Define Classes and Methods in JavaScript Labs

Perform these labs on your own computer using Visual Studio 2022 or later, or VS Code 1.8x or later, to ensure you understand the lessons presented in the corresponding videos and lectures.

Lab 1: Create an RV Class

Create a file named **class**.**html** and add the following HTML into the new file.

```
<!DOCTYPE html>
<html>

<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width" />

  <title>Create an RV Class</title>

  <link href="styles/site.css" rel="stylesheet" />
</head>

<body>
  <header>
    <h1>Create an RV Class</h1>
  </header>

  <main>
    <p>Open your F12 Browser Tools to view the
results.</p>
  </main>

  <script>
    'use strict';

    // Create a class named RV
```

```
    // DO NOT USE a constructor


    // Create an instance of an RV class


    // Set all the properties
    // of the RV object


    // Display the RV object
    // in the console window

  </script>
</body>

</html>
```

Write the appropriate code to create a class named **RV**. Without using a constructor, define four properties; *rvid*, *year*, *make*, and *model*.

Create an instance of the RV class and put it into a variable named *rv*. Set the properties to the following values.

- rvid = 1
- year = 2025
- make = "Newmar"
- model = "New Aire"

Display the *rv* variable in the console window.

Try It Out

Display the **class.html** file in your browser and your page should look like Figure 1.
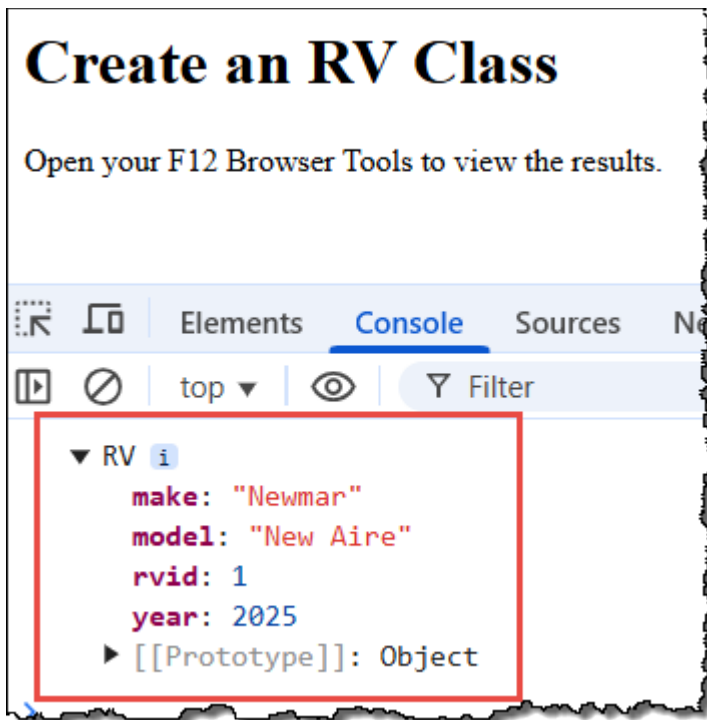
Figure 1: Create an RV class with appropriate properties.

Lab 2: Initialize Properties in Constructor

Create a file named **class-init**.**html** and add the following HTML into the new file.

```
<!DOCTYPE html>
<html>

<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width" />

  <title>Initialize Properties in Constructor</title>

  <link href="styles/site.css" rel="stylesheet" />
</head>

<body>
  <header>
    <h1>Initialize Properties in Constructor</h1>
  </header>
```

```
  <main>
    <p>Open your F12 Browser Tools to view the
results.</p>
  </main>

  <script>
    'use strict';

    // Create a class named RV


    // Create an instance of an RV class



    // Display the RV object
    // with default in the console window

  </script>
</body>

</html>
```

Write the appropriate code to create a class named **RV**. Define four properties; *rvid*, *year*, *make*, and *model* in the class.

Create a **constructor** in which you initialize the four properties to the following values.

- rvid = 0
- year = The current year
- make = an empty string
- model = an empty string

Create an instance of the RV class and put it into a variable named *rv*.

Display the *rv* variable in the console window.

