LinkedIn Learning 6. JavaScript Essential Training

1. JavaScript Introduction

JSX: A syntax extension of JavaScript created for React JavaScript framework.

Components: Objects, Methods, Functions, Template Literals, Arrays

JavaScript: The core language sometimes referred to as vanilla JavaScript or Vanilla JS.

ECMAScript: The browser specification of the JavaScript language

Formatting and Linting

    // single line comment   /* Multi-line */

- `<script>` in HTML Doc `</script>` tags after element rendered
- `<script src=" ">` external file `</script>`

async: Browser downloads JavaScript in parallel while HTML renders when JavaScript is fully loaded, rendering stops while JS is executed. `<script src=" " async></script>`

defer = Browser downloads JS in parallel while HTML renders, then defers execution of JS until HTML rendering is complete `<script src=" " defer></script>`

JS Modules

several files `<script type="modules" src=" "></script>`
import in script.Js file, export in separate Js file.

3. Objects

Object : JavaScript objects are collections of data and functionality stored as properties and methods that describe the object.
   i. variable holds data

2. Const backpack = {}; Curly brackets define data as an object.

3. properties defined using key—value pairs. (: ,)

4. Methods are properties containing functions

"this" keyword refers to the current object.

properties   name 🐷 "something" ☁

             ↑   ↑

            key   value

Template Literals

(`` ` ` ``)   Const   something = ` < ></ >
      ∟ insert HTML code.   < ></ >
                               < > </> `

document. body. innerHTML = something ;

use $ { } to replace the current HTML code.

to get the value from Java Script
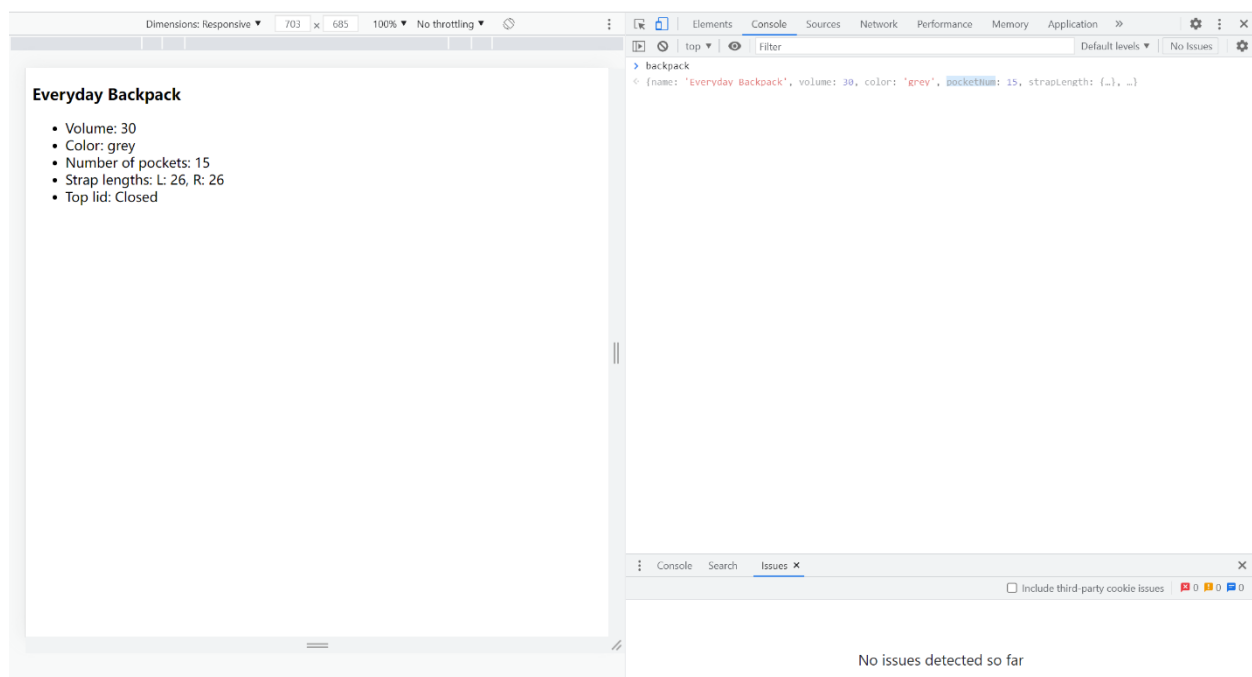
traditional string output

const   something = "<h1>" + xxx. xxx + "</h1>" ;

4a. After studying this course, I know the basic rules and how to write JavaScript code. For example, what are Objects, Properties and Template Literals.

