The provided code has a small issue: it does not initialize the accumulator `acc` with the initial value. Since the `reduce()` function is used without an initial value, it will use the first element of the array (`1` in this case) as the initial value for `acc`. Additionally, the accumulator object should have `evens` and `odds` properties to store the even and odd numbers, respectively.

Let's fix the code by initializing the accumulator properly and adding the `evens` and `odds` properties:

```javascript

const numbers = [1, 2, 3, 4, 5];

const result = numbers.reduce((acc, num) => {

if (num % 2 === 0) {

acc.evens.push(num);

} else {

acc.odds.push(num);

}

return acc;

}, { evens: [], odds: [] });

console.log(result);

```

Now, let's explain the output of the fixed code:

The `reduce()` method iterates through each element of the `numbers` array, and for each number, it checks if it's even or odd. Based on this check, it appends the number to the respective array property `evens` or `odds` in the accumulator object.

The output of the code will be:

```

{ evens: [2, 4], odds: [1, 3, 5] }

```

Explanation:

- The accumulator object starts with an empty `evens` array and an empty `odds` array: `{ evens: [], odds: [] }`.

- When the `reduce()` function processes the first element, which is `1`, it's an odd number, so it's pushed to the `odds` array in the accumulator: `{ evens: [], odds: [1] }`.

- The second element is `2`, which is an even number, so it's pushed to the `evens` array in the accumulator: `{ evens: [2], odds: [1] }`.

- The third element is `3`, an odd number, so it's pushed to the `odds` array: `{ evens: [2], odds: [1, 3] }`.

- The fourth element is `4`, an even number, so it's pushed to the `evens` array: `{ evens: [2, 4], odds: [1, 3] }`.

- The fifth and final element is `5`, an odd number, so it's pushed to the `odds` array: `{ evens: [2, 4], odds: [1, 3, 5] }`.

After iterating through all elements, the `reduce()` method returns the final accumulator object, which contains the `evens` and `odds` arrays with the even and odd numbers, respectively. The output is logged to the console.