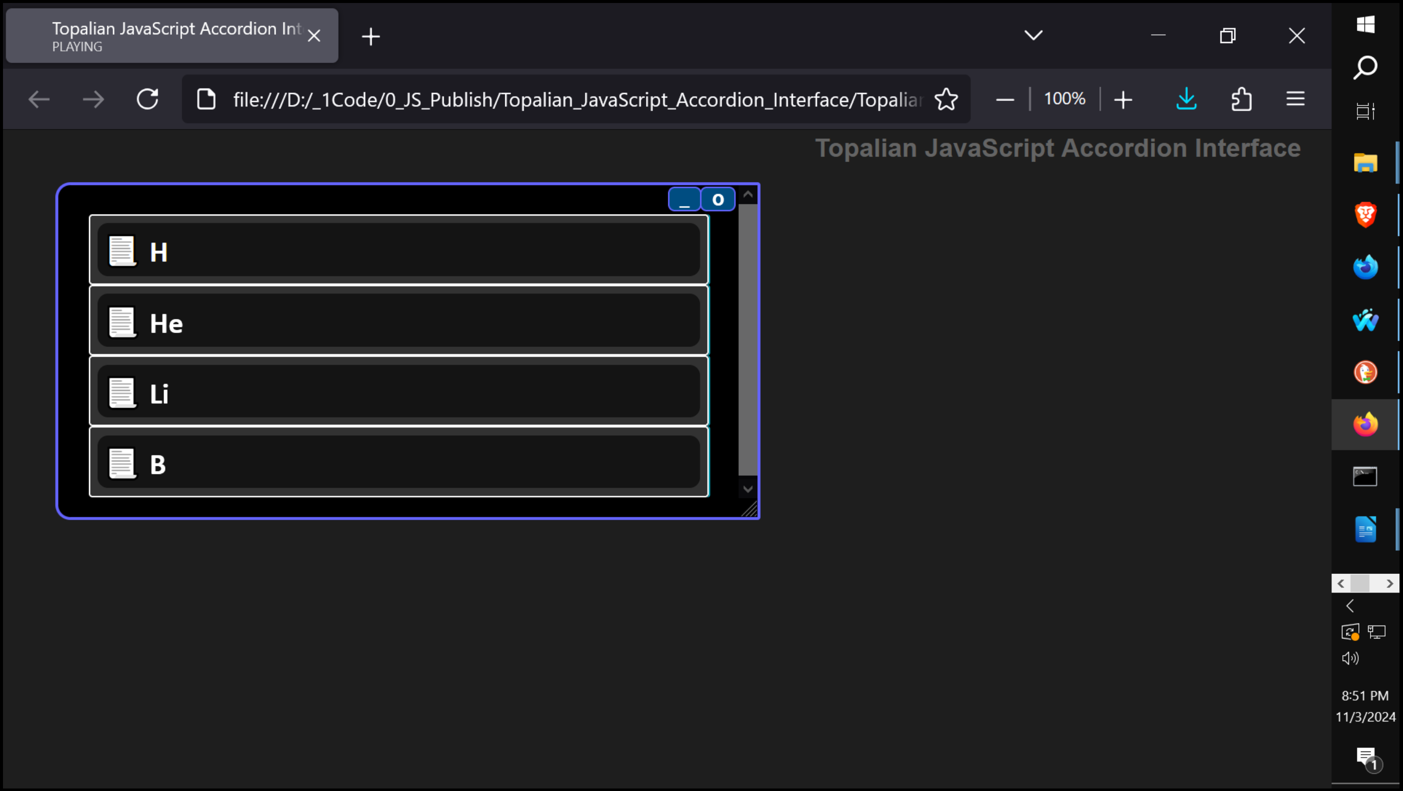
Topalian JavaScript Accordion Interface

by

Christopher Andrew Topalian

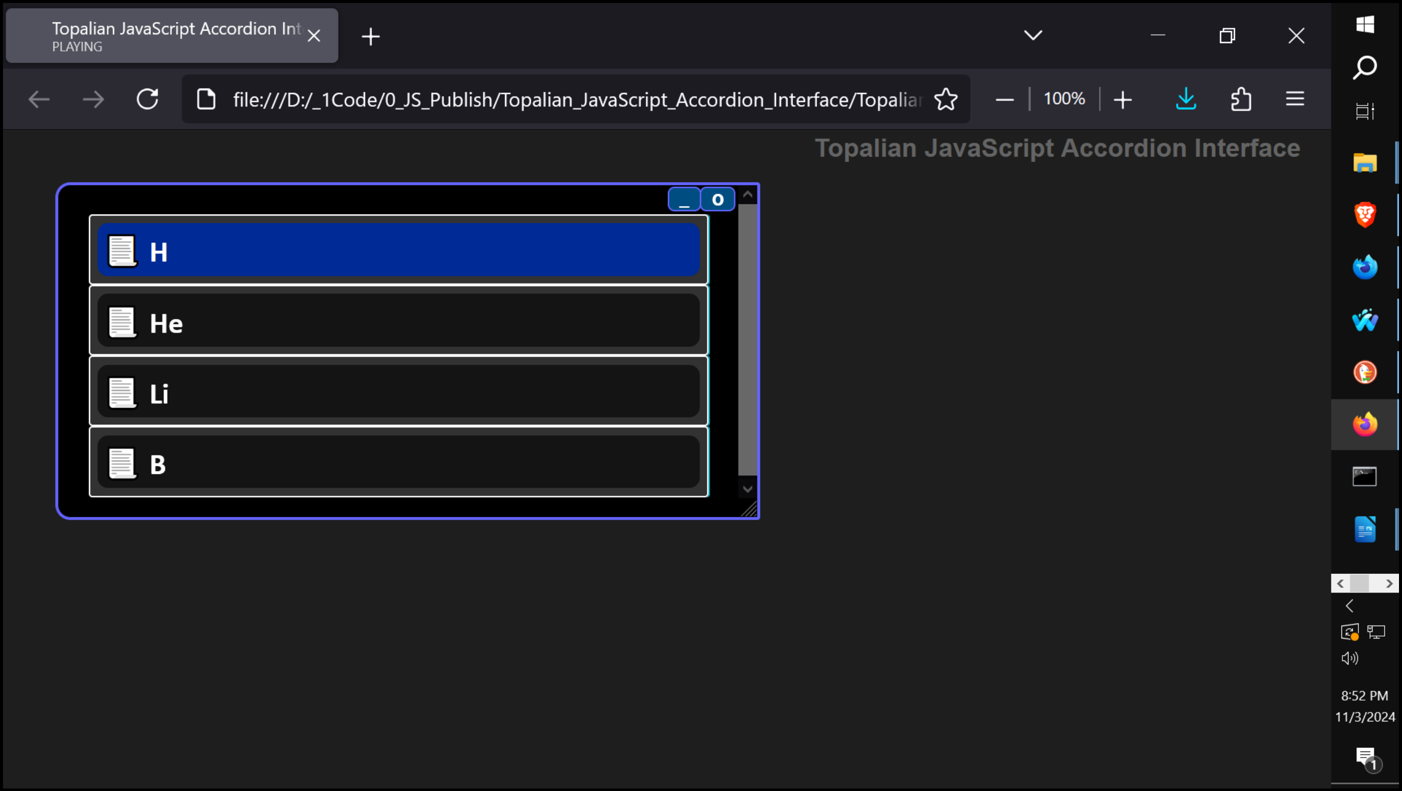
Dedicated
to
God the Father

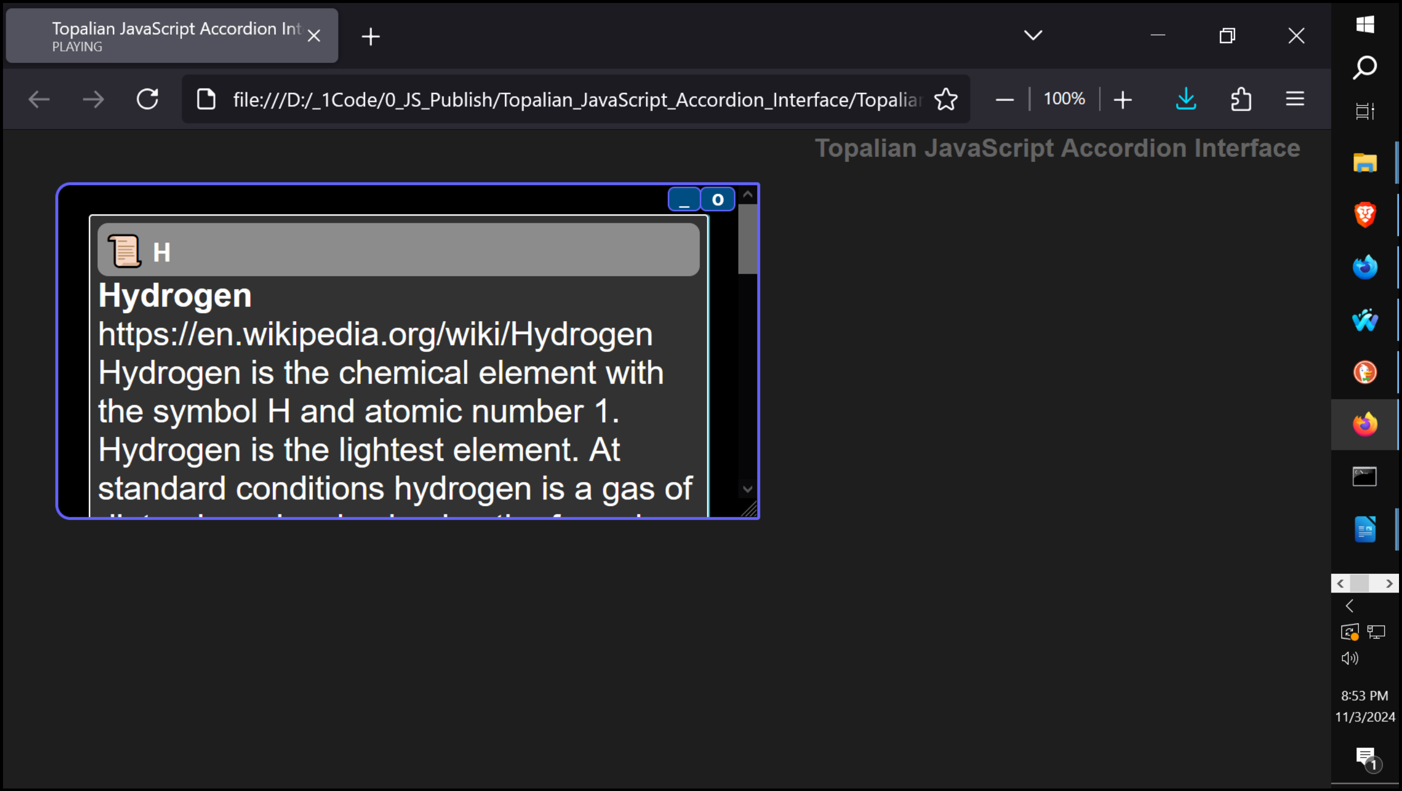


Topalian JavaScript Accordion Int
PLAYING

file:///D:/_1Code/0_JS_Publish/Topalian_JavaScript_Accordion_Interface/Topalia

Topalian JavaScript Accordion Interface

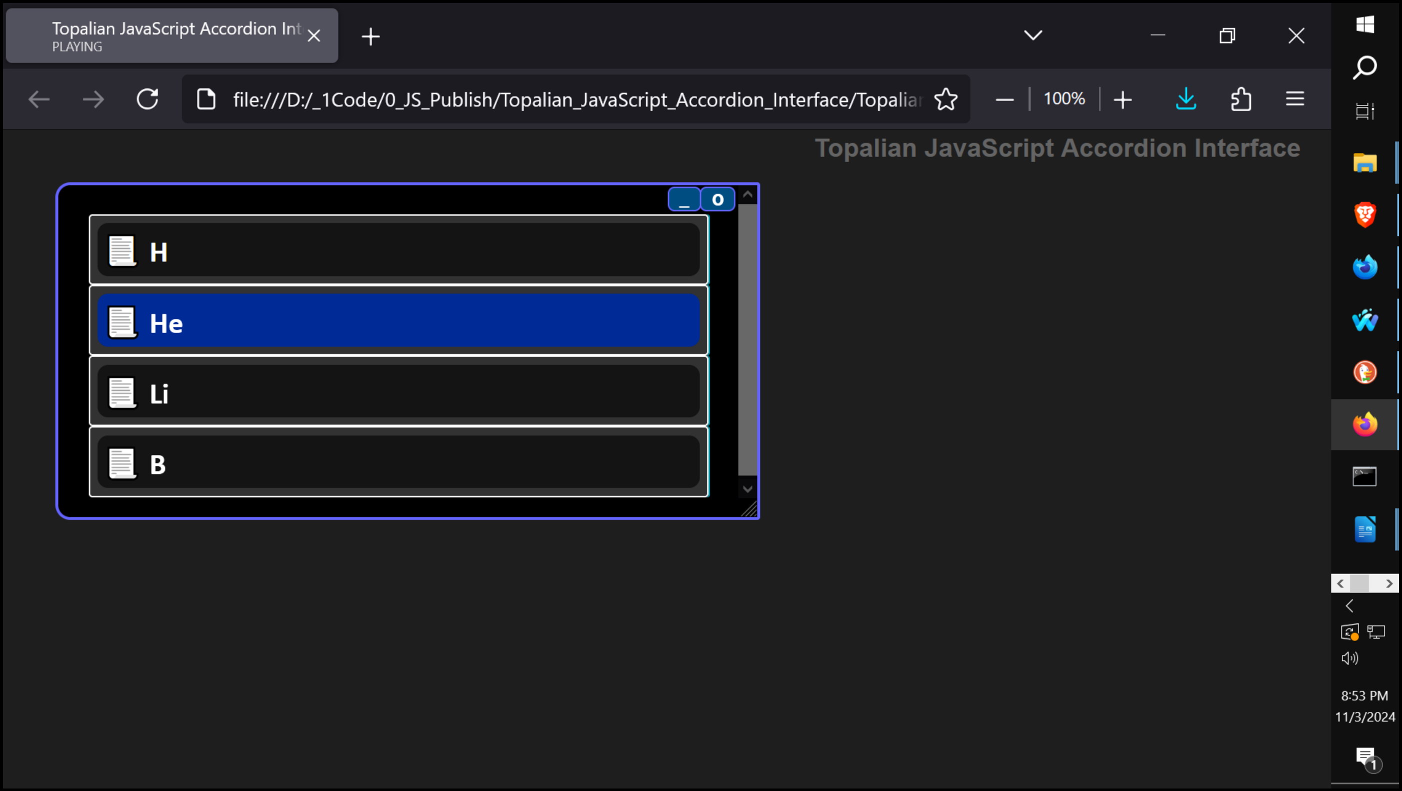

