**JavaScript Methods: Agenda and Practice Questions**

**Agenda:**

1. **Introduction to JavaScript Methods**
   * Importance of mastering array, string, and object methods.
2. **Array Methods Overview**
   * Detailed explanations of frequently used array methods.
   * Practice questions focused on array transformations, dry runs, and logic building.
3. **String Methods Overview**
   * Key string manipulation techniques.
   * Practice questions emphasizing string handling and transformations.
4. **Object Methods Overview**
   * Understanding common object operations.
   * Practice questions centered on object property manipulation.
5. **Wrap-up**
   * Review of challenges and areas of improvement.

**Array Methods: Key Explanations**

1. **push()** – Adds one or more elements to the end of an array and returns the new length of the array.
2. **pop()** – Removes the last element from an array and returns that element.
3. **shift()** – Removes the first element from an array and returns that element.
4. **unshift()** – Adds one or more elements to the beginning of an array and returns the new length of the array.
5. **map()** – Creates a new array populated with the results of calling a provided function on every element.
6. **filter()** – Creates a new array with all elements that pass the test implemented by the provided function.
7. **reduce()** – Executes a reducer function on each array element, resulting in a single output value.
8. **forEach()** – Executes a provided function once for each array element, but does not return a new array.
9. **find()** – Returns the value of the first element that satisfies the provided testing function.
10. **every() / some()** – Checks whether all or some elements in an array satisfy a condition.
11. **sort()** – Sorts the elements of an array in place and returns the array.
12. **slice()** / **splice()** – slice returns a portion of an array, while splice adds/removes elements from an array.
13. **includes()** – Determines whether an array includes a certain value.
14. **concat()** – Merges two or more arrays into one.
15. **set()** – Typically used in the context of sets (ES6), which are collections of unique values.

**String Methods: Key Explanations**

1. **split()** – Splits a string into an array of substrings.
2. **toUpperCase() / toLowerCase()** – Changes string case.
3. **charAt()** – Returns the character at a specified index.
4. **substring() / substr()** – Returns a portion of the string.
5. **replace()** – Replaces a substring with a new string.
6. **trim()** – Removes whitespace from both ends of a string.
7. **includes()** – Determines if one string contains another.
8. **indexOf() / lastIndexOf()** – Finds the index of a character or substring.
9. **repeat()** – Returns a new string with a specified number of copies.
10. **match()** – Retrieves matches from a string using a regular expression.

**Object Methods: Key Explanations**

1. **Object.keys()** – Returns an array of a given object's property names.
2. **Object.values()** – Returns an array of a given object's values.
3. **Object.entries()** – Returns an array of a given object's key-value pairs.
4. **Object.assign()** – Copies properties from one or more source objects to a target object.
5. **hasOwnProperty()** – Checks whether an object has a specific property as its own (not inherited).
6. **delete** – Removes a property from an object.
7. **Object.freeze()** / **Object.seal()** – Prevents modification or addition of new properties to an object.
8. **Object.create()** – Creates a new object with a specified prototype.
9. **Object.fromEntries()** – Transforms a list of key-value pairs into an object.
10. **Object.is()** – Compares two values to determine if they are the same.

**Practice Questions**

**Array Methods:**

1. Given an array of numbers, use push() to add a new number to the end of the array, then return the new length of the array.
2. Create a function that uses pop() to remove the last element of an array and returns that element.
3. Use shift() to remove the first element from an array and log the remaining array.
4. Write a function that uses unshift() to add an element to the beginning of an array.
5. Using map(), create a new array from an array of numbers where each number is squared.
6. Create a function that filters out all strings shorter than 4 characters from an array using filter().
7. Write a reduce() function that sums all numbers in an array.
8. Perform a dry run of the forEach() method to log each array element along with its index.
9. Using find(), return the first object in an array that has a property isActive set to true.
10. Sort an array of words by their length using the sort() method.
11. Combine two arrays of fruits using concat().
12. Remove the last 2 elements of an array and add 2 new ones at the end using splice().
13. Create a set from an array of numbers and return its size using set().

**String Methods:**

1. Split a sentence into an array of words and then join them back with dashes using split() and join().
2. Convert a string to uppercase and count how many characters are in the string.
3. Find the position of the first occurrence of the word "bootcamp" in a sentence using indexOf().
4. Replace all spaces in a string with underscores using replace().
5. Extract the last 5 characters of a string using substring() or substr().
6. Check if a string contains the substring "JavaScript" using includes().

**Object Methods:**

1. Write a function that returns an array of keys of an object using Object.keys().
2. Convert an object into an array of [key, value] pairs using Object.entries().
3. Merge two objects using Object.assign(), where the second object overrides properties from the first.
4. Write a function that freezes an object and tries to modify one of its properties.
5. Create an object and check if a given key exists using hasOwnProperty().

**End of Document**

This revised agenda and series of questions will give you a comprehensive foundation in mastering essential JavaScript methods through understanding and application. Feel free to reach out if you have any further modifications or specific areas you'd like to explore!