Arrays

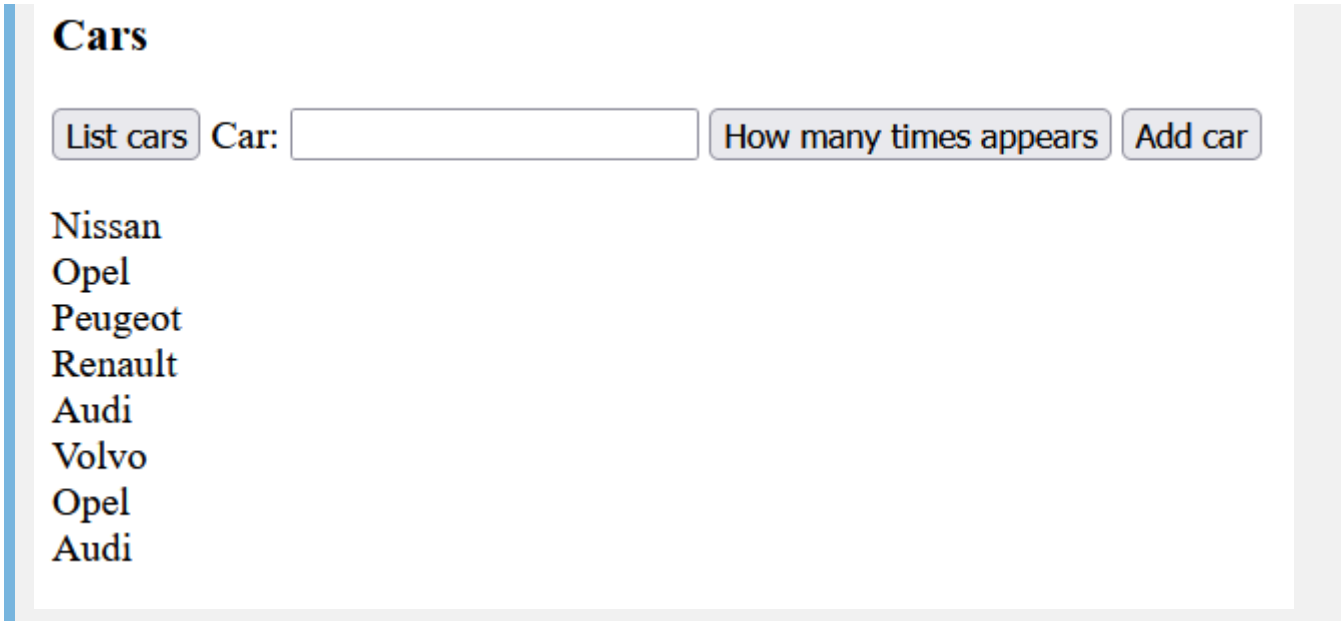
You can find HTML-templates for the exercises in folder `templates`.

1. Cars

The array to be used is:

```
let cars = ["Nissan", "Opel", "Peugeot", "Renault", "Audi", "Volvo", "Opel",
"Audi"];
```

a) Create a program that lists the cars in the array



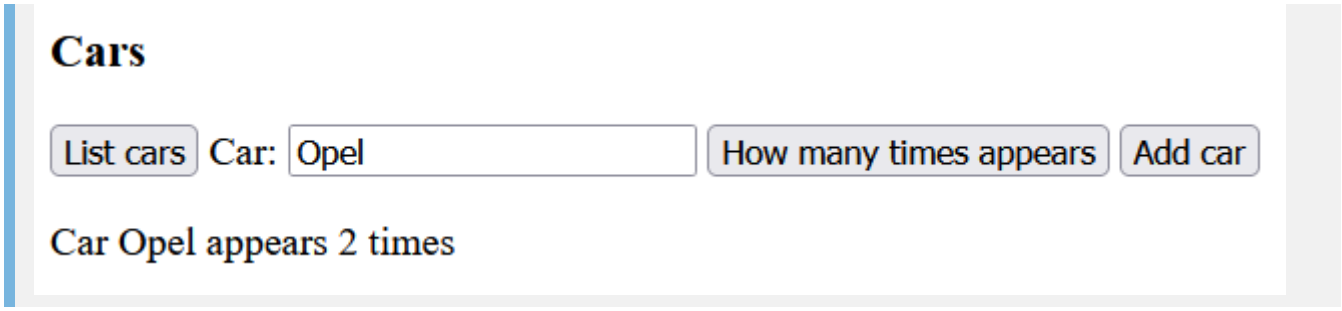
- Declare the array inside a script element.
- List the array using a for loop.

```
// Declare a variable to collect the cars in the array
let text = "";

// Iterate through the array and collect the cars into the text variable
for (let i = 0; i < cars.length; i++) {
    text = text + cars[i] + "<br>";
}
```

- Write the result to an HTML page.

b) Create a program that tells how many times a specific car appears in the array.