Try It Out

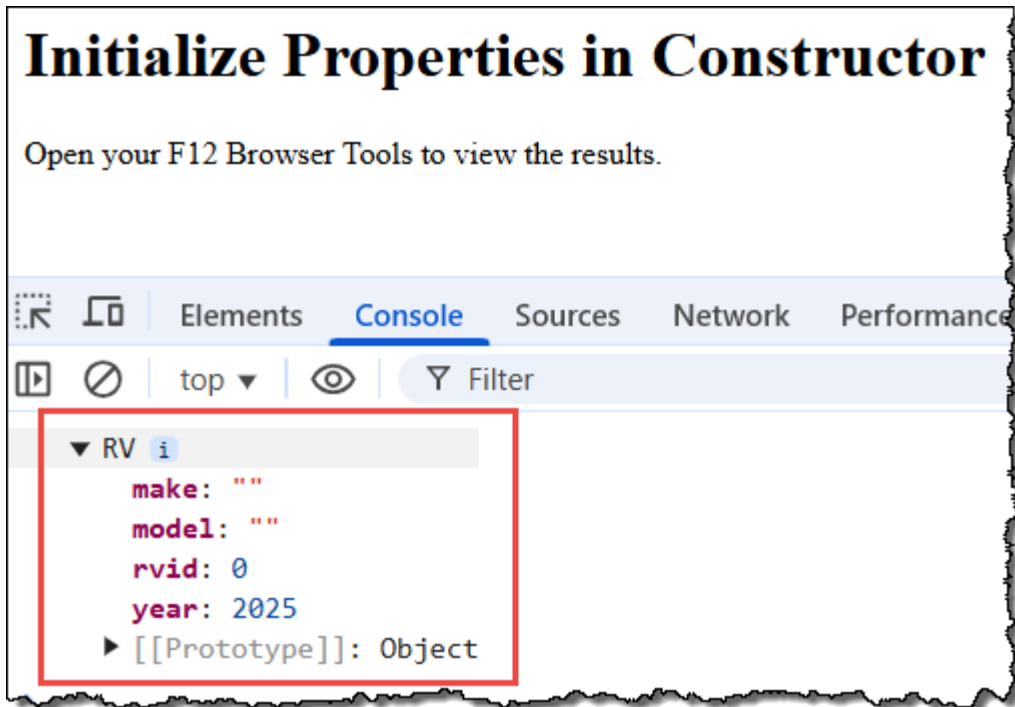Display the **class-init.html** file in your browser and your page should look like Figure 2.

Figure 2: Use the constructor to initialize the properties to valid data.

Lab 3: Add Methods to the RV Class

Create a file named **class-methods**.**html** and add the following HTML into the new file.

```
<!DOCTYPE html>
<html>

<head>
  <meta charset="utf-8" />
  <meta name="viewport" content="width=device-width" />

  <title>Add Methods to the RV Class</title>

  <link href="styles/site.css" rel="stylesheet" />
</head>

<body>
  <header>
    <h1>Add Methods to the RV Class</h1>
  </header>
```

```html
  <main>
    <p>Open your F12 Browser Tools to view the
results.</p>
  </main>

  <script>
    'use strict';

    // Create a class named RV
    class RV {
      // Add constructor with parameters
      // to set each of the properties
      constructor(rvid, year, make, model) {
        // Initialize the properties
        // to valid starting data

      }

      // Add the properties rvid,
      // year, make, and model


      // Add method to return a string
      // with rvid, year, make, and model
      // properties separated by \n


      // Add method to return the
      // entire object as stringified JSON

    }

    // Create an instance of an RV class
    // Pass RV values to the constructor


    // Display the RV object using getInfo()
    console.log(rv.getInfo());
    console.log("");

    // Display the RV object using asJSON()
    console.log(rv.asJSON());
  </script>
</body>

</html>
```

Write the appropriate code to create a class named **RV**. Define four properties; *rvid*, *year*, *make*, and *model* in the class.

Create a constructor into which you pass four parameters; *rvid*, *year*, *make*, and *model*, and use these parameters to initialize the four properties.

Create a method named **getInfo**() that returns a string with each property separated by a \n. Put the appropriate label in front of each such as "RV ID:", "Year: ", etc.

Create a method named **asJSON**() that returns a stringified version of this class.

Create an instance of the RV class and put it into a variable named *rv*. Pass to the constructor the values 1, 2025, "Newmar", "New Aire".

Call the getInfo() method and display the results in the console window.

Call the asJSON() method and display the results in the console window.