Topalian JavaScript Accordion Int
PLAYING
file:///D:/_1Code/0_JS_Publish/Topalian_JavaScript_Accordion_Interface/Topaliar
100%
Topalian JavaScript Accordion Interface
_
O
H
He
Li
B
8:52 PM
11/3/2024

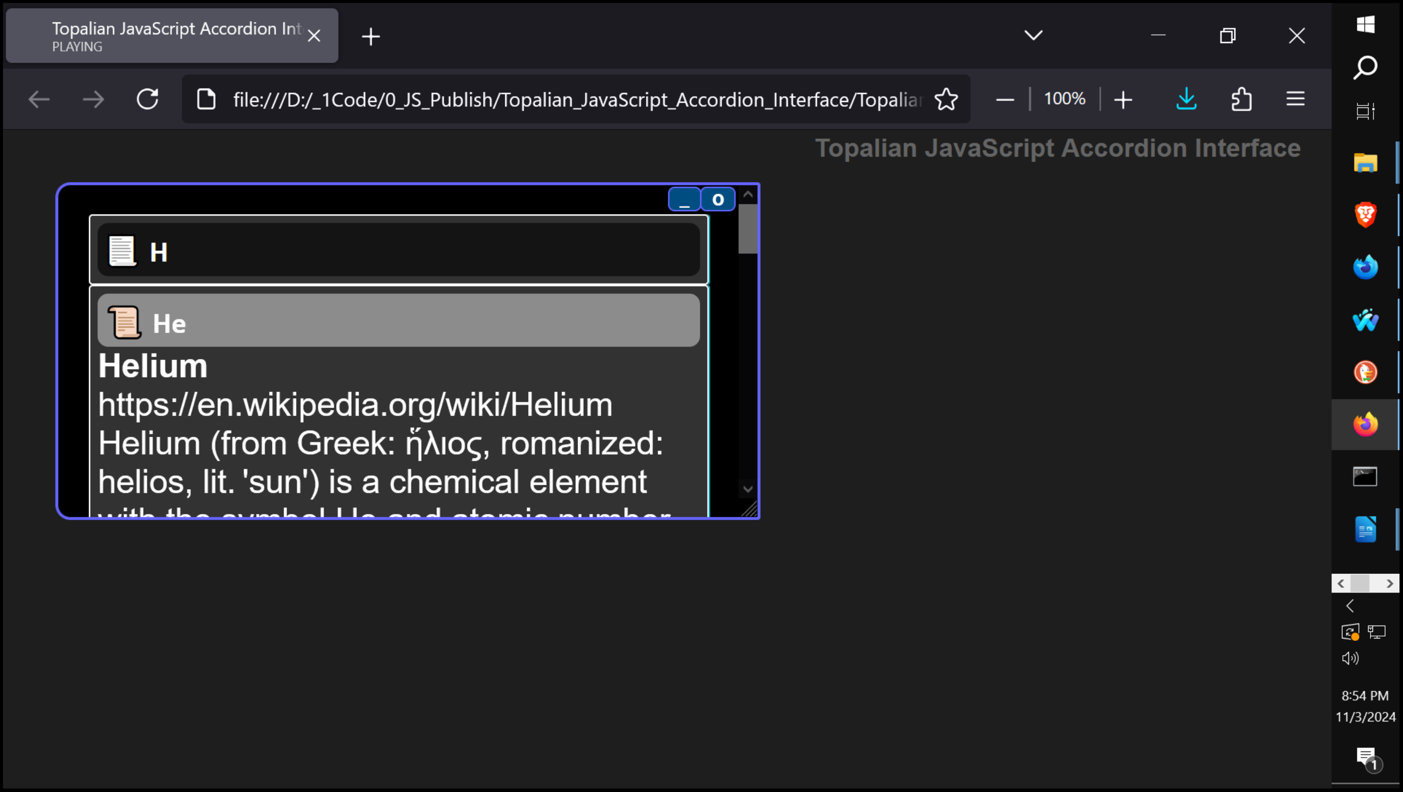

Topalian JavaScript Accordion Int
PLAYING
file:///D:/_1Code/0_JS_Publish/Topalian_JavaScript_Accordion_Interface/Topalia
100%
_
o
8:53 PM
11/3/2024

Topalian JavaScript Accordion Interface

H

Hydrogen
https://en.wikipedia.org/wiki/Hydrogen
Hydrogen is the chemical element with the symbol H and atomic number 1. Hydrogen is the lightest element. At standard conditions hydrogen is a gas of


Topalian JavaScript Accordion Int
PLAYING
file:///D:/_1Code/0_JS_Publish/Topalian_JavaScript_Accordion_Interface/Topalia
100%
Topalian JavaScript Accordion Interface
_
O
H
He
Li
B
8:53 PM
11/3/2024



