

IPI International

Working with IndexedDB

Name : IPI Media Player Indexed Db
Type : HTML
Designer : Paul I Ighofose
Date Created : 21/09/2017
Date Updated : 20/10/2017
Requirements : Google Chrome, Internet Explorer, Microsoft Edge, Firefox, Windows Phone IE, IndexedDB enabled browser
Function : IndexedDB Database and Media Player

Introduction

Determined to achieve data storage functionality with my original Media Player App, I am somewhat happy to have embraced the IndexedDB environment. Utilization of the IndexedDB API I have found that some asynchronous quirks can be overcome. Although I prefer the use of Web SQL's syntax, the speed and storage size makes up for not being able to offer the user more results control without adding an encyclopaedia of scripts to determine the right response to their requests.

Package Structure

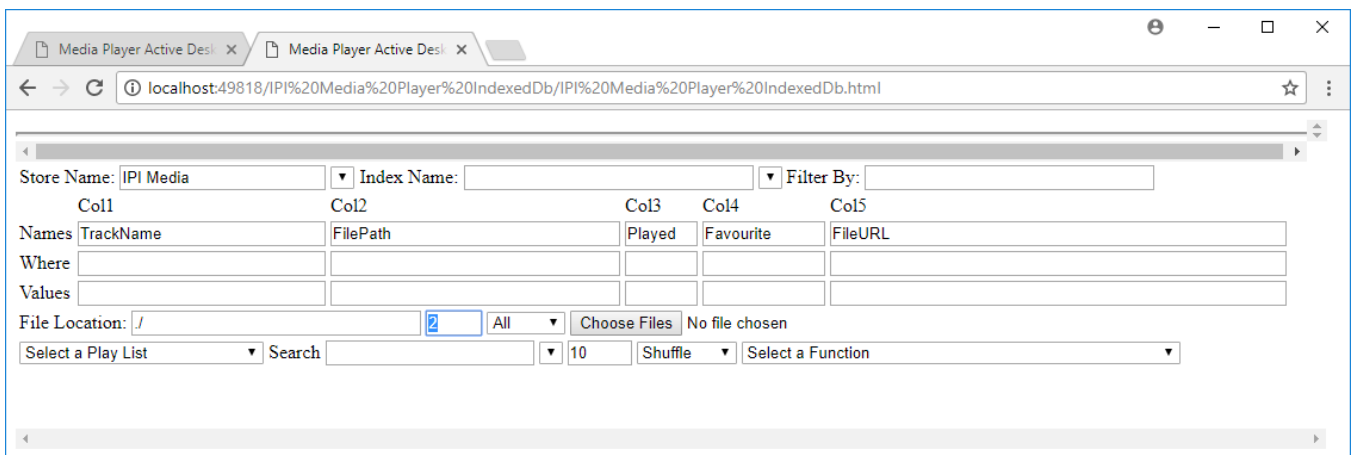
The main documents needed to run this package are all included in the ZIP file. The HTML file is encoded as UTF 8 and the HTM as Windows 1252. The additional files Radio Db.htm and YouTube Db.htm can be used to load sample entries into the default store IPI Media.

Operational Guide

Save either the HTM or HTML file to either of the following:

- Disk drive / Flash disk containing media that is available over Bluetooth or Wi-Fi networks
- Windows Phone 8.1 App with IE Browser functions
- Intranet or Internet folder

1. Open the file using either Google Chrome or your device browser. I have added a wait input as the time to return all records increases with larger record sets and the sort function needs to wait before acting



2. The default **Store Name** is IPI Media. Using different Store Names will allow you to split your data into smaller data sets should you require speedier results store updates. It can also act as creating favourite lists. Only file URL/URI links are stored. When no store has been created or all stores have been dropped the **selector** next to the store name will have no entries in it. Any existing store can be selected in the selector which in turn will update the text of the store name.