4b. There are many ways to link HTML files and JS files to achieve more and better functions. In the web development process, try to choose the best and fastest way to write JS code, which can save more time and make the code look simpler and clearer, which is more helpful for teamwork.
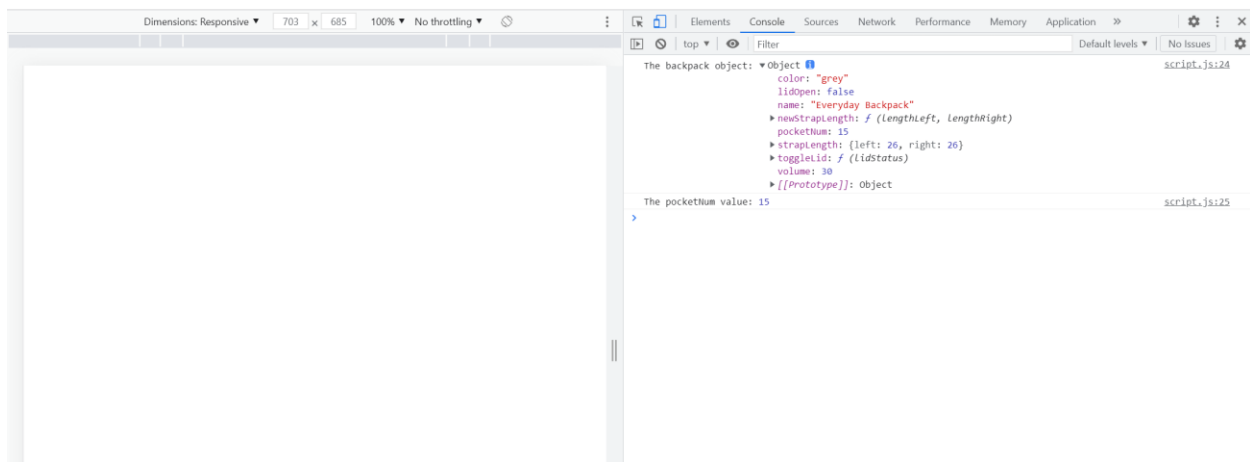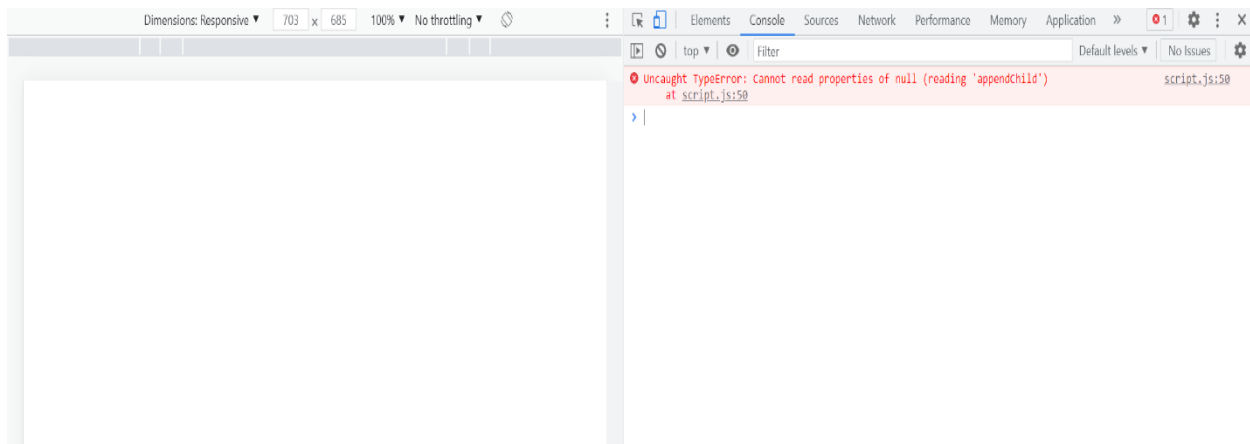
4c. Similarly, I still recommend that all students study this course well. For students with some experience in Java code, this course is not difficult and easy to learn.
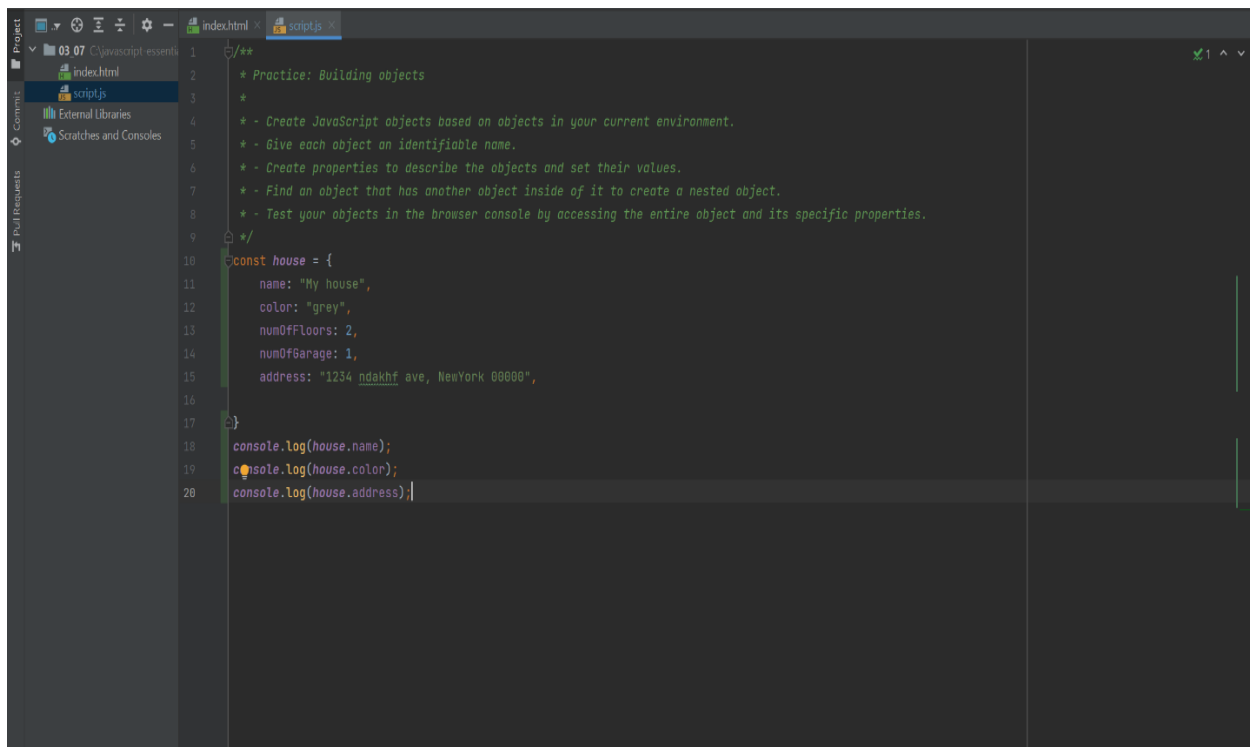
4d. This course took me 2 hours to compete. (Including organizing notes and screenshots.)

