MehdiBouchachi
Web Developer

THE COMPLETE FRONT-END DEVELOPMENT

BY MEHDI BCH

SECTION

JAVASCRIPT REVEIW

LECTURE

THE ARRAY
(MAP, FILTER, REDUCE, SORT) METHOD

THE ARRAY (MAP, FILTER, REDUCE, SORT) METHOD

BEFORE WE START

👉 Each movie has **its own review field** containing ratings. Notably, these review counts may differ across movies.

```
rating: 3.7,
```

```
rating: 4.8,
```

```
rating: 4.47,
```

```
rating: 4.5,
```

```
rating: 4.44,
```

👉 Our objective is to determine **the average rating** for a **chosen genre** and **arrange the movies** within that genre based on their ratings.

THE ARRAY (MAP, FILTER, REDUCE, SORT) METHOD

THE ARRAY METHODS

👉 To achieve this, we'll use array methods like map(), filter(), reduce(), and sort().

👉 These tools will help us efficiently handle and analyze the movie data.

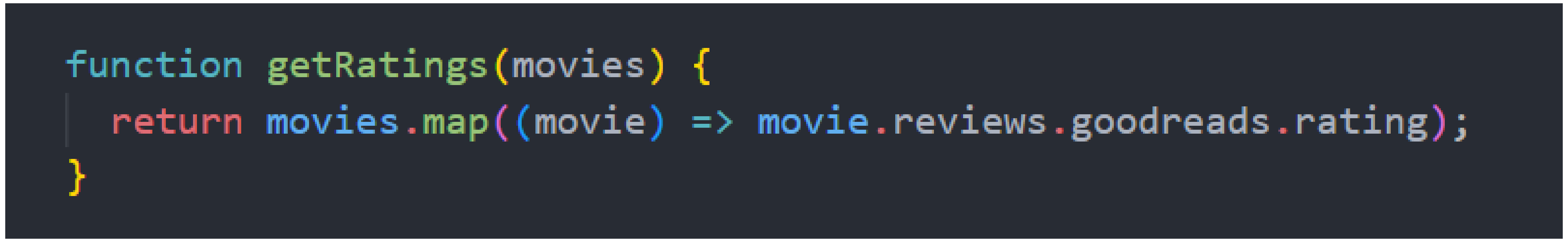
THE ARRAY (MAP, FILTER, REDUCE, SORT) METHOD

MAP() METHOD

👉 The map method is used to create a new array from an existing one, by applying a function to each of the elements of the array.

👉 The map method does not change the original array.

👉 array.map(callback(currentValue, index, array), thisArg)



```
function getRatings(movies) {
  return movies.map((movie) => movie.reviews.goodreads.rating);
}
```

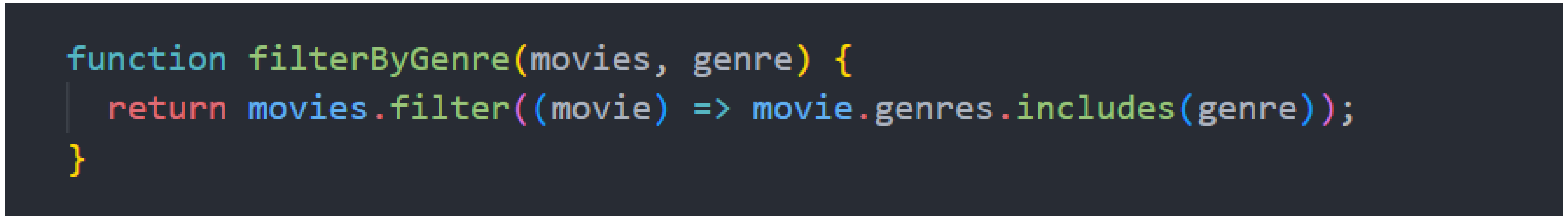
THE ARRAY (MAP, FILTER, REDUCE, SORT) METHOD

FILTER() METHOD

👉 The filter() method creates a new array with all elements that pass a provided condition.