3. A separate script has been written for Internet Explorer as it does not have the ability to use dynamic entries in the creation of field names and is Fixed to only create stores with the following fields:
 - a. Id
 - b. TrackName
 - c. FilePath
 - d. Played
 - e. Favourite
 - f. FileURL
4. For all other browsers except Internet Explorer you can specify the field names on the **Names** line. Id is hard coded in and is auto-incremented. Select the **Create Store** option from the Select a Function selector

The screenshot shows a web application interface for managing a database. At the top, there are input fields for 'Store Name' (containing 'IPI Media'), 'Index Name', and 'Filter By'. Below these are five columns labeled Col1 through Col5. The 'Names' row contains 'TrackName', 'FilePath', 'Played', 'Favourite', and 'FileURL'. The 'Where' and 'Values' rows are empty. Below the columns, there is a 'File Location' field with a dropdown set to 'All' and a 'Choose Files' button. A 'Select a Play List' dropdown is followed by a 'Search' field and a 'Shuffle' button. The 'Select a Function' dropdown menu is open, showing options: 'Select a Function', 'Delete Database', 'Create Store' (highlighted in blue), 'Delete Store', and 'View/Refresh Results Store'.

5. As you can see, the default columns are present in the input boxes for the line beginning with **Names**. Making these editable allows you more choice on the design of your stores.
6. Because IndexedDB needs a database version change in the creation; deletion; and altering of stores and indexes I have included a **Delete Database** function. After deleting the database ensure to Refresh the web page and creating a new store will create a database as Version 1
7. Use the input boxes on the line beginning with **Values** to enter your row entries followed by selecting the **Insert Row** option from the Select a Function selector. Also note below, that you can use the results of the populated store to create insert statements for a Web SQL version or just a copy table of the results which IndexedDB can use to populate the Store Name given in the Store Name field (change it to a store name of choice before running the insert function). Creating as HTML will include Table, TR and TD element tags around each statement so that it can be copied and saved as a HTML document that can be accessed by other devices to populate their own Web SQL databases. By changing the **Store Name** before creation will populate the insert statements with the Name currently displayed. Hence creating the inserts as a Table will allow you to upload you filtered/non filtered results in to another Web SQL Store. There was no need to create fancy insert statements for IndexedDB due to trying to accommodate using dynamically driven Column Names and the variable input syntax. So it just uses a copy of the results table placed inside the iFrame

This screenshot shows the same application interface as the previous one, but with a different function selected. The 'Select a Function' dropdown menu is open, and 'Create Upload Table from Selected' is highlighted in blue. Other visible options include 'Create Upload Table from All', 'Create WebSQL Insert Transaction from Results as HTML', and 'Create WebSQL Insert Transaction from Results as Table'.

8. To refresh or view the results store select the **View/Refresh Results Store** option. The results displayed can be filtered using the **Where** line input boxes and is ordered by **TrackName** and filter at row level. You specify which Index to retrieve results from by using the selector next to the **Index Name** input. You have only 2 options in the **Filter By** input which filters at Index level

The screenshot shows the application interface with the following fields and controls:

- Store Name:** IPI Media
- Index Name:** (empty)
- Filter By:** (empty)
- Col1:** TrackName
- Col2:** FilePath
- Col3:** Played
- Col4:** Favourite
- Col5:** FileURL
- Where:** (empty)
- Values:** (empty)
- File Location:** /
- All:** (checked)
- Choose Files:** No file chosen
- Select a Play List:** (empty)
- Search:** (empty)
- 10:** (empty)
- Shuffle:** (empty)
- Select a Function:** View/Refresh Results Store

- Use Whole values. This will look for exact matches in the given index. If nothing is returned it will then look for matches where that filter text can be found within the returned values. Separating multiple entries with a comma (,) will enable a search criteria where value is ??? or ???
 - Use partial values to force the exact match function to return no results and therefor move on to the partial search function. Again separate multiple values with a comma
9. Mistakes happen! The View/Refresh function add an extra column on the end to allow you to select particular lines. The **Delete Row** option will delete all selected lines. The **Delete All** will just delete all lines displayed in the results table. The **Reset Played** like above will only affect the filtered records setting their Played Value to 0. The **Replace Text** will be restricted to the current result set but will prompt you for 1 error, 1 replacement and the column number to carry out those replacements in.

