

# **(Notes) Map, filter and reduce**

## **Question 1: Array.map()**

The map() method in JavaScript creates a new array populated with the results of calling a provided function on every element in the calling array. It doesn't modify the original array but returns a new modified array based on the callback function's logic.

## **Question 2: Array.filter()**

The filter() method creates a new array with all elements that pass the test implemented by the provided function. It returns a filtered array based on the condition specified in the callback function, where only elements that satisfy the condition are included.

## **Question 3: Array.reduce()**

The reduce() method executes a reducer function (that you provide) on each element of the array, resulting in a single output value. It's often used for aggregating data, such as calculating a sum, by iterating through the array and accumulating the results based on the logic in the callback function.

## **Question 4: Map Polyfill**

This code snippet adds a custom implementation of the map() method to the Array prototype. It mimics the functionality of the native map() method by iterating through the array and applying a callback function to each element to create a new array with the modified values.

## **Question 5: Filter Polyfill**

Similar to the Map Polyfill, this code adds a custom implementation of the filter() method to the Array prototype. It iterates through the array and applies a callback function to each element, returning a new array containing only the elements that meet the specified condition in the callback.

## **Question 6: Reduce Polyfill**

This snippet introduces a custom implementation of the reduce() method for arrays. It iterates through the array, applying a callback function that performs reduction or aggregation operations, similar to the native reduce() method, and returns the final accumulated result.

## **Question 7: map vs foreach**

This section compares the usage of map(), forEach(), and traditional for() loops for processing data in an array of student records. It demonstrates how each method can be used to achieve specific tasks, such as transforming data, filtering based on conditions, and aggregating information, highlighting the differences in syntax and approach between these array iteration methods.