👉 It iterates through each element of the array and includes elements that satisfy the condition in the new array.

👉 array.filter(callback(element, index, array), thisArg)



```
function filterByGenre(movies, genre) {
  return movies.filter((movie) => movie.genres.includes(genre));
}
```

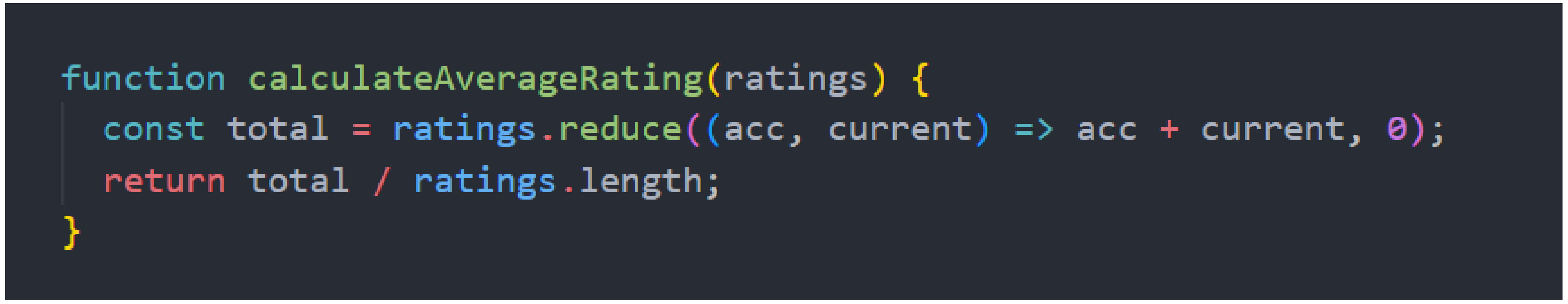
THE ARRAY (MAP, FILTER, REDUCE, SORT) METHOD

REDUCE() METHOD

👉 The reduce() method applies a function against an accumulator and each element in the array to reduce it to a single value.

👉 It iterates over each element of the array and accumulates a single value based on the provided function.

👉 array.reduce(callback(accumulator, currentValue, index, array), initialValue)



```
function calculateAverageRating(ratings) {
  const total = ratings.reduce((acc, current) => acc + current, 0);
  return total / ratings.length;
}
```