The screenshot shows the application interface with the following fields and controls:

- Store Name:** IPI Media
- Index Name:** (empty)
- Filter By:** (empty)
- Col1:** TrackName
- Col2:** FilePath
- Col3:** Played
- Col4:** Favourite
- Col5:** FileURL
- Where:** (empty)
- Values:** (empty)
- File Location:** /
- All:** (checked)
- Choose Files:** No file chosen
- Select a Play List:** (empty)
- Search:** (empty)
- 10:** (empty)
- Shuffle:** (empty)
- Select a Function:** Delete Row

TrackName	FilePath	Played	
- Cecelia.mp3	c:\users\paul\music\	0	L
- Charlie Says.mp3	c:\users\paul\music\	0	L
- Cookie Jar.mp3	c:\users\paul\music\	0	L
1 Mix House	http://www.internetradiouk.c	0	R

10. The iFrame is used to display web pages, the video element and reformatted stores to use in uploads. You can display it by selecting the **Show/Hide iFrame** option. The **Copy iFrame** option will provide a prompt containing the inner text of the iFrame as a single line string. This sometimes avoid resettling with IOS's rejigging display. IndexedDB did not function as expected when trying to upload tables using formatted multiline web pages containing tables, whereas this issue was not experience in IOS using Web SQL.

The screenshot shows the application interface with the following fields and controls:

- Store Name:** IPI Media
- Index Name:** (empty)
- Filter By:** (empty)
- Col1:** TrackName
- Col2:** FilePath
- Col3:** Played
- Col4:** Favourite
- Col5:** FileURL
- Where:** (empty)
- Values:** (empty)
- File Location:** /
- All:** (checked)
- Choose Files:** No file chosen
- Select a Play List:** (empty)
- Search:** (empty)
- 10:** (empty)
- Shuffle:** (empty)
- Select a Function:** Show/Hide iFrame

11. Get Document Location, Get iFrame Location and Get iFrame Video Source will populate the FilePath value input. Copying the returned value of the Get Document Location function in to the File Location will then cause the iFrame to load the parent page in to the iFrame. Couple this with the Upload from Database Results in iFrame option will allow you to drop a store currently displayed in both the parent and iFrame. Change column structure renaming or adding columns then load back the results being held in the iFrame (no need for temp db). The **Make Results Table Full Page** will ensure that only the results table is displayed in the iFrame for use in self Uploads

The screenshot shows a web application interface with a table of tracks and a dropdown menu. The table has columns: TrackName, FilePath, Played, Favourite, and FileURL. The dropdown menu is open, showing a list of functions including 'Make Results Table Full Page', 'Get Document Location', 'Copy iFrame', 'Get iFrame Location', 'Get iFrame Video Source', 'Clear Editor', 'Run Editor As JavaScript', 'Create iFrame Table Element from Editor', 'Create iFrame Table Element from iFiles Display', 'Create iFrame Table Element from Mobilelite Display', 'Upload iFrame Table Element', and 'Upload from Database Results in iFrame'.

Col1	Col2	Col3	Col4	Col5
Names	TrackName	FilePath	Played	Favourite
Where	eagle			
Values				

File Location: / All Choose Files No file chosen

Select a Play List Search 10 Shuffle Select a Function

TrackName	FilePath	Played
The Eagle	http://www.964eagle.co.uk/radioplayer/	0
The Eagle	https://www.964eagle.co.uk/listenatwork/	0

Store Name: IPI Media Index Name: Filter By:

Col1	Col2	Col3	Col4
Names	TrackName	FilePath	Played
Where			
Values			

File Location: http://localhost:49818/IPI%20Media% All Choose Files No file chosen

Select a Play List Search 10 Shuffle S

TrackName	FilePath	Played
- Cecelia.mp3	c:\users\paul\music\	0

Local Cecelia.mp3

12. Overcoming browser restrictions means copying the results of a web page display of files on the web server displayed in the iFrame to the Editor Text Area then selecting the **Create iFrame Table Element from Editor** (selecting it twice as the 1st time fails due to browser restrictions). You can make the Editor visible by selecting the **Show/Hide Editor** option. Then after creating the reformatted store element in the iFrame select the **Upload iFrame Table Element** to load the displayed rows in to the database. If using the iFiles App on your iPhone or iPad there is no need to copy the results to the editor as the **Create iFrame Table Element from iFiles Display** will do it all. The same goes for the results given by a Mobilelite Dongle file listing using the **Create Table Element from Mobilelite Display**.