Try It Out

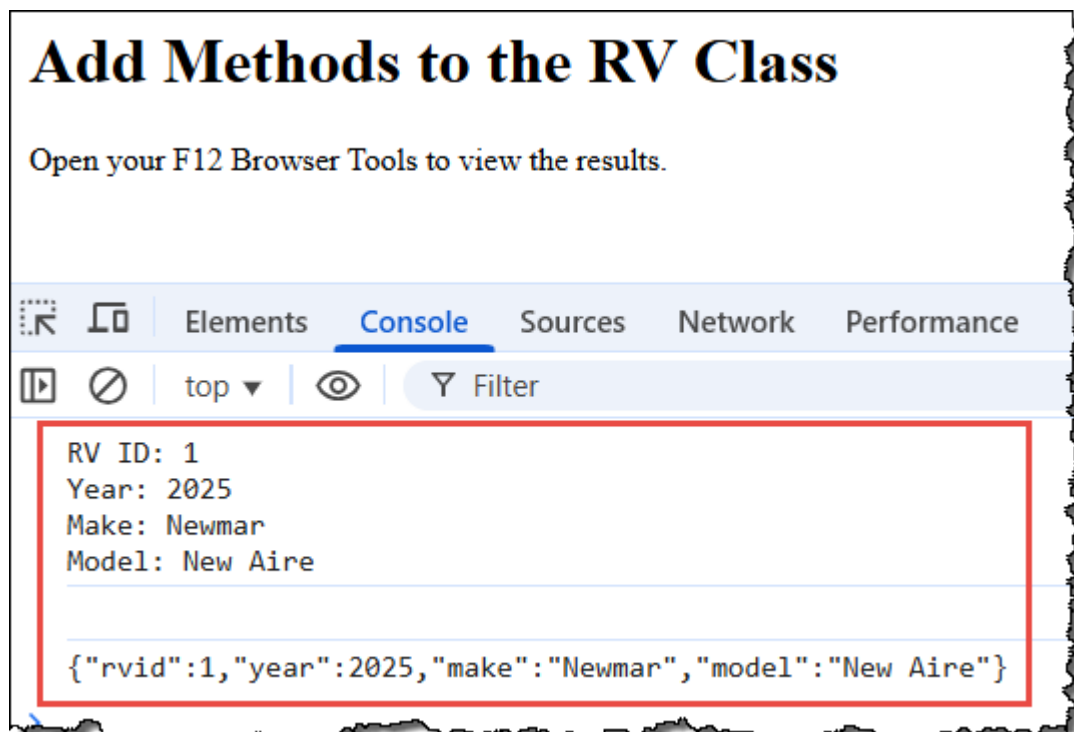Display the **class-methods.html** file in your browser and your page should look like Figure 3.

Figure 3: Add methods to your class to display data the way you want.

THIS PAGE INTENTIONALLY LEFT BLANK

Answers

Lab 1

```javascript
// Create a class named RV
// DO NOT USE a constructor
class RV {
  // Add the properties
  // rvid, year, make,
  // and model here
  rvid;
  year;
  make;
  model;
}

// Create an instance of an RV class
let rv = new RV();

// Set all the properties
// of the RV object
rv.rvid = 1;
rv.year = 2025;
rv.make = "Newmar";
rv.model = "New Aire";

// Display the RV object
// in the console window
console.log(rv);
```

Lab 2

```javascript
// Create a class named RV
class RV {
  // Create a constructor and
  // initialize the properties
  // to valid starting data
  constructor() {
    this.rvid = 0;
    this.year = new Date().getFullYear();
    this.make = "";
    this.model = "";
  }
```

```
  // Add the properties
  // rvid, year, make,
  // and model here
  rvid;
  year;
  make;
  model;
}

// Create an instance of an RV class
let rv = new RV();

// Display the RV object
// with default in the console window
console.log(rv);
```

Lab 3

```
// Create a class named RV
class RV {
  // Add constructor with parameters
  // to set each of the properties
  constructor(rvid, year, make, model) {
    // Initialize the properties
    // to valid starting data
    this.rvid = rvid;
    this.year = year;
    this.make = make;
    this.model = model;
  }

  // Add the properties rvid,
  // year, make, and model
  rvid;
  year;
  make;
  model;

  // Add method to return a string
  // with rvid, year, make, and model
  // properties separated by \n
  getInfo() {
    return `RV ID: ${this.rvid}\nYear:
${this.year}\nMake: ${this.make}\nModel: ${this.model}`;
  }
```

```
    // Add method to return the
    // entire object as stringified JSON
    asJSON() {
      return JSON.stringify(this);
    }
}
```