

Topalian JavaScript Journal

by
Christopher Andrew Topalian

All Rights Reserved
Copyright 2000-2024

Dedicated to God the Father

Topalian JavaScript Journal

file:///D:/_1Code/0_JS_Publish/Topalian_JavaScript_Journal/Topalian_JavaScript_

Topalian JavaScript Journal

```
[{"id":0,"date":"10/20/2024, 5:42:51 AM","text":"There was a loud noise outside about 2 minutes ago."},{ "id":1,"date":"10/20/2024, 5:43:25 AM","text":"There are flashing lights outside of the window now."}]
```

I traveled to the store earlier today and saw something interesting in the sky.

Post

-

+

Delete

10/20/2024, 5:42:51 AM
There was a loud noise outside about 2 minutes ago.

-

+

Delete

10/20/2024, 5:43:25 AM
There are flashing lights outside of the window now.

-

+

Delete

10/20/2024, 5:44:44 AM
I traveled to the store earlier today and saw something

5:45 AM
10/20/2024

Topalian JavaScript Journal

file:///D:/_1Code/0_JS_Publish/Topalian_JavaScript_Journal/Topalian_JavaScript_

```
[{"id":0,"date":"10/20/2024, 5:42:51 AM","text":"10/20/2024, 5:42:51 AM\nThere was a loud noise outside about 2 minutes ago.\n\nAdditional Info: I remember that it sounded like a low
```

Topalian JavaScript Journal

Type Journal Entry Here...

Post

-

+

Delete

```
10/20/2024, 5:42:51 AM
There was a loud noise outside about 2 minutes ago.

Additional Info: I remember that it sounded like a low
frequency thud.
```

5:46 AM
10/20/2024

Topalian JavaScript Journal

file:///D:/_1Code/0_JS_Publish/Topalian_JavaScript_Journal/Topalian_JavaScript_

```
[{"id":0,"date":"10/20/2024, 5:42:51 AM","text":"10/20/2024, 5:42:51 AM\nThere was a loud noise outside about 2 minutes ago.\n\nAdditional Info: I remember that it sounded like a low frequency thud."},{\n  "id":1,"date":"10/20/2024, 5:43:25 AM","text":"There are flashing lights outside of the window now."},\n  {"id":2,"date":"10/20/2024, 5:44:44 AM","text":"I traveled to the store earlier today and saw something interesting in the sky."}]
```

Topalian JavaScript Journal

Type Journal Entry Here...

Post

-

+

Delete

10/20/2024, 5:42:51 AM
There was a loud noise outside about 2 minutes ago.