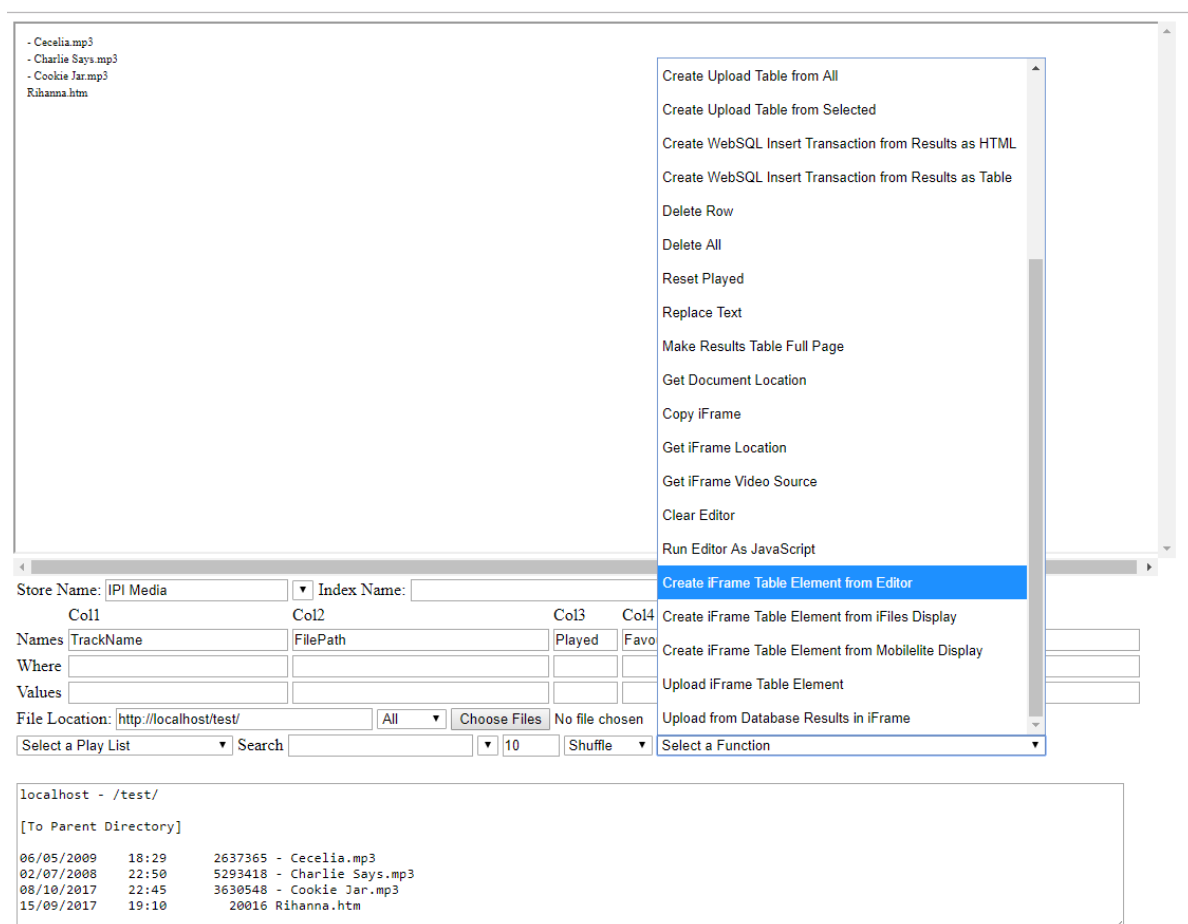
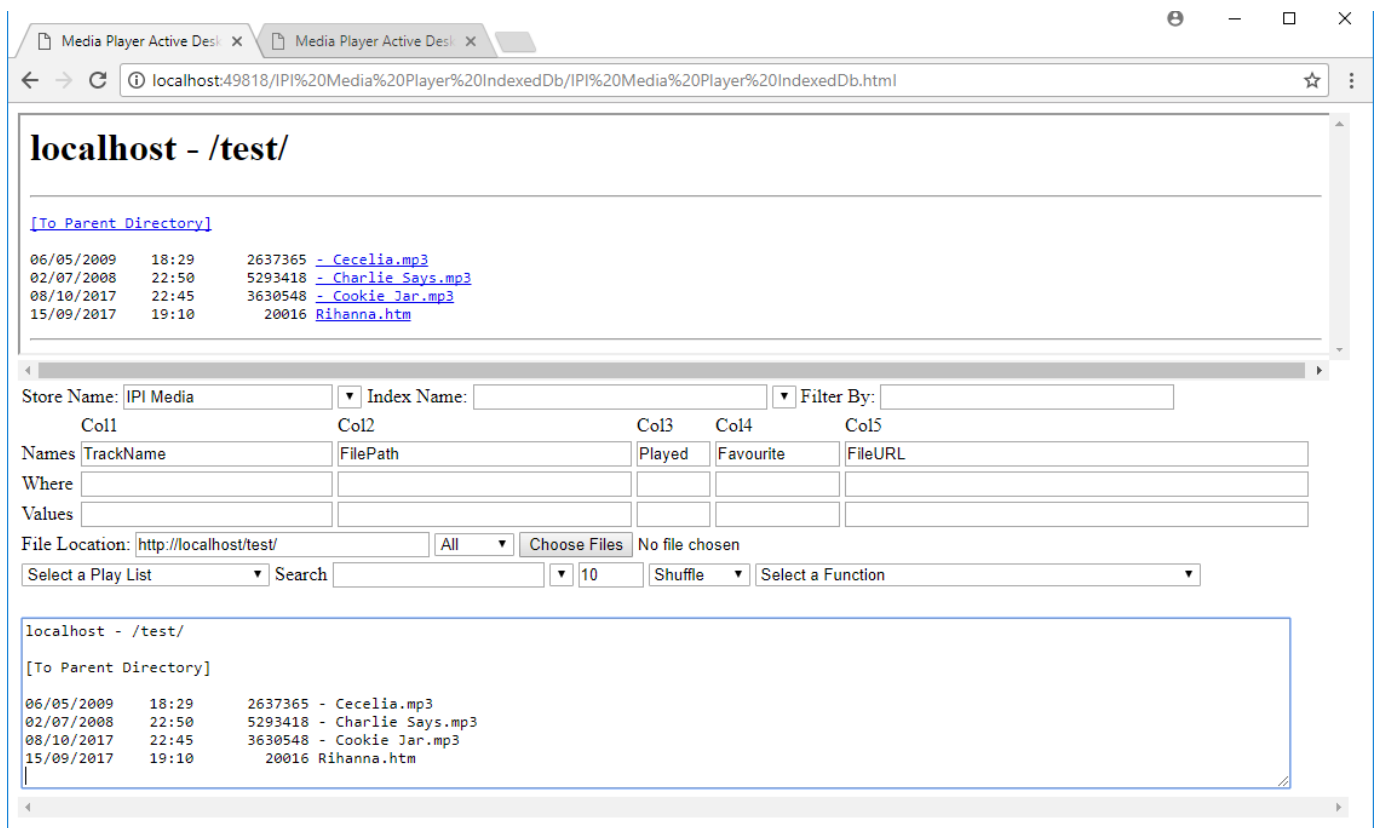
The screenshot shows the same web application interface as before, but with the 'Show/Hide Editor' option selected in the dropdown menu. The table of tracks is still visible, and the dropdown menu is open, showing the 'Show/Hide Editor' option highlighted.

