

Church of Scyence

Software Learning JS



October 18, 2023

Church of scyence

https://github.com/churchofscyence

Table of Contents

[Introduction 2](#_Toc152068032)

[Starting Learning JavaScript Application 2](#_Toc152068033)

[Start the Application with Angular CLI 2](#_Toc152068034)

[Start the Application with the Docker Script 3](#_Toc152068035)

[Application Guide 4](#_Toc152068036)

[Navigation Bar 4](#_Toc152068037)

[Initialize Arrays Interface 5](#_Toc152068038)

[Element to the Array 5](#_Toc152068039)

[Remove the Element from the Array 6](#_Toc152068040)

[Change the arrays to be used 6](#_Toc152068041)

[Code Editor Interface 7](#_Toc152068042)

[Using the Code Editor 7](#_Toc152068043)

[Array Result Types 7](#_Toc152068044)

[Application Coding 7](#_Toc152068045)

[References 8](#_Toc152068046)

# Introduction

Members of the church used the Google Angular framework to build this software package called Learning JavaScript. No database exists for this application, so your coding in the Code Editor will not be saved. Learning JavaScript is meant to be used at all skill levels, from high school to college. We wrote this software package so new developers can practice JavaScript. Students are encouraged to take notes to author their tutorials to understand the language better. No tutorial will be provided with this application. To start this application, use the Docker Script or Angular command-line interface (CLI). The student must install Angular CLI and NodeJS on the command line to start the application with the CLI.  It is recommended that the user install an editor such as Microsoft Visual Studio Code or IntelliJ Jet Brains WebStorn. To use the Docker Script, the user must install the Docker Desktop.

A screenshot of a computer

Description automatically generated

# Starting Learning JavaScript Application

## Start the Application with Angular CLI

The first step is to install Node.js, which will also install the Node Package Manager (NPM). There are instructions in the Reference Section of this guide. The students can find all the tutorials on the Church of Scynce GitHub site in the Script Repository. The next step is to populate the node modules folder by running the ***npm install*** command from the Windows Command Line or Mac Terminal Windows. Node Package Manager reads the angular JSON file to download all the necessary libraries.

A screenshot of a computer

Description automatically generated

The ***ng service*** command builds, deploys, serves, and watches your angular code changes. Angular CLI runs Webpack to build and bundle all JavaScript and CSS code. In turn, Webpack calls the TypeScript loaders, which fetches all .ts files in the Angular project and then transpiles them to JavaScript files, which browsers can understand.