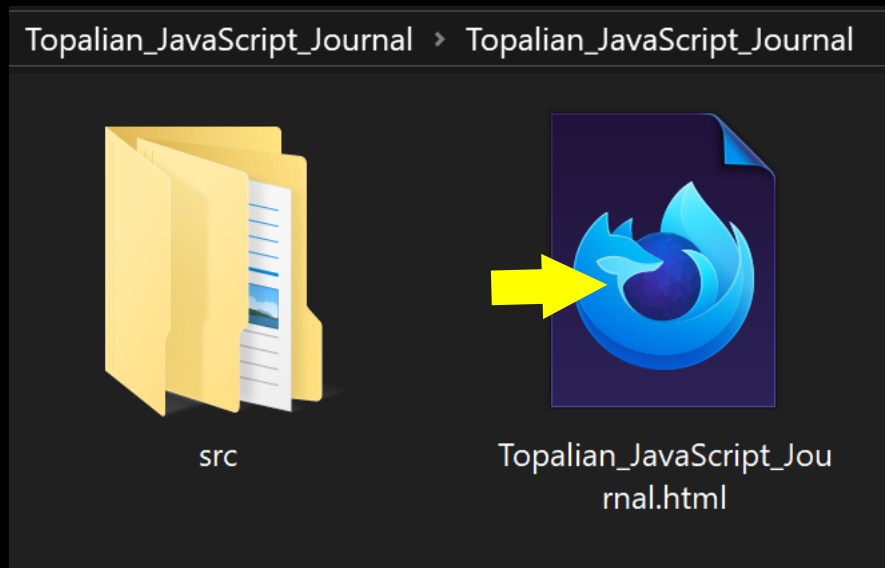
-

+

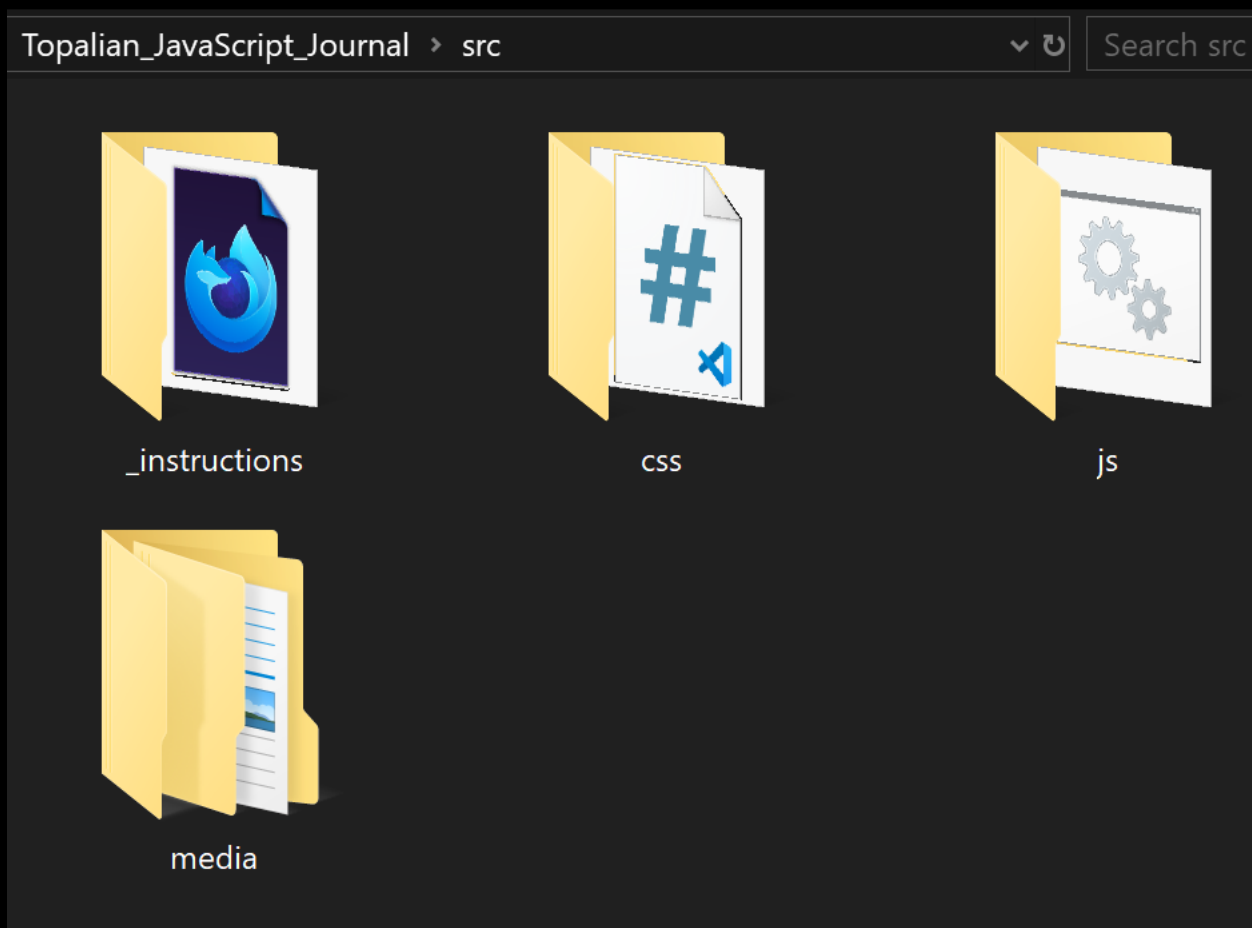
Delete

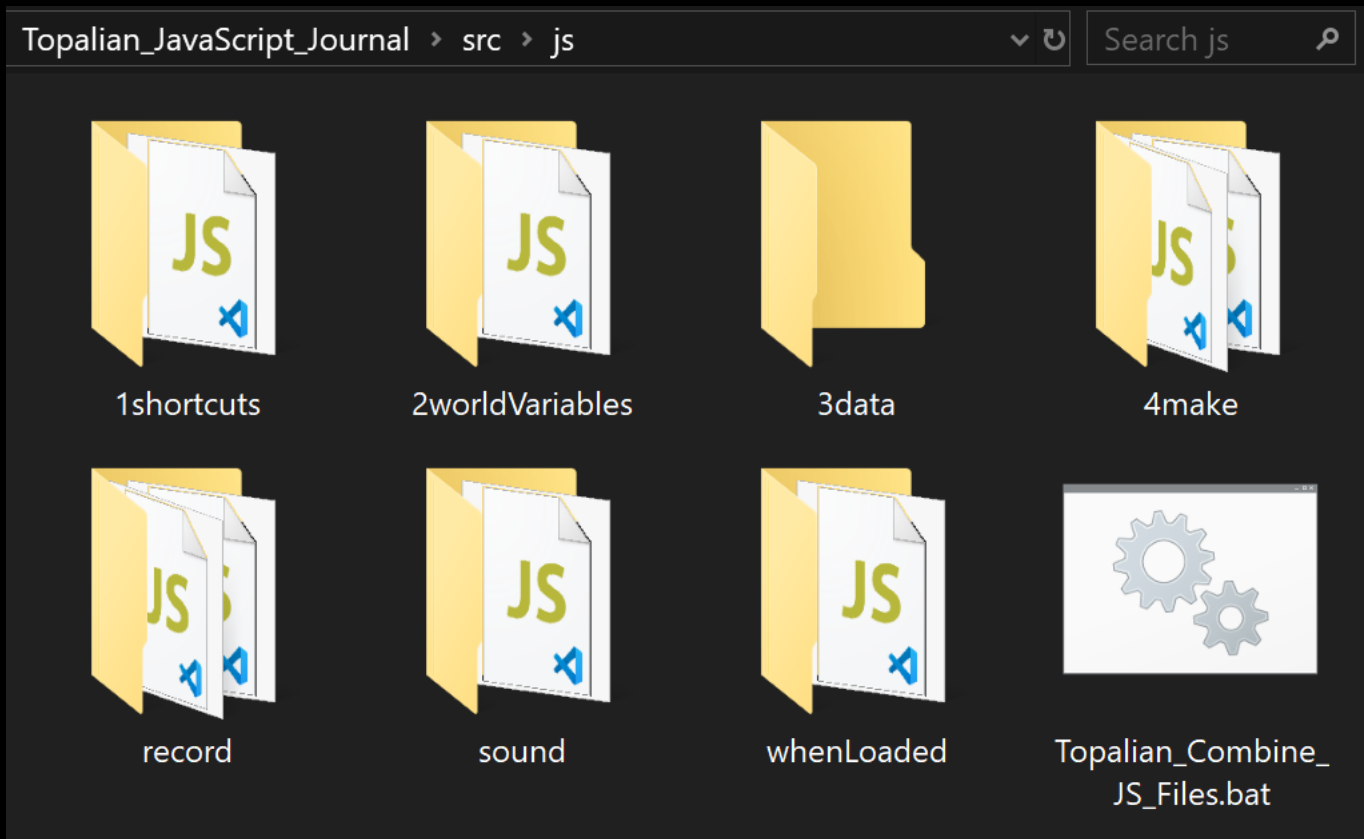
10/20/2024, 5:43:25 AM
There are flashing lights outside of the window now.