Col1	Col2	Col3	Col4	Col5
Names	TrackName	FilePath	Played	Favourite
Where				
Values				

File Location: / All Choose Files No file chosen

Select a Play List Search 10 Shuffle Select a Function

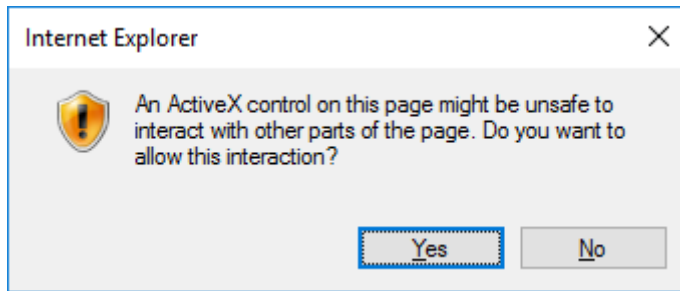
Show/Hide Editor



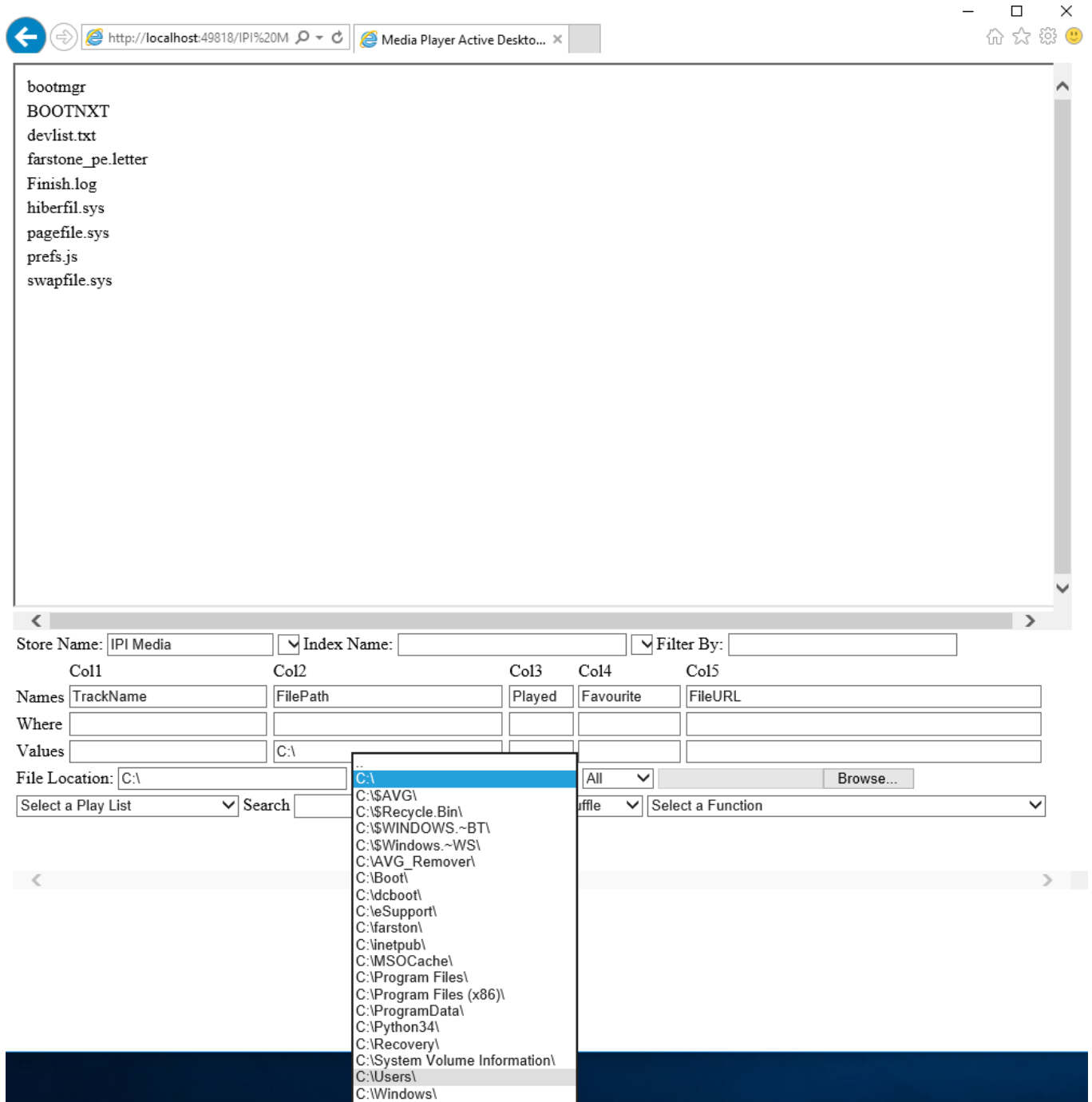
13. I have forced the Pages to run in Internet Explorer 10 compatible mode to add in VB Scripting function of searching and retrieving local system files. Please note that this function is only available in Internet Explorer when it is loaded from a web server; the internet; or Visual Studio IIS Express module. Right clicking on the pages from a local file folder and opening with either Firefox or Google Chrome will also allow you to search your system files and retrieve lists.

With Internet Explorer there is no need to copy the results from the iFrame to the TextArea as required in Firefox and Google Chrome.