Everyday Backpack

- Volume: 30
- Color: grey
- Number of pockets: 15
- Strap lengths: L: 26, R: 26
- Top lid: Closed
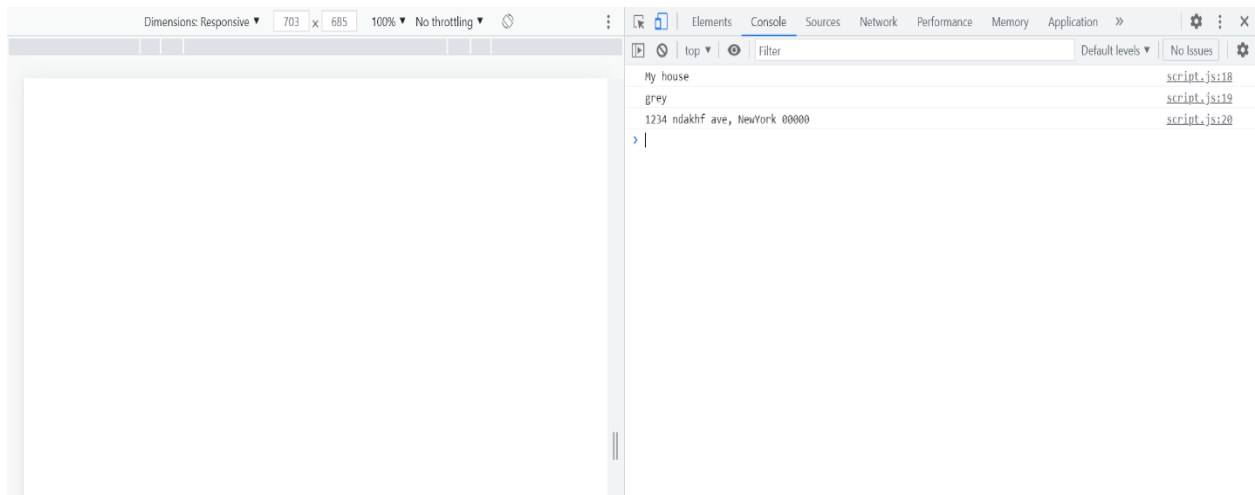
```javascript
 2      🔶 Create a Backpack object, populate some HTML to display its properties.
 3      */
 4
 5      // Single line comment
 6
 7      /* Multi-line comment
 8      See! this line is also commented out! */
 9
10      const updateBackpack = (update) => {
11          let main = document.querySelector( selectors: "main"); // main is an element
12          main.innerHTML = markup(backpack);
13          console.info(update);
14      };
15
16      const backpack = {
17          name: "Everyday Backpack",
18          volume: 30,
19          color: "grey",
20          pocketNum: 15,
21          strapLength: {
22              left: 26,
23              right: 26,
24          },
25          lidOpen: false,
26          toggleLid: function (lidStatus) {
27              this.lidOpen = lidStatus;
28              updateBackpack( update: `Lid status changed.`);
29          },
30          newStrapLength: function (lengthLeft, lengthRight) {
31              this.strapLength.left = lengthLeft;
32              this.strapLength.right = lengthRight;
33              updateBackpack( update: `Strap lengths updated.`);
```