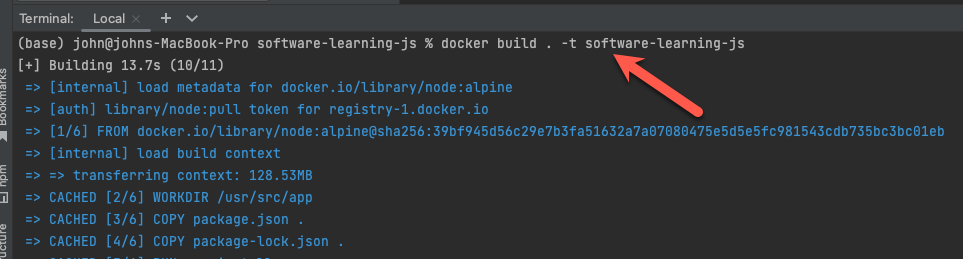
A screenshot of a computer

Description automatically generated

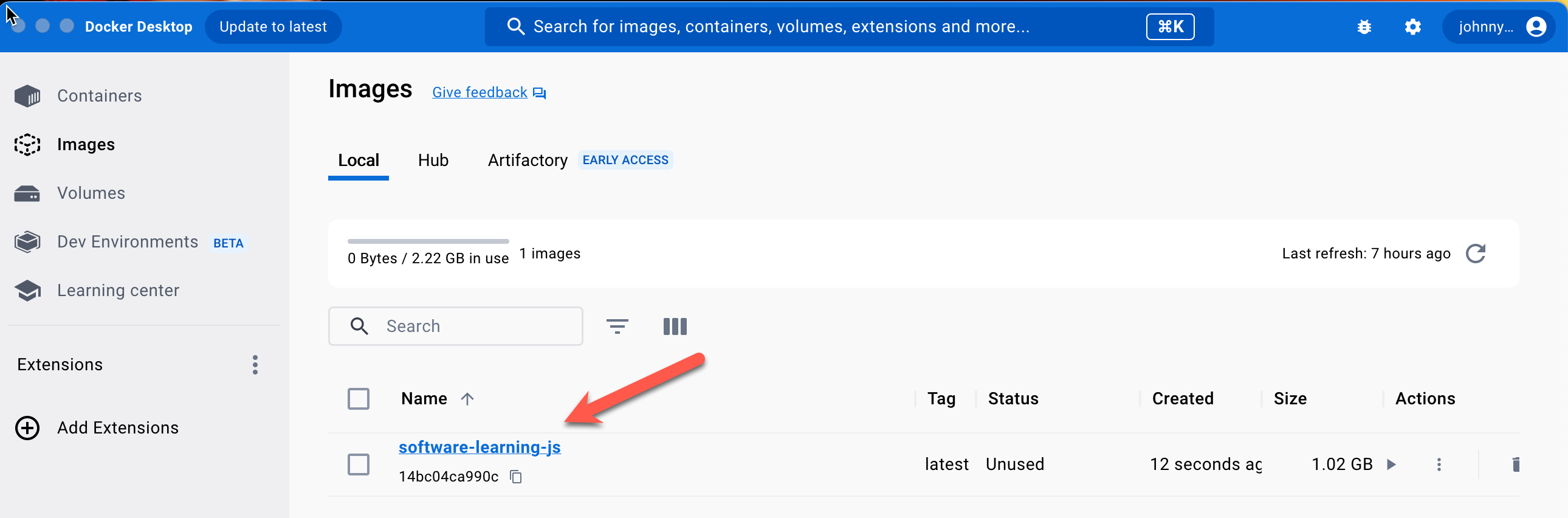
## Start the Application with the Docker Script

First, the user must build a Docker image. A Docker image is a read-only template that contains instructions for creating a container that can run on the Docker platform. It provides a convenient way to package up applications and preconfigured server environments, which you can use for pre-use or share publicly with other Docker users. Docker images are also the starting point for anyone using Docker for the first time. From the command line, run the following command.

$ docker build . -t sofware-learning-js



In the Docker Desktop, you can see that the image was created. You can see the size of the image. It should be approximately one gigabyte. The name of the image is software-learning-js. The users can also check the creation time to verify that the image was completed successfully.



The Next step is to create a container from the image. A Docker container image is a lightweight, standalone, executable package of software that includes everything needed to run an application: code, runtime, system tools, system libraries, and settings.

$ docker run -p 4200:4200 software-learning-js

A screen shot of a computer

Description automatically generated

You can go into Docker Desktop to access the container. After creating the container, the user can open the webpage browser and navigate to the following URL.

http://localhost:4200/

A screenshot of a computer

Description automatically generated

# Application Guide

Suppose you want to change the ***concatenate*** from the ***Method Calls*** to the ***Function Calls*** navigation grouping. In that case, you can go to the Application Coding section of this document to rearrange the methods grouping. You can also create new navigation groups for each method. I picked the navigation groups for this application because I just tried to divide them up equally so it would be easier for the user to find the different methods. It's a good training exercise for the user to review all the various methods and decide how to group the JavaScript Methods.

## Navigation Bar

You can select the part of the JavaScript language you want to practice from the navigation panel.  The user can choose an **Array, *String,*** or ***Regular Expression*.** The user can choose from two navigation menus: the traditional Dropdown Links or Bread Crumb Menu. If users read left from right, they would select Bread Crumb Navigation.

A screenshot of a computer

Description automatically generated

Traditional Dropdown Links

The traditional dropdown link would be a better choice if the user is used to reading from top to bottom. This menu also allows you to quickly search through the different methods to search for the one you're looking for.