14. After type the drive path (C:\) in the File Location input in Internet Explorer you will be prompted to Allow Active X Components. Without this you will not be able to do the search and retrieval. It will also unhide the Folder Selector



15. I guess at a later date I can reduce the folders listed populating the folder selector. But for now it's all there. Selecting a folder will then populate the File Location input with that entry and update the iFrame with the files located in that folder. The upper most selector is a ... which allows you to go back one level.



16. As you can see, the resulting list is already in the right format for uploading in to the database. Internet Explorer may be better than Firefox and Google Chrome in this respect as it allows you to update a database created when using a particular server whereas the other 2 will need you to create a new database as per the file path on your hard drive, but there again you can always open the file locally and still have it's database populate with both local and networked links. Unlike Internet Explorer and Google Chrome, Firefox requires you to put **file://** before the file path in order to recognise the file type on playback. Make sure to fill in the FilePath and Favourite inputs before using the **Upload iFrame Table Element** option to upload to the database

Store Name: IPI Media Index Name: Filter By:

Col1	Col2	Col3	Col4	Col5
Names	TrackName	FilePath	Played	Favourite
Where				
Values				

File Location: C:\Users\Paul\Music\ C:\Users\Paul\Music\ All Browse...

Select a Play List Search 10 Shuffle Select a Function

17. With Google Chrome you will have to type the **file://** and the full path to the folder containing the files you want in the File Location input. Then display the Editor Text area. Highlight the iFrame display and paste it in the Editor area. Again you will need to use the function **Create iFrame Table Element from Editor** (selecting it twice as the 1st time fails due to browser restrictions). Once done, remember to put the Folder Path in the FilePath input and an entry in the Favourite input. The Upload function appends the TrackName to the FilePath entry to create the FileURL and puts 0 for the Played entry. What is unusual about Google Chrome is that it also returns all hidden objects in it's display. So I have ensured that the code filters these out when reformatting the Editor Text in to the new iFrame resulting Table. I have added in a Selector for the extension types to be loaded. Selecting all will load everything. **file://** at the beginning of the File Location input is used by my code to get the right formatting but is not necessary in the FilePath input

Index of C:\users\paul\music\

Name	Size	Date Modified
- Cecelia.mp3	2.5 MB	06/05/2009, 18:29:28
- Charlie Says.mp3	5.0 MB	02/07/2008, 22:50:00
- Cookie Jar.mp3	3.5 MB	08/10/2017, 22:45:40
AlbumArt_{3719D521-EA91-4010-8CA3-F67B84AA22DA}_Large.jpg	26.5 kB	08/10/2017, 22:45:33
AlbumArt_{3719D521-EA91-4010-8CA3-F67B84AA22DA}_Small.jpg	6.8 kB	08/10/2017, 22:45:33
AlbumArtSmall.jpg	6.8 kB	08/10/2017, 22:45:33
desktop.ini	504 B	21/08/2017, 16:29:56
Folder.jpg	26.5 kB	08/10/2017, 22:45:33
Rihanna.htm	19.5 kB	15/09/2017, 19:10:01

Store Name: IPI Media Index Name: Filter By:

Col1	Col2	Col3	Col4	Col5
Names	TrackName	FilePath	Played	Favourite
Where				
Values		c:\users\paul\music\	Local	

File Location: file://c:\users\paul\music\ All Choose Files No file chosen

Select a Play List Search 10 Shuffle Select a Function

- Cecelia.mp3
- Charlie Says.mp3
- Cookie Jar.mp3
- Rihanna.htm

Store Name: IPI Media Index Name: Filter By:

Col1	Col2	Col3	Col4	Col5
Names	TrackName	FilePath	Played	Favourite
Where				
Values		c:\users\paul\music\	Local	

File Location: file://c:\users\paul\music\ All Choose Files No file chosen

Select a Play List Search 10 Shuffle Select a Function

Name	Size	Last Modified
- Cecelia.mp3	2576 KB	06/05/2009 18:29:28
- Charlie Says.mp3	5170 KB	02/07/2008 22:50:00
- Cookie Jar.mp3	3546 KB	08/10/2017 22:45:40
Rihanna.htm	20 KB	15/09/2017 19:10:01

All
Music
Picture
Video
JS
ASF
AVI
BAT
BMP
CDA
CSV
DOC
DVD
DVX
FLV
GIF
ICO
INF
INI
JPG

Store Name: IPI Media Index Name: Filter By:

Col1	Col2	Col3	Col4	Col5
Names	TrackName	FilePath	Played	Favourite
Where				
Values		file:///c:\users\paul\music\	Local	

File Location: file:///c:\users\paul\music\ All Browse... No files selected.