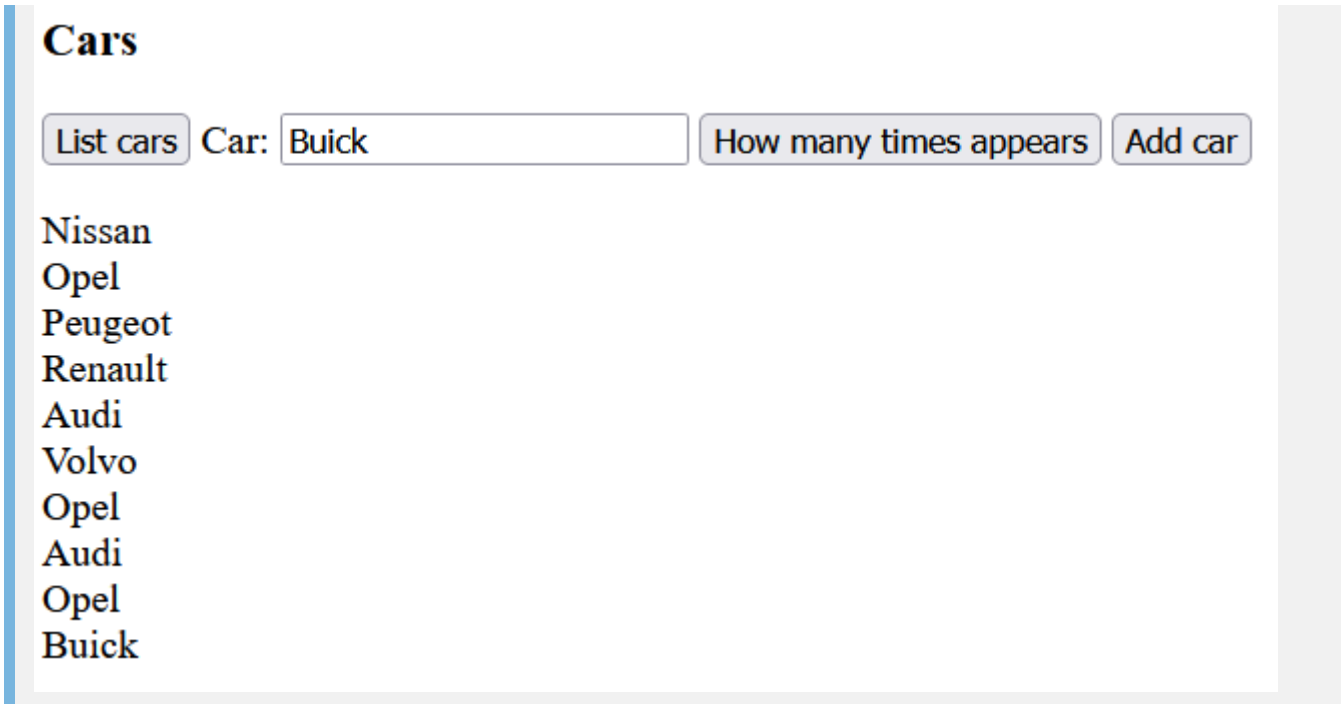


- The search is done using a for loop, where each car in the array is compared to the value of the input field using `if (cars[i] == searchValue)`.
- Declare a `count` variable (initialize it to zero) above the loop to store the number of occurrences.
- If the loop finds that the car in the array matches the value of the input field, increment the `count` variable by one.

```
let count = 0;

// Iterate through the array and collect the values into the text variable
for (let i = 0; i < cars.length; i++) {
    // If the value in the array matches the value of the input field
    if (cars[i] == searchValue) {
        count = count + 1;
    }
}
```

c) Create a program to add a new car to the array



2. Courses

Create a program that displays the course codes for the first semester.