Dimensions: Responsive ▼   703 x 685   100% ▼ No throttling ▼   ⊘   ⋮

Elements Console Sources Network Performance Memory Application »   ⊗1 ✿ ⋮ ✕

▶ ⊘ top ▼ ⊙ Filter   Default levels ▼   No Issues ✿

⊗ Uncaught TypeError: Cannot read properties of null (reading 'appendChild')   script.js:50
    at script.js:50

›

Dimensions: Responsive ▼   703 x 685   100% ▼ No throttling ▼   ⊘   ⋮

Elements Console Sources Network Performance Memory Application »   ✿ ⋮ ✕

▶ ⊘ top ▼ ⊙ Filter   Default levels ▼   No Issues ✿

The backpack object: ▼ Object 🛈   script.js:24
                        color: "grey"
                        lidOpen: false
                        name: "Everyday Backpack"
                      ▶ newStrapLength: ƒ (lengthLeft, lengthRight)
                        pocketNum: 15
                      ▶ strapLength: {left: 26, right: 26}
                      ▶ toggleLid: ƒ (lidStatus)
                        volume: 30
                      ▶ [[Prototype]]: Object
The pocketNum value: 15   script.js:25

›

```javascript
/**
 * Practice: Building objects
 *
 * - Create JavaScript objects based on objects in your current environment.
 * - Give each object an identifiable name.
 * - Create properties to describe the objects and set their values.
 * - Find an object that has another object inside of it to create a nested object.
 * - Test your objects in the browser console by accessing the entire object and its specific properties.
 */
const house = {
    name: "My house",
    color: "grey",
    numOfFloors: 2,
    numOfGarage: 1,
    address: "1234 ndakhf ave, NewYork 00000",

}
console.log(house.name);
console.log(house.color);
console.log(house.address);
```

Console output:

```
My house                                          script.js:18
grey                                              script.js:19
1234 ndakhf ave, NewYork 00000                    script.js:20
>
```

```javascript
/**
 * Practice: Making methods
 *
 * - Create a method for each object property.
 * - The method receives a value to match the property to be changed.
 * - Create a simple function to replace the current property value with the received value.
 * - Test the method by sending new values and checking the properties in the console.
 */

const backpack = {
  name: "Everyday Backpack",
  volume: 30,
  color: "grey",
  pocketNum: 15,
  strapLength: {
    left: 26,
    right: 26,
  },
  toggleLid: function (lidStatus) {
    this.lidOpen = lidStatus;
  },
  newStrapLength: function (lengthLeft, lengthRight) {
    this.strapLength.left = lengthLeft;
    this.strapLength.right = lengthRight;
  },

  newPocketNum: function (pockets) {
    this.pocketNum = pockets;
  }
};
console.log("old number", backpack.pocketNum);
backpack.newPocketNum( pockets: 28);
console.log("new Number", backpack.pocketNum);
```

Console output:

```
old number 15                          script.js:31
new Number 28                          script.js:33
>
```