A screenshot of a computer

Description automatically generated

Bread Crumb Menu

## Initialize Arrays Interface

When I decided on the Navigation Bar Structure for the array, I grouped the methods into **Method Calls**, **Data Structures**, and **Function Calls.** The Method Calls are single-line methods with object and method parameters. In the Data Structures group, I put all queries and stacks or methods used for the looping through an array. The **Function Calls** are functions that are passed in the methods parameter list.

### Element to the Array

To add an element to the first you, you must choose by selecting the array from the Choose Table Drop Down Box. Next, type the element in the Netow Element Text Box. For example, I would like Fruits from the Choose Table to add Strawberries to the fruits array. Then, type Strawberry in the **New Element Text Box**. Finally, click the **Add Button** below the Add Element Form.

A screenshot of a computer

Description automatically generated

The user will see the word **Strawberry** appear in the fruit array's last element.

A screenshot of a computer screen

Description automatically generated

### Remove the Element from the Array

The user deletes an element from an array by clicking the **Remove Button** in the **Action Column** table. For example, if the user wants to remove the third Element, Apple, from the table, click the Remove Button in the same row as the Apple row.

A screenshot of a computer

Description automatically generated

After clicking the **Remove Button**, the user will see the Apple disappear from the third row.

A screenshot of a computer

Description automatically generated

### Change the arrays to be used

If a user wants to use a different array in the Code Editor, you can select which in the **Select Array Form**. For example, the users wish to use the Unique Integer instead of the Peoples array. In the **Choose Table Drop dropdown box**, select People Array. Then select Unique Integer in the **Array List dropdow**n. Finally, click the **Select Button**.

A screenshot of a computer

Description automatically generated

After clicking the Select Button, you can see the two arrays the user can use in the Code Editor: fruits and unique integers.