Topalian JavaScript Accordion Int
PLAYING
file:///D:/_1Code/0_JS_Publish/Topalian_JavaScript_Accordion_Interface/Topalia
100%
8:54 PM
11/3/2024

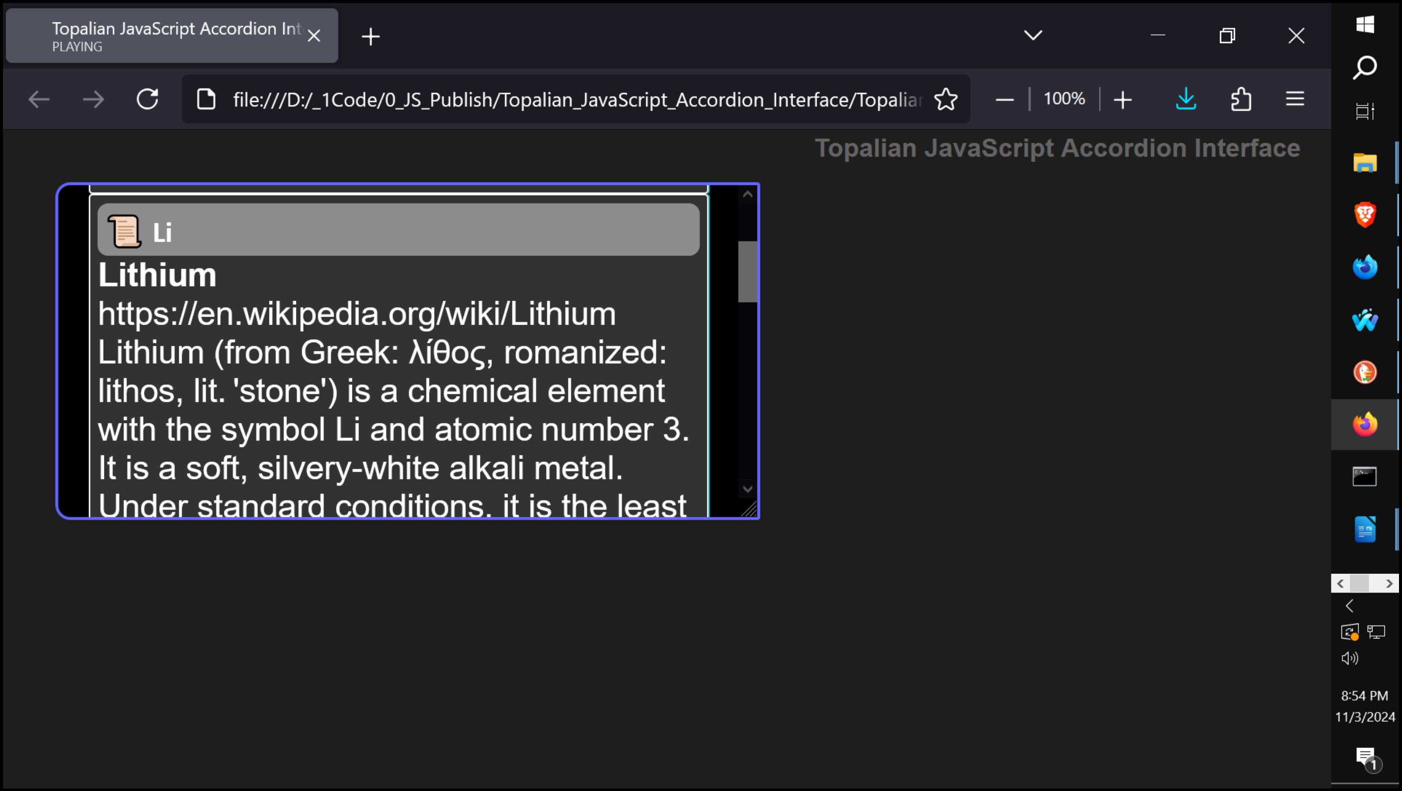
Topalian JavaScript Accordion Interface

_ o

📃 H

📜 He

Helium
https://en.wikipedia.org/wiki/Helium
Helium (from Greek: ἥλιος, romanized: helios, lit. 'sun') is a chemical element