5:47 AM
10/20/2024

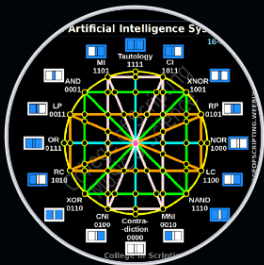


**Double Click the html file
to Open the Application**





To see our project structured as above, visit:
<https://github.com/ChristopherTopalian>



Christopher Topalian
 ChristopherTopalian

 Dedicated to God the Father

Logician. Inventor of True Artificial Intelligence. Computer Scientist. Computer Programming

**Below, is the online version of our application,
 which puts all content in one file for easy
 uploading.**

```
<!-- Dedicated to God the Father -->
<!-- All Rights Reserved Christopher Andrew
Topalian Copyright 2000-2024 -->
<!-- https://github.com/ChristopherTopalian -->
<!--
https://github.com/ChristopherAndrewTopalian
-->
<!--
Topalian_JavaScript_Journal_(Online_Version)
.html -->
<!-- Version 001 - (2024-10-20) -->
```

```
<html>
```

```
<head>
```

```
<title> Topalian JavaScript Journal </title>
```

```
<link rel = "icon" href =
"src/media/texture/icon.png" type =
"image/png">
```

```
<style>
```

```
/* style001.css */
```

```
body
```

```
{
```



```
background-color: rgb(40, 40, 40);  
font-family: arial;  
font-size: 30px;  
color: rgb(255, 255, 255);  
}
```

```
hr  
{  
    width: 500px;  
    margin: 8px;  
}
```

```
textarea  
{  
    background-color: rgb(0, 0, 0);  
    color: rgb(255, 255, 255);  
}
```

```
button  
{  
    background-color: rgb(0, 77, 129);  
    margin-left: 2px;  
    margin-right: 2px;  
    margin-top: 1px;  
    margin-bottom: 1px;  
    border-style: solid;
```

```
border-width: 1px;  
border-color: rgb(100, 100, 255);  
border-radius: 4px;  
padding-left: 7px;  
padding-right: 7px;  
padding-top: 1px;  
padding-bottom: 1px;  
opacity: 1.0;  
font-size: 14px;  
font-weight: bold;  
color: rgb(255, 255, 255);  
text-align: center;  
}
```

```
button:hover  
{  
    background-color: rgb(7, 31, 47);  
    border-color: rgb(0, 255, 255);  
    color: rgb(0, 255, 255);  
}
```

```
button:active  
{  
    color: rgb(255, 0, 255);  
}
```

```
/* -- */
```

```
/* Dedicated to God the Father */
```

```
/* All Rights Reserved Christopher Andrew  
Topalian Copyright 2000-2024 */
```

```
/* https://github.com/ChristopherTopalian */
```

```
/*
```

```
https://github.com/ChristopherAndrewTopalian  
*/
```

```
</style>
```

```
<script>
```

```
// shortcuts.js
```

```
function ge(whichId)
```

```
{
```

```
    let result =
```

```
document.getElementById(whichId);
```

```
    return result;
```

```
}
```

```
function ce(whichType)
```

```
{
```

```
    let result =  
document.createElement(whichType);  
    return result;  
}
```

```
function ba(whichElement)  
{  
    let result =  
document.body.append(whichElement);  
    return result;  
}
```

```
function ct(whichText)  
{  
    let result =  
document.createTextNode(whichText);  
    return result;  
}
```

```
function gr(whichId)  
{  
    let result =  
ge(whichId).getBoundingClientRect();  
    return result;  
}
```

```
function cl(whichMessage)
{
    let result = console.log(whichMessage);
    return result;
}

//--//

// Dedicated to God the Father
// All Rights Reserved Christopher Andrew
// Topalian Copyright 2000-2024
// https://github.com/ChristopherTopalian
//
// https://github.com/ChristopherAndrewTopalian
//
// https://sites.google.com/view/CollegeOfScripting
//

// world_variables.js

let counter = 0;

let recordEntry;

let records = [];
```

```
//--//
```

```
// Dedicated to God the Father
```

```
// All Rights Reserved Christopher Andrew  
Topalian Copyright 2000-2024
```

```
// https://github.com/ChristopherTopalian
```

```
//
```

```
https://github.com/ChristopherAndrewTopalian
```

```
//
```

```
https://sites.google.com/view/CollegeOfScripting
```

```
// makeInterface.js
```

```
function makeInterface()
```

```
{
```

```
    makeMainLayout();
```

```
    makePostLayout();
```

```
    makeRecordsTextBox();
```

```
    makeTextInputForPosting();
```

```
    makePostButton();
```

```
    listenForPostButtonClick();
```

```
listenForPostButtonHover();

makeTitleOfApp();
}

//--//

// Dedicated to God the Father
// All Rights Reserved Christopher Andrew
// Topalian Copyright 2000-2024
// https://github.com/ChristopherTopalian
//
// https://github.com/ChristopherAndrewTopalian
//
// https://sites.google.com/view/CollegeOfScripti
ng

// makeMainLayout.js

