Create Class Properties 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 Getters and Setters

Create a file named **class-get-set**.**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 Getters and Setters</title>

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

<body>
  <header>
    <h1>Create Getters and Setters</h1>
  </header>

  <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
        this.rvid = rvid;
        this.year = year;
        this.make = make;
        this.model = model;
        // Initialize private fields here


      }

      // Add the fields rvid, year,
      // make, model, minLength,
      rvid;
      year;
      make;
      model;

      // Add private fields
      // minLength/maxLength here



      // Public Property Getters and Setters
      // for minLength and maxLength

    }

    // Create an instance of an RV class
    // Pass RV values to the constructor
    let rv = new RV(1, 2025, "Newmar", "New Aire");
    rv.minLength = 30;
    rv.maxLength = 38;

    console.log(`rv.minLength = ${rv.minLength}`);
    console.log(`rv.maxLength = ${rv.maxLength}`);
  </script>
</body>

</html>
```

Add two private fields to this class named *minLength* and *maxLength*. Initialize these two fields to 24 and 45 respectively in the constructor. Create property getters and setters for each of these two fields.

Try It Out

Display the **class-get-set.html** file in your browser and your page should look like Figure 1.

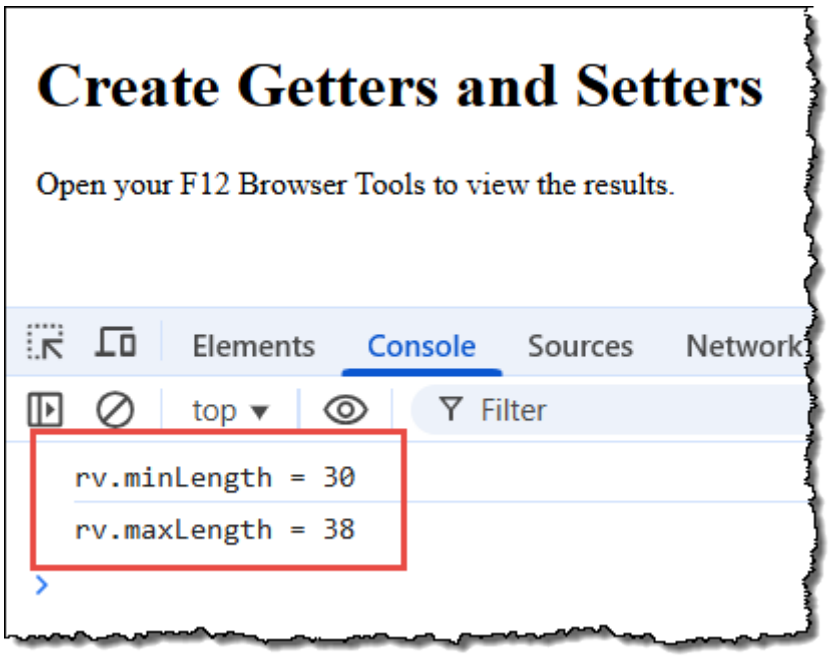

Figure 1: Add two private fields that are exposed from the class using property getters and setters.

Lab 2: Add a Private Method for Validation

Create a file named **class-private-method.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 a Private Method for Validation</title>

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

<body>
  <header>
    <h1>Add a Private Method for Validation</h1>
  </header>

  <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
        this.rvid = rvid;
        this.year = year;
        this.make = make;
        this.model = model;
        // Initialize private fields here
        this.#minLength = 24;
        this.#maxLength = 45;
      }

      // Add the fields rvid, year,
      // make, model, minLength,
      rvid;
      year;
      make;
      model;

      // Add private fields
      // minLength/maxLength here
      #minLength;
      #maxLength;
```

```
      // Public Property Getters and Setters
      // for minLength and maxLength
      get minLength() {
        return this.#minLength;
      }
      set minLength(value) {
        this.#minLength = value;
        // Call method to validate
        // minLength to maxLength

      }
      get maxLength() {
        return this.#maxLength;
      }
      set maxLength(value) {
        this.#maxLength = value;
        // Call method to validate
        // minLength to maxLength

      }

      // Private method to ensure minLength
      // is less than maxLength

    }

    // Create an instance of an RV class
    // Pass RV values to the constructor
    let rv = new RV(1, 2025, "Newmar", "New Aire");
    rv.minLength = 35;
    rv.maxLength = 34;

    console.log(`rv.minLength = ${rv.minLength}`);
    console.log(`rv.maxLength = ${rv.maxLength}`);
  </script>
</body>

</html>
```

Write a private method named **checkLength**() that checks if the *minLength* field is less than the *maxLength* field. If it is not, set the *minLength* field to one less than the value in the *maxLength* field.