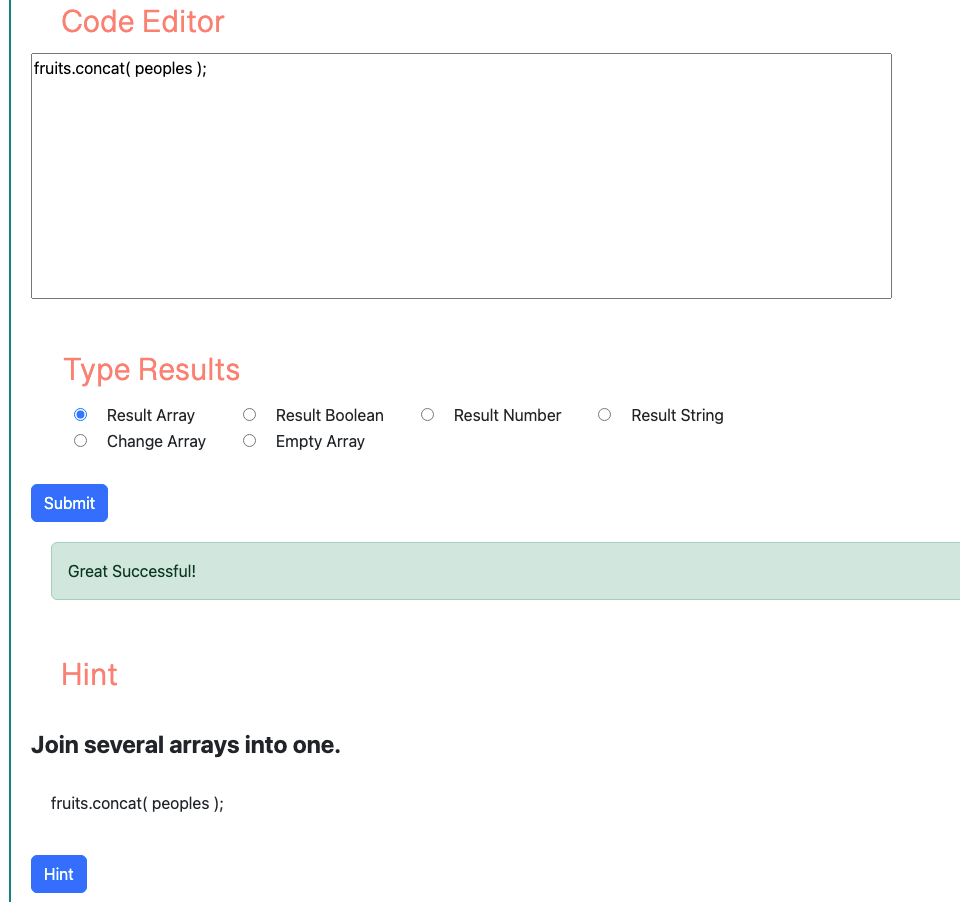
A screenshot of a computer

Description automatically generated

## Code Editor Interface

### Using the Code Editor

When designing this application, I only focused on the most essential code to memorize. Assignment statements are declared in the Initial Section of the application, so the user should not declare them in the Code Editor. The user can click the Hint Button to see the code, which can be copied and pasted into the Code Editor.



### Array Result Types

Before the user clicks the Submit Button, the user must select Type Results.

* ***Result Array*** - This method type returns a new array. For example, if the user wants to concatenate the fruits and people. The concat method needs to be assigned to a variable.
* ***Result Boolean*** - The method type returns a true or false value. For example, if the user wants to check if Mango is in the fruits array. The includes method would return true.
* ***Result Number*** - This method type returns the index number of the array. For example, the user can use the indexOf method to find the index number of the Mango element in the fruits array. This method would return 4.
* ***Result String*** - This method type returns a string. For example, if the user wants to convert an Array to a String, the user can use the join method.
* ***Change Array*** - This method type changes the existing array. For example, the user can use the Fill Method to fill the array with the array with the same element.
* ***Empty Array*** - This method type uses an empty array with a push method to store elements in an Empty array. One example of using the Empty Array is when using the Entries Method. In this method, you can use the for loop to iterate over an array.
* ***Exec Method -*** This result type is for the Regular Expression Exec Method. The method returns an array with index, input, and groups as element names.
* ***Replace Method -*** This is used with the Regular Expression where the matching characters are represented by "X" in the String.
* ***Reg Expr Return Boolean -*** Regular Expression Test Method returns a Boolean value.
* ***Match All Method -*** For Regular Expression Match All Method
* ***Replace All Method -*** For Regular Expression Replace All Method

# Application Coding

## File Locations

The data for each page is stored in a JavaScript Map containing a Context Class. Each page URL includes the ***Method Category*** Array, String, or Regular Expression. The last part of the URL is the ***Method Name***.

A screenshot of a phone

Description automatically generated

The set environment method gets called when the webpage first loads up. The application can select the correct data service by knowing the Method Category.

A screenshot of a computer

Description automatically generated

The following JavaScript Map populates the Concatenate Method webpage.

A screen shot of a computer program

Description automatically generated

All the business logic for the Submit Button is in the Results Class. Three components could populate the Type Results area within the bottom component in the inner directory based on the Method Category.

A screenshot of a computer

Description automatically generated

## Change the Navigation Method Grouping

# References

* [How to Install Node.js on Window 10](https://www.youtube.com/watch?v=__7eOCxJyow)
* [How to install Nodejs on Mac [Any version & Easy method]](https://www.youtube.com/watch?v=SwUKKCS3r3c)
* [Running a project with ng serve](https://www.youtube.com/watch?v=-w-RfHcLt5U)
* [How To Install Docker on Windows 11](https://www.youtube.com/watch?v=WDEdRmTCSs8)
* [Docker Desktop for macOS Setup and Tips](https://www.youtube.com/watch?v=gcacQ29AjOo)
* [JavaScript Tutorial for Beginners: Learn JavaScript in 1 Hour](JavaScript%20Thttps:/www.youtube.com/watch?v=W6NZfCO5SIk)
* [Regular Expressions (RgeEx) Tutorial -Net Ninja](https://www.youtube.com/playlist?list=PL4cUxeGkcC9g6m_6Sld9Q4jzqdqHd2HiD)