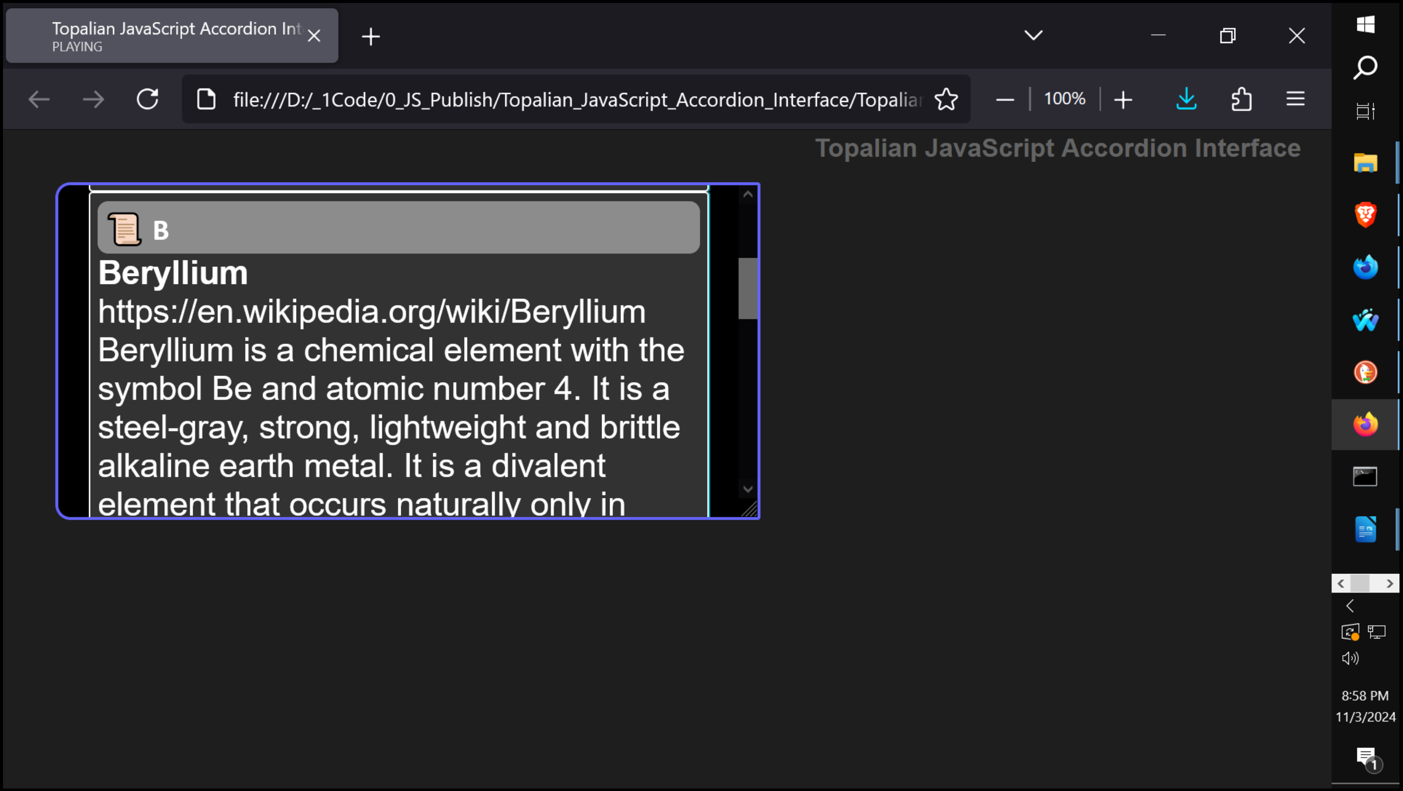


Topalian JavaScript Accordion Int
PLAYING
file:///D:/_1Code/0_JS_Publish/Topalian_JavaScript_Accordion_Interface/Topaliar
100%
8:54 PM
11/3/2024

Topalian JavaScript Accordion Interface

📜 Li

Lithium
https://en.wikipedia.org/wiki/Lithium
Lithium (from Greek: λίθος, romanized: lithos, lit. 'stone') is a chemical element with the symbol Li and atomic number 3. It is a soft, silvery-white alkali metal. Under standard conditions, it is the least



Topalian JavaScript Accordion Int
PLAYING
file:///D:/_1Code/0_JS_Publish/Topalian_JavaScript_Accordion_Interface/Topaliar
100%
8:58 PM
11/3/2024

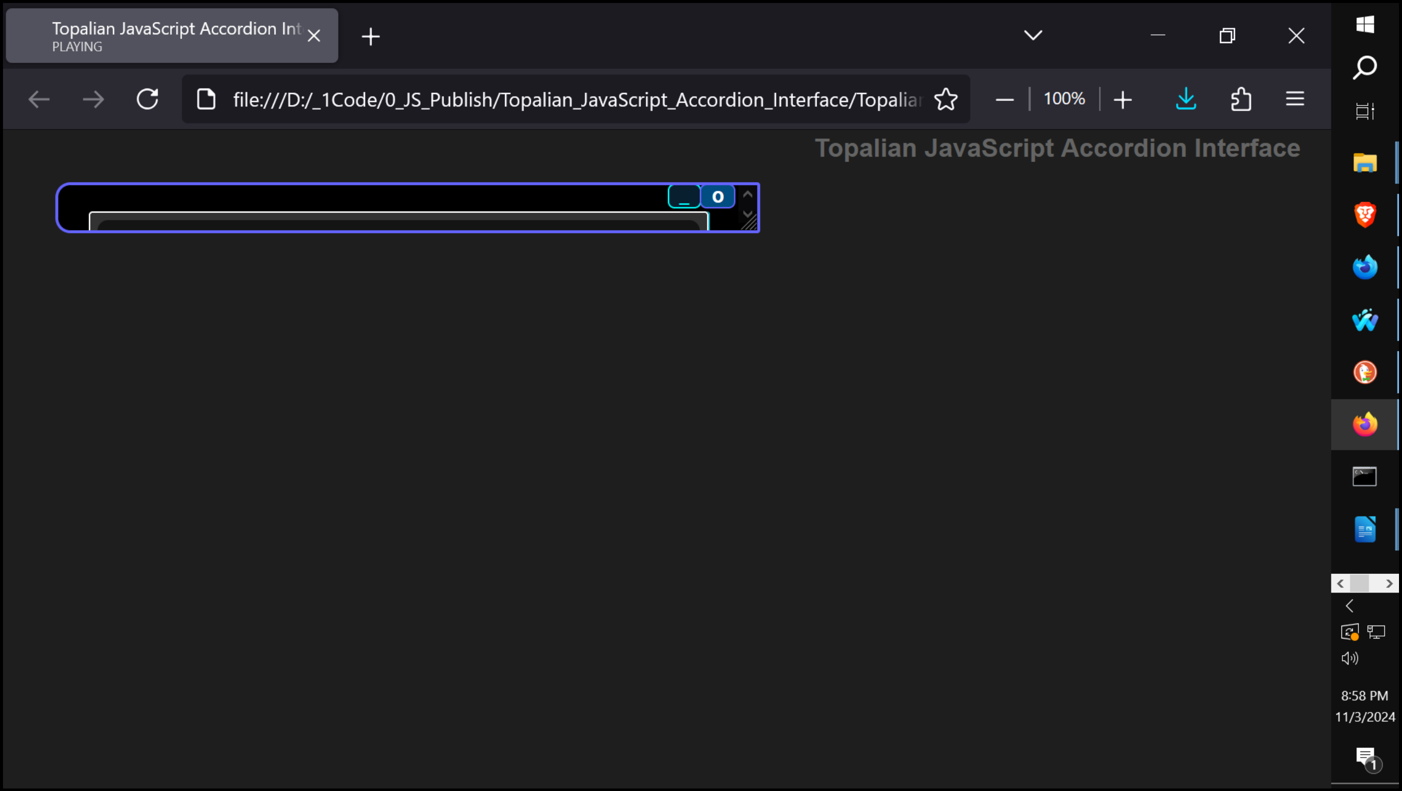
Topalian JavaScript Accordion Interface

B

Beryllium

https://en.wikipedia.org/wiki/Beryllium

Beryllium is a chemical element with the symbol Be and atomic number 4. It is a steel-gray, strong, lightweight and brittle alkaline earth metal. It is a divalent element that occurs naturally only in