Try It Out

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

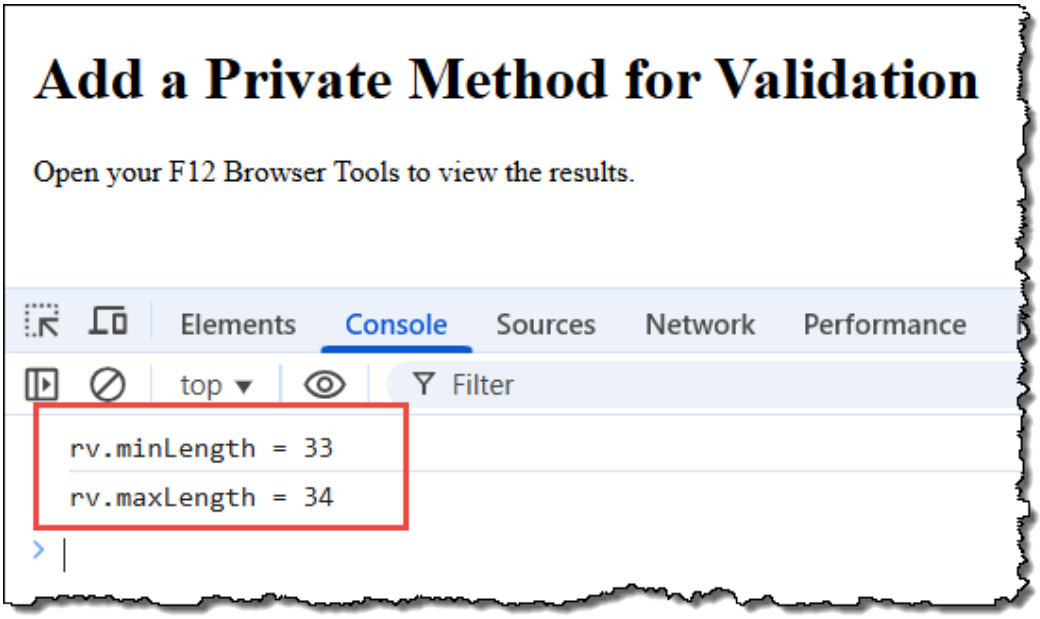

Figure 2: Ensure minLength field value is always less than the maxLength field value.

THIS PAGE INTENTIONALLY LEFT BLANK

Answers

Lab 1

```
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;
    // Initialize private fields here
    this.#minLength = 24;
    this.#maxLength = 45;
  }

  // Add the fields rvid, year,
  // make, model, minLength,
  rvid;
  year;
  make;
  model;

  // Add private fields
  // minLength/maxLength here
  #minLength;
  #maxLength;

  // Public Property Getters and Setters
  // for minLength and maxLength
  get minLength() {
    return this.#minLength;
  }
  set minLength(value) {
    this.#minLength = value;
  }
  get maxLength() {
    return this.#maxLength;
  }
  set maxLength(value) {
    this.#maxLength = value;
  }
```

```
}
```

Lab 2

```
// Public Property Getters and Setters
// for minLength and maxLength
get minLength() {
  return this.#minLength;
}
set minLength(value) {
  this.#minLength = value;
  // Call method to validate
  // minLength to maxLength
  this.#checkLength();
}
get maxLength() {
  return this.#maxLength;
}
set maxLength(value) {
  this.#maxLength = value;
  // Call method to validate
  // minLength to maxLength
  this.#checkLength();
}

// Private method to ensure minLength
// is less than maxLength
#checkLength() {
  if (this.#minLength > this.#maxLength) {
    this.#minLength = this.#maxLength - 1;
  }
}
```