The array is:

```
let courses = ["STU001HH1A", "ICB001HH1A", "COM001HH1A", "DIG001IT1A",
"SOF001IT1A", "ICI001IT1A", "ICB001IT1A"];
```

> **Courses**
>
> STU001HH1A
> ICB001HH1A
> COM001HH1A
> DIG001IT1A
> SOF001IT1A
> ICI001IT1A
> ICB001IT1A

3. Rainfall

Create a program that calculates the annual rainfall in Helsinki.

> **Annual rainfall**
>
> Annual rainfall in Helsinki is 637.0 mm

Below is an array containing the rainfall for each month.

```
let rainfall = [47.0, 36.6, 34.7, 37.0, 41.9, 47.5, 61.7, 74.8, 65.4, 69.7, 66.1,
54.6];
```

The code is conceptually:

```
let total = 0;
for (let i = 0; i < rainfall.length; i++) {
  // total = total + the value at index i in the array
}

// write the result
```

4. Temperatures

Create a program that displays the average monthly temperatures in Helsinki and their average.

> **Monthly average temperatures in Helsinki**
>
> 1. -3.3
> 2. -4.7
> 3. -1.3
> 4. 3.9
> 5. 10.2
> 6. 14.6
> 7. 17.8
> 8. 16.3
> 9. 11.5
> 10. 6.6
> 11. 1.6
> 12. -2.0
>
> Annual average temperature in Helsinki is 5.9

Below is an array containing the average temperature for each month.

```
let temperatures = [-3.3, -4.7, -1.3, 3.9, 10.2, 14.6, 17.8, 16.3, 11.5, 6.6, 1.6,
-2.0];
```

Hints

- The `toFixed(1)` method can be used to format the result to one decimal place.

5. Electricity prices

Below are the hourly electricity prices for a certain day from 0-12 o'clock.

```
let hourlyPrices = [6.91, 3.26, 1.62, 0.66, 0.91, 0.99, 3.13, 5.01, 16.45,
23.23, 17.32, 8.29, 3.77];
```

Write a program that lists the hourly prices, and the lowest and highest electricity price for the period.

Price at 0.00: 6.91 c/kWh
Price at 1.00: 3.26 c/kWh
Price at 2.00: 1.62 c/kWh
Price at 3.00: 0.66 c/kWh
Price at 4.00: 0.91 c/kWh
Price at 5.00: 0.99 c/kWh
Price at 6.00: 3.13 c/kWh
Price at 7.00: 5.01 c/kWh
Price at 8.00: 16.45 c/kWh
Price at 9.00: 23.23 c/kWh
Price at 10.00: 17.32 c/kWh
Price at 11.00: 8.29 c/kWh
Price at 12.00: 3.77 c/kWh

Lowest price: 0.66 c/kWh
Highest price: 23.23 c/kWh

6. Gift wishes

Create a program that asks for gift wishes, lists them, and tells how many wishes have been given.

Requirements:

- Clear the input field after adding a wish.
- Prevent empty wishes from being added to the array. An empty wish means when nothing or only spaces are entered in the wish field.

Gift wishes

Wish: Add

Wishes entered:

- Wireless headphones
- Book: The Pragmatic Programmer
- Gift card to a coffee shop

Total wishes: 3

Hints

- You can implement the list as an HTML unordered list (`<ul>`).

Bonus task

- Set the mouse cursor (focus) into the wish field after adding a wish.

7. Comma-separated values

CSV is a format where data is in a string, with each piece of information (field) separated by a delimiter.

Use the JavaScript string method `split` to split a csv-record into individual fields and use the field values to build the output.

```
let studentRecord = "Jane Doe;12/24/1990;Helsinki;Haaga-Helia University of
Applied Sciences";
";
```

> **Comma separated values**
>
> **Name:** Jane Doe
> **Date of birth:** 12/24/1990
> **City:** Helsinki
> **School:** Haaga-Helia University of Applied Sciences

8. Lottery

Create a program that draws 7 different lottery numbers between 1-40. No lottery number may appear more than once!

> **Lottery numbers**
>
> 22 4 30 28 21 17 11

Bonus tasks

- Sort the numbers in ascending order. Note that method `sort` does not work for number arrays as one might expect.
- Style the numbers using the CSS rules in the template.

> **Lottery numbers**
>
> 5 18 26 28 33 38 40