function makeMainLayout()
{
    let mainDiv = ce('div');
    mainDiv.id = 'mainLayout';
    mainDiv.style.display = 'flex';
    mainDiv.style.flexDirection = 'column';
    ba(mainDiv);
}
```

```
}
```

```
//--//
```

```
// Dedicated to God the Father  
// All Rights Reserved Christopher Andrew  
// Topalian Copyright 2000-2024  
// https://github.com/ChristopherTopalian  
//  
// https://github.com/ChristopherAndrewTopalian  
//  
// https://sites.google.com/view/CollegeOfScripting
```

```
// makePostButton.js
```

```
function makePostButton()  
{  
  let mainDiv = ce("div");  
  mainDiv.style.display = 'flex';  
  mainDiv.style.display = 'row';  
  ge('mainLayout').append(mainDiv);
```

```
//--//
```

```
let postButton = ce("button");
```



```
postButton.id = "postButton";  
postButton.textContent = "Post";  
postButton.style.width = 70 + 'px';  
mainDiv.append(postButton);  
}
```

```
function listenForPostButtonClick()  
{  
  if (ge('postButton'))  
  {  
    let postListener =  
ge('postButton').addEventListener('click',  
function(theEvent)  
  {  
    clickSound();  
  
    let theText = ge("textWords").value;  
  
    //-//  
  
    let controlsContainer = ce('div');  
    controlsContainer.style.margin = '5px';  
  
ge('postLayout').append(controlsContainer);  
  
    //-//  
  }  
}
```

```
let theNavigation = ce("div");
theNavigation.style.display = 'flex';
theNavigation.style.flexDirection = 'row';
theNavigation.id = "navigation" +
counter;
ge('postLayout').append(theNavigation);

//-//

counter += 1;

if (counter == 0)
{
    records = [];
    ge("recordsTextBox").value = "";
}

//-//

let minimizeButton = ce("button");
minimizeButton.textContent = "_";
minimizeButton.style.height = 30 + 'px';
minimizeButton.onclick = function()
{
    clickSound();
}
```

```
// adjust to minimum height
ge(theTextEntry.id).style.height = 60 +
"px";
};
minimizeButton.onmouseover =
function()
{
    hoverSound();
};

theNavigation.append(minimizeButton);

// -//

let maximizeButton = ce("button");
maximizeButton.style.height = 30 + 'px';
maximizeButton.textContent = "+";
maximizeButton.onclick = function()
{
    clickSound();

    ge(theTextEntry.id).style.height = 200
+ "px";
```

```
        ge(theTextEntry.id).style.height =  
ge(theTextEntry.id).scrollHeight + 'px';  
    };  
    maximizeButton.onmouseover =  
function()  
    {  
        hoverSound();  
    };  
    theNavigation.append(maximizeButton);  
  
//-//
```

```
let deleteButton = ce("button");  
deleteButton.textContent = "Delete";  
deleteButton.style.height = 30 + 'px';  
deleteButton.onclick = function()  
{  
    clickSound();  
}
```

```
deleteThisRecordButton(theTextEntry.id);
```

```
        ge("recordsTextBox").value =  
JSON.stringify(records);  
    };  
    deleteButton.onmouseover = function()
```

```
{
    hoverSound();
};
theNavigation.append(deleteButton);
```

```
//-//
```

```
let theTextEntry = ce("textarea");
theTextEntry.style.width = '700px';
theTextEntry.style.height = '60px';
theTextEntry.style.borderRadius = '8px';
theTextEntry.style.paddingLeft = '10px';
theTextEntry.style.paddingRight =
'10px';
theTextEntry.style.overflowY = 'scroll';
theTextEntry.style.fontSize = '20px';
theTextEntry.id = counter - 1;
theTextEntry.title = counter - 1;
```

```
//-//
```

```
let theDate = new
Date().toLocaleString();
```

```
theTextEntry.value = theDate + "\n" +
ge("textWords").value;
```

```
    addRecord(theTextEntry.id, theDate,  
ge("textWords").value);
```

```
    //-//
```

```
    theTextEntry.onkeyup = function()  
{  
    editRecord(theTextEntry.id);  
};
```

```
    theNavigation.append(theTextEntry);  
});  
}  
}
```

```
function listenForPostButtonHover()  
{  
    let postListenerHover =  
ge('postButton').addEventListener('mouseover'  
, function(theEvent)  
    {  
        hoverSound();  
    });  
}
```

//--//

// Dedicated to God the Father

// All Rights Reserved Christopher Andrew
Topalian Copyright 2000-2024

// <https://github.com/ChristopherTopalian>

//

<https://github.com/ChristopherAndrewTopalian>

//

<https://sites.google.com/view/CollegeOfScripting>

// makePostLayout.js

function **makePostLayout**()

{

let mainDiv = **ce**('div');

mainDiv.id = 'postLayout';

mainDiv.style.display = 'flex';

mainDiv.style.flexDirection = 'column';

ba(mainDiv);