Topalian JavaScript Accordion Int
PLAYING
file:///D:/_1Code/0_JS_Publish/Topalian_JavaScript_Accordion_Interface/Topaliar
100%
Topalian JavaScript Accordion Interface
_
0
8:58 PM
11/3/2024

```
<!-- Dedicated to God the Father -->
<!-- All Rights Reserved Christopher Andrew Topalian Copyright 2000-2024 -->
<!-- https://github.com/ChristopherTopalian -->
<!-- https://github.com/ChristopherAndrewTopalian -->
<!-- Topalian_JavaScript_Accordion_Interface_(Online_Version).html -->
<!-- Version 001 - (2024-11-04) -->

<html>
<head>
<title> Topalian JavaScript Accordion Interface </title>
```

```
<style>

/* style001.css */

body
{
    background-color: rgb(30, 30, 30);
    color: rgb(255, 255, 255);
    font-weight: bold;
}

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);
    cursor: pointer;
}

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

.details2 > summary {
     list-style-type: '📃 ';
}

.details2[open] > summary {
    list-style-type: '📜 ';
    background-color: rgb(140,140,140);
}
```

```
.details2
{
    width: 400px;
    /* width: 100%; */
    /* list-style-type: '▶'; */
    font: 20px Arial;
    color: white;
    background-color: rgb(50, 50, 50);
    border-style: solid;
    border-width: 1px;
    border-radius: 2px;
    padding-left: 4px;
    padding-right: 4px;
    box-shadow: 1px 0px 0px rgba(3, 226, 255, 0.685);
    cursor: pointer;
    overflow-wrap: anywhere;
    user-select: none;
}

.summary2
{
    border-radius: 7px;
    padding: 4px;
    font: 20px Calibri;
    text-align: left;
    font-weight: bold;
    background-color: rgb(20,20,20);
    color: rgb(255,255,255);
}

.summary2:hover
{
    background-color: rgb(0, 42, 149);
}
```

```
/* ---- */

/* 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 */

</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/CollegeOfScript
ing

// periodicTableArray.js

let periodicTableArray = [
    {
        name: "Hydrogen",
        abbreviation: "H",
        url:
"https://en.wikipedia.org/wiki/Hydrogen",
        description: "Hydrogen is the chemical
element with the symbol H and atomic
number 1. Hydrogen is the lightest element.
At standard conditions hydrogen is a gas of
diatomic molecules having the formula H2. It
is colorless, odorless, tasteless,[8] non-toxic,
and highly combustible. Hydrogen is the
most abundant chemical substance in the
universe, constituting roughly 75% of all
normal matter.[9][note 1] Stars such as the
Sun are mainly composed of hydrogen in the
plasma state. Most of the hydrogen on Earth
exists in molecular forms such as water and
organic compounds. For the most common
isotope of hydrogen (symbol 1H) each atom
has one proton, one electron, and no
neutrons. "
    },
    {
        name: "Helium",
        abbreviation: "He",
        url:
"https://en.wikipedia.org/wiki/Helium",
```

description: "Helium (from Greek: ἥλιος, romanized: helios, lit. 'sun') is a chemical element with the symbol He and atomic number 2. It is a colorless, odorless, tasteless, non-toxic, inert, monatomic gas, the first in the noble gas group in the periodic table.[a] Its boiling and melting point are the lowest among all the elements. Helium is the second lightest and second most abundant element in the observable universe (hydrogen is the lightest and most abundant). It is present at about 24% of the total elemental mass, which is more than 12 times the mass of all the heavier elements combined. Its abundance is similar to this in both the Sun and in Jupiter. This is due to the very high nuclear binding energy (per nucleon) of helium-4, with respect to the next

```
three elements after helium. This helium-4
binding energy also accounts for why it is a
product of both nuclear fusion and
radioactive decay. Most helium in the
universe is helium-4, the vast majority of
which was formed during the Big Bang. Large
amounts of new helium are being created by
nuclear fusion of hydrogen in stars. "
    },
    {
        name: "Lithium",
        abbreviation: "Li",
        url:
"https://en.wikipedia.org/wiki/Lithium",
        description: "Lithium (from Greek: λίθος,
romanized: lithos, lit. 'stone') is a chemical
element with the symbol Li and atomic
number 3. It is a soft, silvery-white alkali
```

metal. Under standard conditions, it is the least dense metal and the least dense solid element. Like all alkali metals, lithium is highly reactive and flammable, and must be stored in vacuum, inert atmosphere or inert liquid such as purified kerosene or mineral oil. When cut, it exhibits a metallic luster, but moist air corrodes it quickly to a dull silvery gray, then black tarnish. It never occurs freely in nature, but only in (usually ionic) compounds, such as pegmatitic minerals, which were once the main source of lithium. Due to its solubility as an ion, it is present in ocean water and is commonly obtained from brines. Lithium metal is isolated electrolytically from a mixture of lithium chloride and potassium chloride. "