THE ARRAY (MAP, FILTER, REDUCE, SORT) METHOD

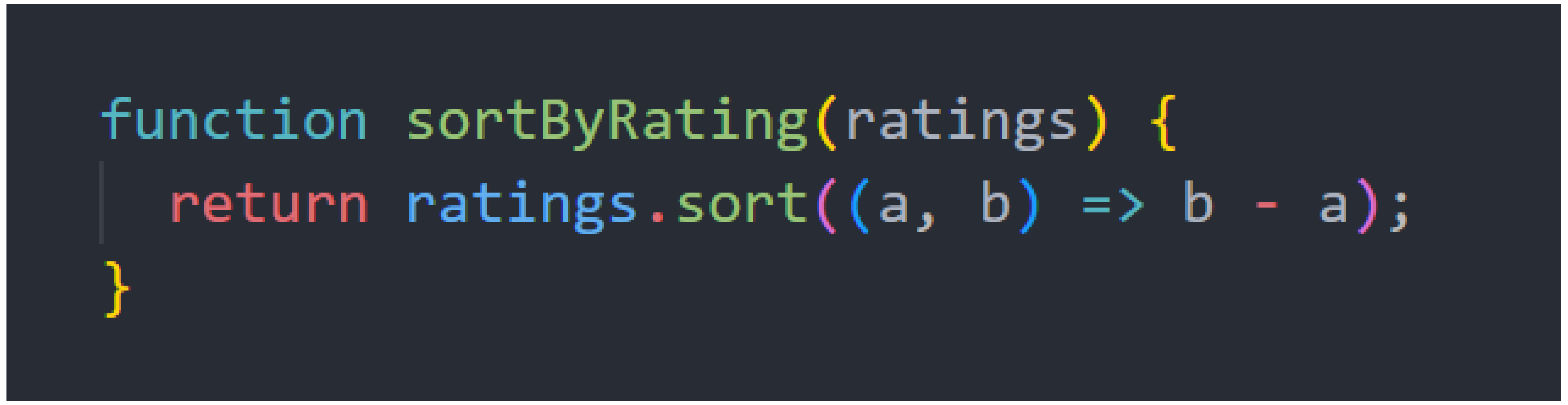
SORT() METHOD

👉 The sort() method sorts the elements of an array in place and returns the sorted array.

👉 By default, it sorts elements as strings in ascending order. However, you can provide a custom comparison function to define your own sorting logic.

👉the original array is modified, and there is no need to assign the result to a new variable.

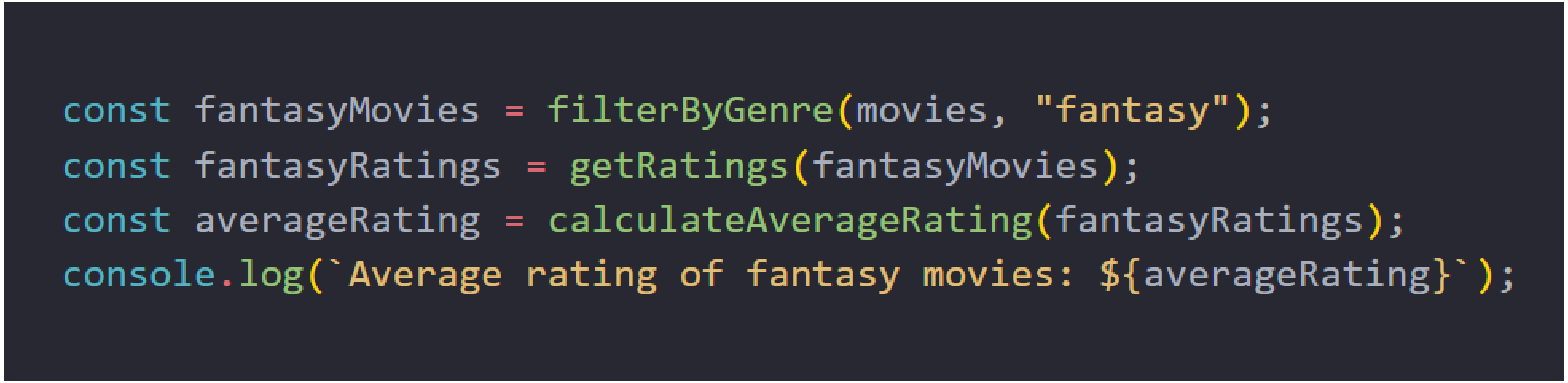
👉 array.sort(compareFunction)



```javascript
function sortByRating(ratings) {
  return ratings.sort((a, b) => b - a);
}
```

THE ARRAY (MAP, FILTER, REDUCE, SORT) METHOD

AVERAGE RATING



```javascript
const fantasyMovies = filterByGenre(movies, "fantasy");
const fantasyRatings = getRatings(fantasyMovies);
const averageRating = calculateAverageRating(fantasyRatings);
console.log(`Average rating of fantasy movies: ${averageRating}`);
```

THE ARRAY (MAP, FILTER, REDUCE, SORT) METHOD

SORTING MOVIES

```
const fantasyMovies = filterByGenre(movies, "fantasy");
const fantasyRatings = getMoviesRatings(fantasyMovies);
const sortedFantasyMovies = sortByRating(fantasyRatings);
```

SEE YOU SOON...