}

//--//

// Dedicated to God the Father

```
// All Rights Reserved Christopher Andrew  
Topalian Copyright 2000-2024  
// https://github.com/ChristopherTopalian  
//  
https://github.com/ChristopherAndrewTopalian  
//  
https://sites.google.com/view/CollegeOfScripti  
ng
```

```
// makeRecordsTextBox.js
```

```
function makeRecordsTextBox()  
{  
  let mainDiv = ce("div");  
  mainDiv.id = 'mainRecords';  
  ge('mainLayout').append(mainDiv);  
  
  //-//  
  
  let theTextBox = ce("textarea");  
  theTextBox.value = records;  
  theTextBox.id = "recordsTextBox";  
  theTextBox.style.width = 500 + 'px';  
  theTextBox.style.height = 50 + 'px';  
  mainDiv.append(theTextBox);  
}
```



```
//--//
```

```
// Dedicated to God the Father
```

```
// All Rights Reserved Christopher Andrew  
Topalian Copyright 2000-2024
```

```
// https://github.com/ChristopherTopalian
```

```
//
```

```
https://github.com/ChristopherAndrewTopalian
```

```
//
```

```
https://sites.google.com/view/CollegeOfScripting
```

```
// makeTextInputForPosting.js
```

```
function makeTextInputForPosting()
```

```
{
```

```
    let textWords = ce("textarea");
```

```
    textWords.id = "textWords";
```

```
    textWords.style.width = '70%';
```

```
    textWords.style.height = '30vh';
```

```
    textWords.style.paddingLeft = '20px';
```

```
    textWords.style.paddingRight = '20px';
```

```
    textWords.style.paddingTop = '20px';
```

```
    textWords.style.paddingBottom = '20px';
```

```
    textWords.style.borderRadius = '8px';
```

```
textWords.style.backgroundColor = 'rgb(0, 0, 0)';
textWords.style.fontFamily = 'arial';
textWords.style.fontSize = '20px';
textWords.style.color = 'rgb(255, 255, 255)';
textWords.placeholder = 'Type Journal Entry Here...';
ge('mainLayout').append(textWords);
}
```

```
//--//
```

```
// Dedicated to God the Father
// All Rights Reserved Christopher Andrew Topalian Copyright 2000-2024
// https://github.com/ChristopherTopalian
//
https://github.com/ChristopherAndrewTopalian
//
https://sites.google.com/view/CollegeOfScripting
```

```
// makeTitleOfApp.js
```

```
function makeTitleOfApp()
{
```

```
// mainDiv
let mainDiv = ce('div');
mainDiv.style.position = 'absolute';
mainDiv.style.right = '20px';
mainDiv.style.top = '2px';
mainDiv.style.zIndex = 1;
ba(mainDiv);

// -//

// titleOfApp
let titleOfApp = ce('a');
titleOfApp.id = 'titleOfApp';
titleOfApp.textContent = 'Topalian JavaScript
Journal';
titleOfApp.href =
'https://github.com/christophertopalian/';
titleOfApp.target = '_blank';
titleOfApp.style.fontSize = '17px';
titleOfApp.style.fontWeight = 'bold';
titleOfApp.style.textDecoration = 'none';
titleOfApp.style.color = 'gray';
mainDiv.append(titleOfApp);
}

// - -//
```

```
// Dedicated to God the Father
// All Rights Reserved Christopher Andrew
Topalian Copyright 2000-2024
// https://github.com/ChristopherTopalian
//
https://github.com/ChristopherAndrewTopalian
//
https://sites.google.com/view/CollegeOfScripti
ng
```

```
// addRecord.js
```

```
function addRecord(whichId, whichDate,
whichWords)
{
    // if records exist
    if (records.length > 0)
    {
        let largestId =
Math.max.apply(Math,records.map(function(x)
        {
            return x.id;
        }
    ));
```

```
        addingOneToLargestId = largestId + 1;
```

```
}  
// else if no records exist, such as after  
clearing all records  
else  
{  
    addingOneToLargestId = 0;  
  
    records = [];  
}  
  
recordEntry =  
{  
    id: addingOneToLargestId,  
    date: whichDate,  
    text: whichWords  
}  
  
records.push(recordEntry);  
  
cl("records = " + JSON.stringify(records) +  
";");  
  
ge("recordsTextBox").value =  
JSON.stringify(records);  
  
return records;
```

```
}
```

```
//--//
```

```
// Dedicated to God the Father
```

```
// All Rights Reserved Christopher Andrew
```

```
Topalian Copyright 2000-2024
```

```
// https://github.com/ChristopherTopalian
```

```
//
```

```
https://github.com/ChristopherAndrewTopalian
```

```
//
```

```
https://sites.google.com/view/CollegeOfScripting
```

```
// deleteThisRecordButton.js
```

```
function deleteThisRecordButton(whichId)
```

```
{
```

```
    let warningDelete = confirm("DELETE ONLY  
This Record?");
```

```
    if (warningDelete == true)
```

```
    {
```

```
        // cl("Okay");
```

```
    }
```

```
    else if (warningDelete == false)
```

```
{  
    alert("Canceled");  
  
    return;  
}  
  
records.splice(whichId, 1);  
  
cl("records = " + JSON.stringify(records) +  
";");  
  
ge(whichId).remove();  
  
ge("navigation" +  
parseInt(whichId)).remove();  
  
cl("navigation" + parseInt(whichId));  
  
cl("records length = " + records.length);  
  
if (whichId == 0)  
{  
    counter = records.length;  
  
    records = [];  
}
```

```
else
{
    counter = records.length;
}
}

//--//

// Dedicated to God the Father
// All Rights Reserved Christopher Andrew
// Topalian Copyright 2000-2024
// https://github.com/ChristopherTopalian
//
// https://github.com/ChristopherAndrewTopalian
//
// https://sites.google.com/view/CollegeOfScripting
//

// editRecord.js

function editRecord(whichId)
{
    let existingId = records[whichId].id;

    recordEntry =
    {
```



```
    id: existingId, // whichId
    date: records[whichId].date,
    text: ge(whichId).value
}
```

```
records.splice(whichId, 1, recordEntry);
```

```
cl("records = " + JSON.stringify(records) +
";");
```

```
    ge("recordsTextBox").value =
JSON.stringify(records);
}
```

```
//--//
```

```
// Dedicated to God the Father
// All Rights Reserved Christopher Andrew
// Topalian Copyright 2000-2024
// https://github.com/ChristopherTopalian
//
// https://github.com/ChristopherAndrewTopalian
//
// https://sites.google.com/view/CollegeOfScripti
ng
```

```
// sound.js
```

```
let audioCtx = new (window.AudioContext ||  
window.webkitAudioContext)();
```

```
function playTone(frequency, duration,  
waveType, volume, detune, attackTime,  
releaseTime, panValue)  
{  
  let oscillator = audioCtx.createOscillator();  
  // 'sine', 'square', 'sawtooth', 'triangle'  
  oscillator.type = waveType;
```

```
  oscillator.frequency.setValueAtTime(frequency,  
audioCtx.currentTime);  
  oscillator.detune.setValueAtTime(detune,  
audioCtx.currentTime);
```

```
  let gainNode = audioCtx.createGain();  
  let pannerNode =  
audioCtx.createStereoPanner();
```

```
  oscillator.connect(gainNode);  
  gainNode.connect(pannerNode);  
  pannerNode.connect(audioCtx.destination);
```

```
gainNode.gain.setValueAtTime(0,  
audioCtx.currentTime);
```

```
// attack
```

```
gainNode.gain.linearRampToValueAtTime(volu  
me, audioCtx.currentTime + attackTime);  
gainNode.gain.linearRampToValueAtTime(0,
```

```
// release
```

```
audioCtx.currentTime + duration / 1000 -  
releaseTime);
```

```
pannerNode.pan.setValueAtTime(panValue,  
audioCtx.currentTime);
```

```
oscillator.start();  
oscillator.stop(audioCtx.currentTime +  
duration / 1000);  
}
```

```
function hoverSound()  
{  
    playTone(  
        300,    // frequency  
        500,    // duration
```

```
'sine', // waveType
0.2,    // volume
50,     // detune
0.2,    // attackTime
0.2,    // releaseTime
-0.5    // panValue -0.5 is left
);
}
```

```
function clickSound()
{
    playTone(
        400,    // frequency
        500,    // duration
        'sine', // waveType
        0.2,    // volume
        50,     // detune
        0.2,    // attackTime
        0.2,    // releaseTime
        -0.5    // panValue -0.5 is left
    );
}

//--//
```