Select a Play List Search 10 Shuffle Select a Function

18. For Firefox you will need the preceding **file://** indicator before the folder path in both the File Location and FilePath inputs. Like Google you will have to copy the display from the iFrame to the Editor before running the same function to reformat it

file:///C:/Users/Paul/Documents/IPI International/IPI Media Player | Search

Index of file:///c:/users/paul/music/

Up to higher level directory

Name	Size	Last Modified
- Cecelia.mp3	2576 KB	06/05/2009 18:29:28
- Charlie Says.mp3	5170 KB	02/07/2008 22:50:00
- Cookie Jar.mp3	3546 KB	08/10/2017 22:45:40
Rihanna.htm	20 KB	15/09/2017 19:10:01

Store Name: Index Name: Filter By:

Col1	Col2	Col3	Col4	Col5
Names <input type="text" value="TrackName"/>	<input type="text" value="FilePath"/>	<input type="text" value="Played"/>	<input type="text" value="Favourite"/>	<input type="text" value="FileURL"/>
Where <input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Values <input type="text"/>	<input type="text" value="file:///c:/users/paul/music/"/>	<input type="text"/>	<input type="text" value="Local"/>	<input type="text"/>

File Location: No files selected.

Select a Play List Search Select a Function

Name	Size	Last Modified
File: - Cecelia.mp3	2576 KB	06/05/2009 18:29:28
File: - Charlie Says.mp3	5170 KB	02/07/2008 22:50:00
File: - Cookie Jar.mp3	3546 KB	08/10/2017 22:45:40
File: Rihanna.htm	20 KB	15/09/2017 19:10:01

19. To manipulate the document within the iFrame the function **Run Eitor As Javascript** appends **javascript:** to your entry then calls a **document location** function on the iFrame document using the resulting script.
20. I have provided some sample YouTube and Radio station links that can be loaded by selecting either in the **Select a Play List** selector.
21. Choose Files will be further developed, but for now will populate the iFrame with the 1st item selected.
22. Search will populate it's selector with the results of items found in the Results Store containing your string entry. Select an item to scroll it into view.
23. There are different Play Modes determined by the entry in the **Favourite** column
 - a. Radio will not initiate a Set Interval timer and some radio streamer files are only playable in Internet Explorer, so I have tried to find web pages that stream those stations for other browsers
 - b. YouTube also will not initiate the Set Interval timer and disable the video autoplay on the iFrame video element
 - c. All others (e.g IIS Music) will initiate the timer and should a track fail to load after 10 seconds it will then attempt to load the next track. Auto play will take over
 - i. **Increment** will play each track incrementally as long as the Played column value is not **1** and as long as you do not enter **0** in the Played input on the **Where** line, because whilst each track is being played the results table will update with the filtered results
 - ii. **Once** will play the current track then reload the page clearing all
 - iii. **Repeat** will play the same track over and over until Stop or one of the other play modes is selected. It will also not change the Played value to 1
 - iv. **Shuffle** (the default) will shuffle each track. Together with a filter on **Played = 0** will ensure quicker movement through all tracks and playing your entire list once
 - v. **Stop** will wait the default 10 seconds after a track comes to an end or is stopped before reloading the page
24. If you are like me and have all your media on a hefty MicroSD that can be made available on any Wi-Fi or Bluetooth network you may need to put the different network location in the FilePath input. This will allow the play mode to replace the FilePath of the displayed record with the input value.

The **Upload iFrame Table Element** option will also have its values in the **FilePath** and **Favourite** columns changed to any values given in the inputs on the values line.

Be a beginner with IndexedDB I can appreciate its merits. Although I believe Web SQL developed to the point of being able to provide the same storage and speed capabilities is a necessity!

What do I have to learn to get a multiplatform web page that needs no server and gives me a sqlite.db file to do with as I please?+