    },

```
    {
        name: "Beryllium",
        abbreviation: "B",
        url:
"https://en.wikipedia.org/wiki/Beryllium",
        description: "Beryllium is a chemical
element with the symbol Be and atomic
number 4. It is a steel-gray, strong,
lightweight and brittle alkaline earth metal. It
is a divalent element that occurs naturally
only in combination with other elements to
form minerals. Notable gemstones high in
beryllium include beryl (aquamarine,
emerald) and chrysoberyl. It is a relatively
rare element in the universe, usually
occurring as a product of the spallation of
larger atomic nuclei that have collided with
cosmic rays."
```

```
    }
];

//--//

// 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(whichArray)
{
    let menuId = 'accordionMainDiv';

    if (ge(menuId))
    {
        removeElement(menuId);
    }

    //-//

    let mainDiv = ce("div");
    mainDiv.id = menuId;
    mainDiv.style.position = "absolute";
    mainDiv.style.left = 35 + 'px';
    mainDiv.style.top = 35 + 'px';
    mainDiv.style.height = 190 +'px';
    mainDiv.style.paddingLeft = "20px";
    mainDiv.style.paddingRight = "20px";
    mainDiv.style.paddingTop = "15px";
    mainDiv.style.paddingBottom = "15px";
    mainDiv.style.zIndex = "1004";
    mainDiv.style.background = "black";
    mainDiv.style.borderStyle = "solid";
    mainDiv.style.borderWidth = "2px";
    mainDiv.style.borderColor =
"rgba(100,100,255,1.0)";
    mainDiv.style.borderRadius= "10px";
    mainDiv.style.fontFamily = "arial";
    mainDiv.style.fontWeight = "bold";
    mainDiv.style.fontSize = "large";
    mainDiv.style.color = "white";
    mainDiv.style.textAlign = "left";
    mainDiv.style.overflowY = 'scroll';
    mainDiv.style.resize = 'both';
    ba(mainDiv);

    //-//

    makeElementDraggable(mainDiv);

    //-//

    let theMenu = ce("div");
    theMenu.style.marginBottom = 4 + 'px';
    theMenu.style.width = '200px';
    mainDiv.append(theMenu);

    //-//

    // minimize button
```

```
mainDiv.append(makeMinimizeButton(menuI
d));

    //-//

    // maximize button

mainDiv.append(makeMaximizeButton(menuI
d));

    //-//

    for(let x = 0; x < whichArray.length; x++)
    {
        let theDetails = ce("details");
        theDetails.innerHTML =
whichArray[x].name.bold();
```

```
        theDetails.className = "details2";
        theDetails.style.padding = "5px 5px 5px
5px";
        theDetails.style.fontSize = 22 + 'px';
        mainDiv.append(theDetails);

        //-//

        let theSummary = ce("summary");
        theSummary.innerHTML =
whichArray[x].abbreviation;
        theSummary.className = "summary2";
        theSummary.onmouseover = function()
        {
            hoverSound();
        };
        theSummary.onclick = function()
        {
            clickSound();
        };
        theDetails.append(theSummary);

        //-//

        let theUrl = ce("div");
        theUrl.innerHTML = whichArray[x].url;
        theDetails.append(theUrl);

        //-//

        let theDescription = ce("div");
        theDescription.innerHTML =
whichArray[x].description;
        theDetails.append(theDescription);
    }
}
```

```
//--//

// 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

// makeMaximizeButton.js

function makeMaximizeButton(whichMenuId)
{
    let maximizeButton = ce("button");
    maximizeButton.style.position =
"absolute";
    maximizeButton.style.top = "0px";
    maximizeButton.style.right = "0px";
    maximizeButton.textContent = 'O';
    maximizeButton.style.cursor = 'pointer';
    maximizeButton.style.fontSize = '10px';
    maximizeButton.onmouseover = function()
    {
        hoverSound();
    };
    maximizeButton.onclick = function()
    {
        clickSound();
        ge(whichMenuId).style.height = 190 +
'px';
    };
    return maximizeButton;
}

//--//

// 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

// makeMinimizeButton.js
```

```
function makeMinimizeButton(whichMenuId)
{
    let minimizeButton = ce("button");
    minimizeButton.style.marginRight = '10px';
    minimizeButton.style.position =
"absolute";
    minimizeButton.style.top = "0px";
    minimizeButton.style.right = "15px";
    minimizeButton.textContent = '_';
    minimizeButton.style.cursor = 'pointer';
    minimizeButton.style.fontSize = '10px';
    minimizeButton.onmouseover = function()
    {
        hoverSound();
    };
    minimizeButton.onclick = function()
    {
        clickSound();
        ge(whichMenuId).style.height = 0 + 'px';
    };

    return minimizeButton;
}

//--//

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

    //-//

    let titleOfApp = ce('a');
    titleOfApp.id = 'titleOfApp';
    titleOfApp.textContent = 'Topalian JavaScript Accordion Interface';
    titleOfApp.href = 'https://github.com/christophertopalian/';
    titleOfApp.target = '_blank';
    titleOfApp.style.fontSize = '17px';
    titleOfApp.style.fontWeight = 'bold';
    titleOfApp.style.textDecoration = 'none';
    titleOfApp.style.color = 'rgb(100, 100, 100)';
    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/CollegeOfScripting

// 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(vol
ume, 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
        400,     // duration
        'sine',  // waveType
        0.03,    // volume
        -500,    // detune
        0.1,     // attackTime
        0.1,     // releaseTime
        0.0      // 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/CollegeOfScripting

// makeElementDraggable.js

// creating event handler for mouse down
function createMouseDownHandler(element, state)
{
    return function(e)
    {
        // ignore mouse down events on textarea
or input elements
        if (e.target.tagName === 'TEXTAREA' ||
e.target.tagName === 'INPUT')
        {
            return;
        }

        e.preventDefault();

        state.startX = e.clientX;
        state.startY = e.clientY;

        document.onmousemove =
createMouseMoveHandler(element, state);

        document.onmouseup =
createMouseUpHandler();
```

```
    };
}

// creating event handler for mouse move
function createMouseMoveHandler(element,
state) {
    return function(e)
    {
        // ignore mouse move events on textarea
or input elements
        if (e.target.tagName === 'TEXTAREA' ||
e.target.tagName === 'INPUT')
        {
            return;
        }

        e.preventDefault();
        let deltaX = state.startX - e.clientX;
        let deltaY = state.startY - e.clientY;

        state.startX = e.clientX;
        state.startY = e.clientY;

        let newTop = element.offsetTop - deltaY;
        let newLeft = element.offsetLeft - deltaX;

        element.style.top = newTop + "px";
        element.style.left = newLeft + "px";
    };
}

// creating event handler for mouse up
function createMouseUpHandler()
{
    return function()
    {
        document.onmousemove = null;
        document.onmouseup = null;
    };
}

// make an element draggable
function makeElementDraggable(element)
{
    let state = { startX: 0, startY: 0 };

    element.onmousedown =
createMouseDownHandler(element, state);
}

//--//

// 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

// whenLoaded.js

function whenLoaded()
{
    makeInterface(periodicTableArray);

    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/CollegeOfScripting

</script>

</head>
```