// Dedicated to God the Father

```
// All Rights Reserved Christopher Andrew  
Topalian Copyright 2000-2024  
// https://github.com/ChristopherTopalian  
//  
https://github.com/ChristopherAndrewTopalian  
//  
https://sites.google.com/view/CollegeOfScripti  
ng
```

```
// whenLoaded.js
```

```
function whenLoaded()  
{  
    makeInterface();  
  
    ge("recordsTextBox").value =  
JSON.stringify(records);  
}
```

```
//--//
```

```
// Dedicated to God the Father  
// All Rights Reserved Christopher Andrew  
Topalian Copyright 2000-2024  
// https://github.com/ChristopherTopalian
```

```
//  
https://github.com/ChristopherAndrewTopalian  
//  
https://sites.google.com/view/CollegeOfScripti  
ng
```

```
</script>
```

```
</head>
```

```
<body onload = "whenLoaded();">
```

```
</body>
```

```
</html>
```

How to Paste Code from a PDF that has Junk Characters.

When we paste from a pdf into VSCode, it might look like this:

```
function combineJSFiles(directory,  
scriptFilename)  
{  
  let outputFilePath = path.join  
(directory, 'main.js');  
  
  let fileContents = [];
```

We can't leave those junk characters in the code, so we remove them with find/replace.

We Find 1 of the spaces.

We Replace All with the 1 space that we typed.
This gets rid of the junk characters in the code.

We highlight 1 space with our mouse arrow:

```
function combineJSFiles(directory,
scriptFilename)
```

```
{
```

```
  let outputFilePath = path.join
(directory, 'main.js');
```

```
  let fileContents = [];
```

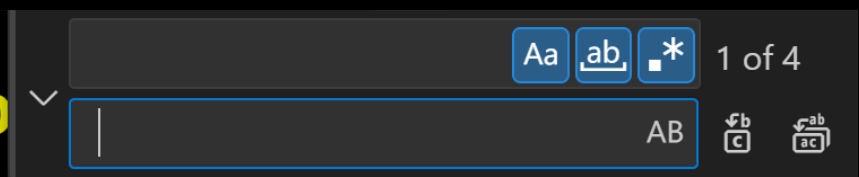
We press Control + H to open the Find/Replace feature and Replace All with our own Space

```
function comb
scriptFilename)
```

```
{
```

```
  let outputFilePath = path.join
(directory, 'main.js');
```

```
  let fileContents = [];
```



Here we see that the Find/Replace All has replaced the junk characters with our working spaces instead:

```
function combineJSFiles(directory,  
scriptFilename)  
{  
    let outputFilePath = path.join  
    (directory, 'main.js');  
  
    let fileContents = [];
```

Now that the code
has no junk characters,
it can run.

How to Combine .js files into one main.js file using Command Prompt

**Version for when we have only
ONE folder of .js files that we
want to combine.**

// HowToCombineJSFilesOneFolder.js

First, we add two new lines at the end of every script. This way they will later combine nicely with a line break in between each script.

We open our js folder.

