information>div {

height: 40px;

color: #1f1e1c;

font-size: 1.2rem;

text-align: center;

line-height: 40px;

background-color: #f3e2ce;

padding: 0px 10px;

margin-top: 15px;

border: 1px solid #f8d2aa;

border-radius: 5px;

overflow: auto;

}

#progressBar {

appearance: none;

width: 100%;

height: 10px;

background-color: #f3efea;

margin: 0px 5px;

margin-top: 20px;

border-radius: 6px;

}

#progressBar::-webkit-slider-thumb {

appearance: var(--progress-appearance);

background-color: var(--progress-background-color);

width: var(--progress-width);

height: var(--progress-height);

border-radius: var(--progress-border-radius);

border: var(--progress-border);

}

#progressBar::-webkit-slider-thumb:hover {

width: var(--progress-hover-width);

height: var(--progress-hover-height);

transition: var(--progress-transition);

background-color: var(--progress-hover-background-color);

}

#progressBar::-moz-range-thumb {

appearance: var(--progress-appearance);

background-color: var(--progress-background-color);

width: var(--progress-width);

height: var(--progress-height);

border-radius: var(--progress-border-radius);

border: var(--progress-border);

}

#progressBar::-moz-range-thumb:hover {

width: var(--progress-hover-width);

height: var(--progress-hover-height);

transition: var(--progress-transition);

background-color: var(--progress-hover-background-color);

}

/\* RWD \*/

@media (max-width: 600px) {

#controlPanel {

width: 98%;

padding: 10px;

}

#controlPanel span {

min-width: 20px;

padding: 4px 6px;

font-size: 1rem;

}

#musicList {

width: 96%;

}

}

JavaScript：

//功能區

var myMusic = document.getElementById("myMusic");

var volumeControl = document.getElementById("volumeControl");

var information = document.getElementById("information");

var progressBar = document.getElementById("progressBar");

var musicList = document.getElementById("musicList")

var functionButtons = document.getElementById("functionButtons");

var txtVolume = volumeControl.children[3]; //音量顯示txt

var rangeVolume = volumeControl.children[0]; //音量調整range

var musicDuration = information.children[0];

var playStatus = information.children[1];

var btnPlay = functionButtons.children[0]; //播放鈕

var infoStatus = information.children[2]; //單曲循環鈕

/////////////////////////////////////////

function musicStatus() {

if (infoStatus.innerText == "單曲循環") {

changeMusic(0);

}

else if (infoStatus.innerText == "隨機播放") {

var n = Math.floor(Math.random() \* musicList.length);

changeMusic(n - musicList.selectedIndex);

}

else if (infoStatus.innerText == "全曲循環" && musicList.length == musicList.selectedIndex + 1) {

changeMusic(0 - musicList.selectedIndex);

}

else if (musicList.length == musicList.selectedIndex + 1) { //是否為最後一首歌

stopMusic();

}

else { //不是最後一首歌就播下一首歌

changeMusic(1);

}

}

function setMode(mode) {

infoStatus.innerHTML = infoStatus.innerHTML == mode ? "正常" : mode;

} //播放模式做互斥

function changeMusic(n) {

var i = musicList.selectedIndex;//選擇的音樂索引

//console.log(i + n);

myMusic.src = musicList.children[i + n].value; //更改音樂來源

/\* 當 i + n 超過 musicList 的子元素數量（我目前最大索引值是4(5首歌)），musicList.children[i + n] 就會出現 undefined \*/

myMusic.title = musicList.children[i + n].innerText;

musicList.children[i + n].selected = true;//選擇的音樂索引

//console.log(btnPlay.innerText);

if (btnPlay.innerText == ";") {

myMusic.onloadeddata = playMusic; //音樂載入完成後，再開始播放音樂

}

}

//時間格式

function getTimeFormat(t) {

var min = parseInt(t.toFixed(0) / 60);

var sec = parseInt(t.toFixed(0) % 60);

min = min < 10 ? "0" + min : min;

sec = sec < 10 ? "0" + sec : sec;

return min + ":" + sec;

}

function setProgress() {

myMusic.currentTime = progressBar.value / 10000; //更新音樂的當前時間

}

//音樂播放時間

function setMusicDuration() {

// 更新時間顯示

musicDuration.innerHTML = getTimeFormat(myMusic.currentTime) + "/" + getTimeFormat(myMusic.duration);

// 更新進度條的值

progressBar.value = myMusic.currentTime \* 10000;

// 計算進度百分比

var w = myMusic.currentTime / myMusic.duration \* 100;

// 更新進度條的背景漸層

progressBar.style.backgroundImage = `linear-gradient(to right, rgb(143, 114, 81) ${w}%,rgb(236, 236, 234) ${w}%)`;

}

/////////////////////////////////////////////////////////////

//目前歌曲長度初始化

function ProgressInitial() {

progressBar.max = myMusic.duration \* 10000; // 設置進度條的最大值為音樂的總時長

setInterval(setMusicDuration, 5);// 每秒更新一次進度條

}

setVolumeByRangeBar(); //初始化音量

//音量調整

function setVolumeByRangeBar() {

//console.log(event.target.value)

txtVolume.value = rangeVolume.value;

myMusic.volume = txtVolume.value / 100;

rangeVolume.style.backgroundImage = `linear-gradient(to right, rgba(252, 160, 55, 0.8) ${rangeVolume.value}%,rgb(255, 255, 255) ${rangeVolume.value}%)`;

}

//音量調整

function changeVolume(v) {

rangeVolume.value = parseInt(rangeVolume.value) + v;

setVolumeByRangeBar(); //呼叫音量調整的function

}

//靜音

function setMute() {

myMusic.muted = !myMusic.muted;

event.target.className = event.target.className == "setMute" ? "" : "setMute";

updateInfo(myMusic.muted ? "已靜音" : "取消靜音");

setTimeout(() => {

updateInfo("目前播放" + myMusic.title);

}, 1000);

}

//快轉倒轉

function changeTime(s) {

myMusic.currentTime += s;

}

function updateInfo(text) {

playStatus.innerHTML = text;

}

//播放音樂

function playMusic() {

//console.log(event.target);

myMusic.play();

event.target.innerHTML = ";";

event.target.onclick = pauseMusic;

ProgressInitial(); //音樂開始播放時，才開始更新進度條的值

updateInfo("目前播放" + myMusic.title);

document.getElementById('logoImage').classList.add('playing');

}

//暫停音樂

function pauseMusic() {

myMusic.pause();

event.target.innerHTML = "4";

event.target.onclick = playMusic;

updateInfo("音樂暫停");

document.getElementById('logoImage').classList.remove('playing');

}

//停止音樂

function stopMusic() {

myMusic.pause();

myMusic.currentTime = 0;

updateInfo("音樂停止");

btnPlay.innerText = "4";

btnPlay.onclick = playMusic;

document.getElementById('logoImage').classList.remove('playing');

}

畫面截圖請貼於此處下方(請截取執行不同功能時的圖至少3張)：