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

</body>

</html>
```

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

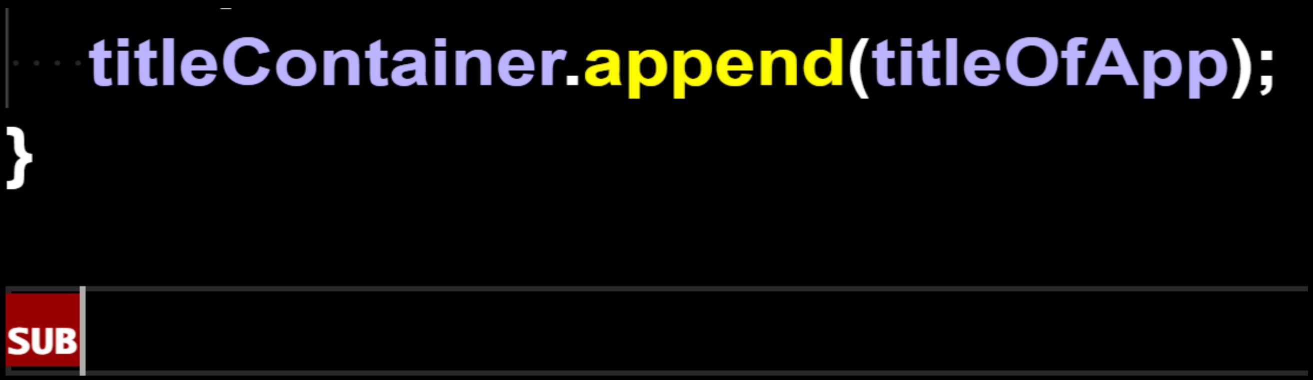
First, we add two new lines at the end of every script. This way they will later combine nicely with a line break in between each script.

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

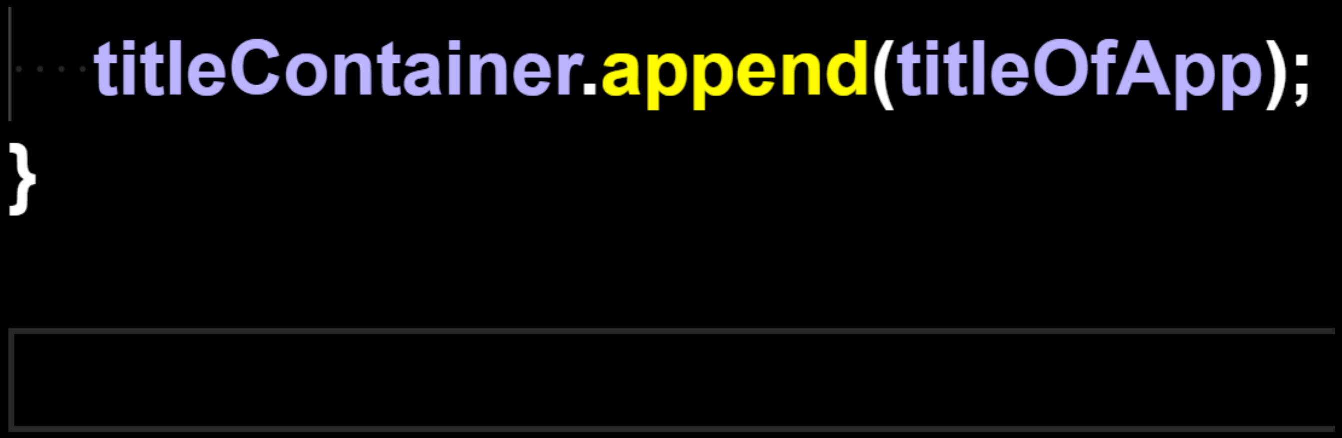
This creates a new file that is named main.js This new file contains all .js files in ONE file. But, there is a junk character at the end of the main.js script that we have to delete. In VSCode the character might be called SUB



```
    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.

// HowToCombineJSFilesAllFolders.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.

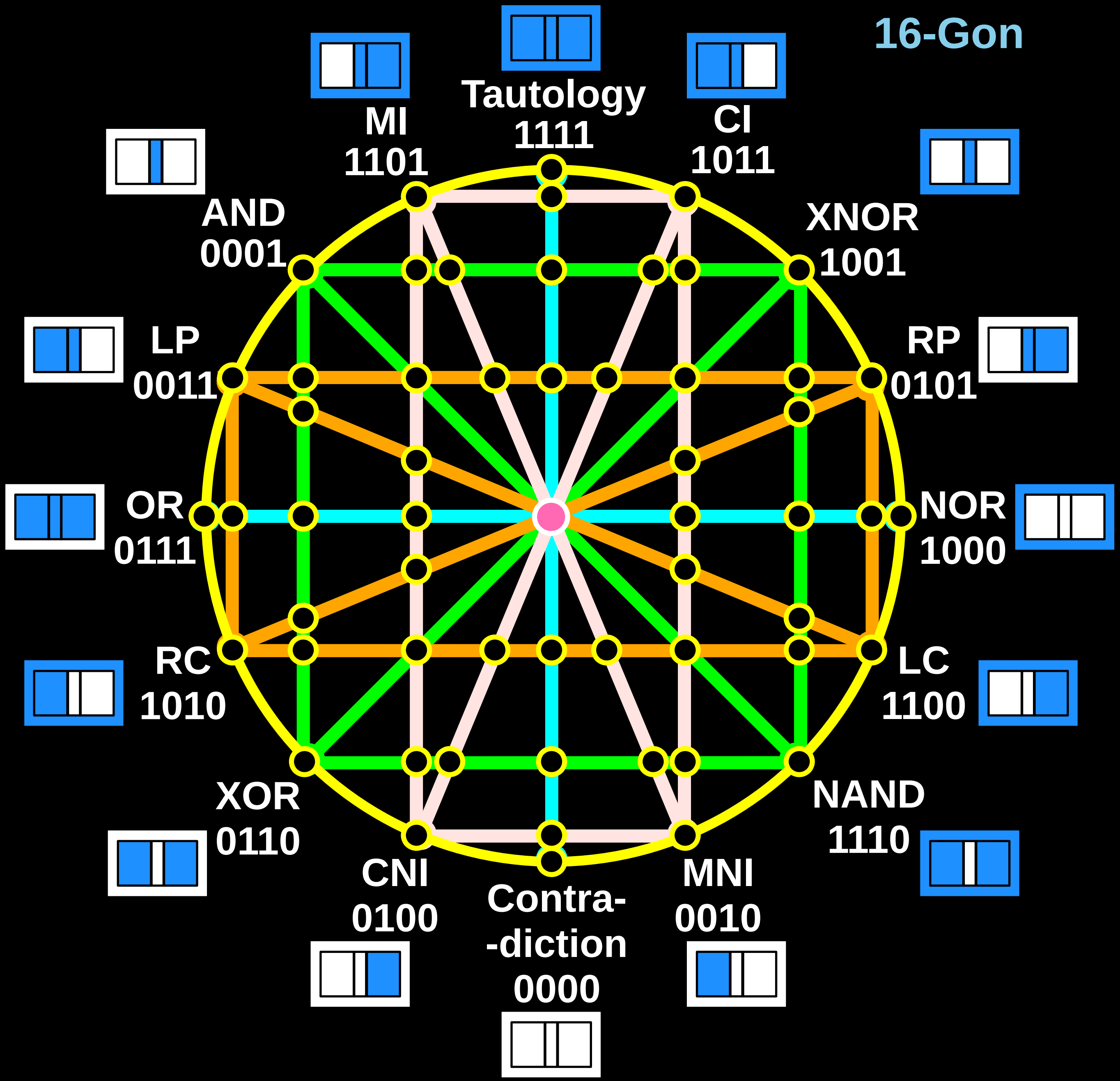
Remember too, that not all file types will combine, but the ones above will.

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

Christopher Topalian

True Artificial Intelligence System

16-Gon



College of Scripting Music & Science

CollegeOfScripting.weebly.com

For More Tutorials:

GitHub.com/ChristopherTopalian

GitHub.com/ChristopherAndrewTopalian

Sites.google.com/view/CollegeOfScripting

CollegeOfScripting.weebly.com

CollegeOfScripting.wordpress.com

Youtube.com/ScriptingCollege

Twitter.com/CollegeOfScript

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