In our js project folder, we type
cmd

in the address bar of the folder and then
press enter

This opens our js folder in the Command
prompt

We type in the words
copy *.js main.js
and then press enter

This creates a new file that is named main.js
This new file contains all .js files in ONE file.

But, there is a junk character at the end of the main.js script that we have to delete. In VSCode the character might be called SUB

```
titleContainer.append(titleOfApp);  
}
```

SUB

We remove this junk SUB character and the code will now run.

```
titleContainer.append(titleOfApp);  
}
```

As we can see, the junk character is removed.

<!-- Now, in our html code, we can type -->
<script src = 'main.js'></script>

This makes it much easier to upload our js code to our website.

How to Combine .js files into one main.js file using Command Prompt

**Version for when we have js
scripts in subfolders in our js
project folder, that we want to
combine.**

// HowToCombineJSFiles.js

TUTORIAL:

How to Combine all .js files in all folders that are in our js folder.

Getting things ready:

We should add two new lines at the end every script. This way they will combine nicely with a line break in between each script.

Step One: Open our js folder

Step Two: Type in the address bar of the js folder, `cmd`, press Enter

This opens our js folder in the command prompt

Step Three: Type the command shown below in the command prompt and then press Enter

```
for /r "%CD%" %i in (*.js) do type "%i" >>
main.js
```

Now we have a newly created .js file named main.js that has all of our js files included into one file.

This makes it easy to upload our application and easy to find out how many lines of code our project is.

To use our main.js file, we include it in our html file code:

```
<script src = 'js/main.js'></script>
```

Happy Scripting :-)

How to Combine .js files into one main.js file using a **batch file**

**Version for when we have js
scripts in subfolders in our js
project folder, that we want to
combine.**

// HowToCombineJSFilesUsingBatFile.js

We can combine all of the .js files that are located in our js folder into one main.js file, using either:

The Command Prompt Method

or

The .bat File Method

The .bat file method is very easy.

We double click the bat file, which is located in our js folder, and it will make a main.js file, which includes all .js files in the js folder, including all .js files in all subdirectories of our js folder.

This is a very easy way to combine our .js files, because we can double click the .bat file anytime, and it will again generate the main.js file, which includes all .js files in the js folder, including all .js files in all subdirectories of our js folder. This makes uploading our application online much easier.

Happy Scripting :-)

:: Topalian_Combine_JS_Files.bat

:: This .bat File Combines All .js files in all folders of our project folder, into one main.js file.

:: To activate this .bat file, we double click the .bat file, while it is located in our js folder.

@echo off

:: set the output file name

set "output=main.js"

:: clear existing output file

type nul > "%output%"

:: loop through all JavaScript files in subdirectories

for /r %%i in (*.js) do (

 :: append the content of each file to the output file

 type "%%i" >> "%output%"

)

echo "JavaScript files combined into %output% successfully."

How to Combine .js files into one main.js file using Node.js

**This version will successfully
combine a single folder of js
files.**

**It also works to combine js files
in all subdirectories.**

// Topalian_Combine_JS_Files.js

```
let fs = require('fs');
let path = require('path');

function combineJSFiles(directory,
scriptFilename)
{
    let outputFilePath = path.join(directory,
'main.js');

    let fileContents = [];

    function traverseFolder(folder)
    {
        let files = fs.readdirSync(folder);

        for (let i = 0; i < files.length; i++)
        {
            let file = files[i];

            let filePath = path.join(folder, file);

            let stats = fs.statSync(filePath);

            if (stats.isDirectory())
```

```

    {
        traverseFolder(filePath);
    }
    else if (path.extname(filePath) === '.js')
    {
        let content = fs.readFileSync(filePath,
'utf8');
        // check if file is not script file itself
        if (filePath !== scriptFilename)
        {
            fileContents.push(content);
        }
    }
}
}

traverseFolder(directory);

fs.writeFileSync(outputFilePath,
fileContents.join('\n'), 'utf8');

console.log(`Combined $
{fileContents.length} .js files into $
{outputFilePath}`);
}

```

```
// get current directory of script  
let currentDirectory = process.cwd();
```

```
// get filename of script  
let scriptFilename = __filename;
```

```
combineJSFiles(currentDirectory,  
scriptFilename);
```

How to Combine .js files into one main.js file using Python

**This version will successfully
combine a single folder of js
files.**

**It also works to combine js files
in all subdirectories.**

Topalian_Combine_JS_Files.py

```
import os
```

```
def combineJSFiles(directory, scriptFileName):  
    outputFilePath = os.path.join(directory,  
'main.js')
```

```
    fileContents = []
```

```
def traverseFolder(folder):
```

```
    for root, dirs, files in os.walk(folder):
```

```
        for file in files:
```

```
            filePath = os.path.join(root, file)
```

```
            if filePath != scriptFileName and
```

```
filePath.endswith('.js'):
```

```
                with open(filePath, 'r',
```

```
encoding='utf-8') as f:
```

```
                    fileContents.append(f.read())
```

```
traverseFolder(directory)
```

```
with open(outputFilePath, 'w',  
encoding='utf-8') as f:
```

```
    f.write('\n'.join(fileContents))
```

```
    print(f"Combined {len(fileContents)} .js files  
into {outputFilePath}")
```

```
# get current directory of script  
currentDirectory =  
os.path.dirname(os.path.abspath(__file__))
```

```
# get filename of script  
scriptFileName = os.path.abspath(__file__)
```

```
combineJSFiles(currentDirectory,  
scriptFileName)
```

What other file types can we combine?

We have combined .js files in this book, but we might choose to instead combine:

.py or .html or .txt

This is very useful for book making.

In each of the scripts shown in this book, we can manually change the parts where it says .js, with .py, if we wanted to, for instance, copy all .py files into one main.py file.

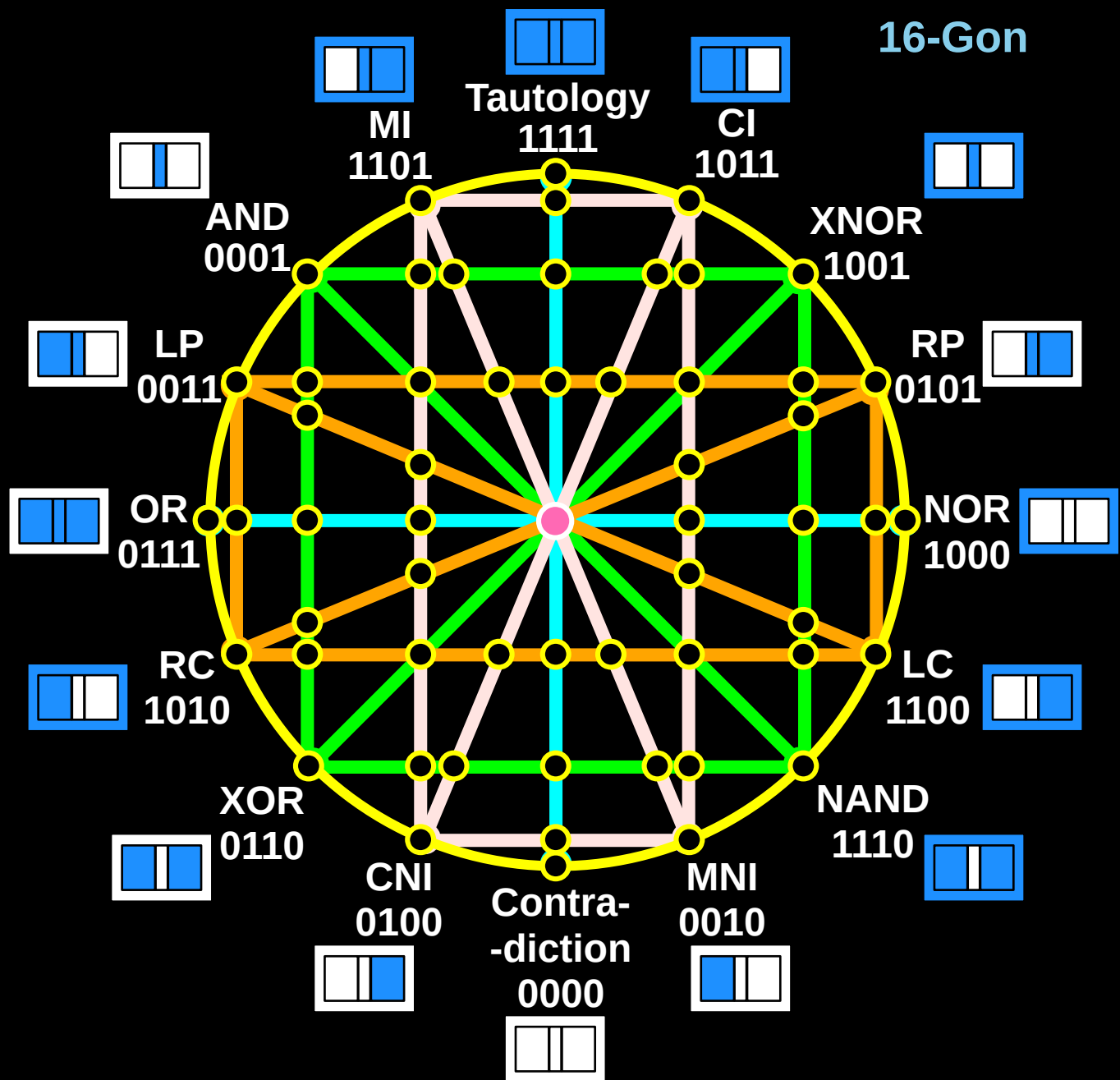
We can do the same thing for .html files, where we change the file type it will be combining to .html and it will combine all .html files into one main.html file.

We add two line breaks at the end of all files, so that there is space between files, when they are combined.

Remember too, that not all file types will combine, but the ones above will.

The original files are not changed. The content from the original files is only copied from.

True Artificial Intelligence System



For More Tutorials:

[GitHub.com/ChristopherTopalian](https://github.com/ChristopherTopalian)

[GitHub.com/ChristopherAndrewTopalian](https://github.com/ChristopherAndrewTopalian)

[Sites.google.com/view/CollegeOfScripting](https://sites.google.com/view/CollegeOfScripting)

CollegeOfScripting.weebly.com

CollegeOfScripting.wordpress.com

[Youtube.com/ScriptingCollege](https://youtube.com/ScriptingCollege)

[Twitter.com/CollegeOfScript](https://twitter.com/CollegeOfScript)

[Rumble.com/user/CollegeOfScripting](https://rumble.com/user/CollegeOfScripting)

Dedicated to God the Father

**This book is created by the
College of Scripting Music & Science.
Always remember, that each time you write a
script with a pencil and paper, it becomes
imprinted so deeply in memory that the
material and methods are learned extremely
well.**

**When you Type the scripts, the same is true.
The more you type and write out the scripts by
keyboard or pencil and paper, the more you
will learn programming!**

**Write and Type every example that you find.
Keep all of your scripts organized.
Every script that you create increases your
programming abilities.
SEEING CODE, is one thing,
but WRITING CODE is another.
Write it, Type it, Speak it, See it, Dream it.**

CollegeOfScripting.weebly.com