

# JavaScript Group Discussion

---

## External JavaScript and Comments

---

1. Discuss possible and distinct five benefits of using external JavaScript
  2. Discuss possible and distinct five benefits of JavaScript comments
  3. Discuss possible and distinct five recommendation on JavaScript comments
- 

## Data Types and Variables

---

1. List and explain with the help of example all JavaScript primitive data types
  2. List and explain with the help of example all JavaScript non-primitive data types
  3. Discuss all rules to be followed while declaring different JavaScript variables
  4. With the help of clear and short source-codes differentiate between JavaScript local variables from JavaScript global variables
  5. In JavaScript, variables can be declared in different way depending on how they are going to be used. How a JavaScript global variable can be declared within function.
  6. Discuss on Internals of global variable in JavaScript
  7. With the help of example, discuss to explain about JavaScript Literals
  8. Explain the difference between null and undefined in JavaScript
- 

## JavaScript Arrays

---

9. Discuss on when do we need to use arrays?
10. The concept of finding the intersection between arrays is widely applicable in scenarios where you need to identify common elements between two sets of data. For example, imagine you have two arrays representing employees in two different committees: Advisory Committee and Steering Committee. You might want to find out which employees are common to both committees. Given bellow committees, find their intersection.

```
const steeringCommittee = ['Alice', 'Bob', 'Charlie'];
const advisoryCommittee = ['Bob', 'David', 'Alice'];

const commonEmployees = steeringCommittee.filter(employee =>
  advisoryCommittee.includes(employee)
);
console.log(commonEmployees); // Output: ['Alice', 'Bob']
```

# JavaScript Group Discussion

11. You may fall in a scenario where you do not need duplicates from an array when developing a given software. Assume and create your array, include duplicated elements in it and implement a simple, clear, short, run-fast function that is able to remove those duplicates.

```
function removeDuplicates(arr) {  
  return [...new Set(arr)];  
}  
  
const arr=[200,1,30,2,30,2]  
console.log(removeDuplicates(arr))
```

12. In online tests, it is recommended to shuffle questions to prevent cheating in exam, shuffling questions makes it more challenging for participants to collaborate or cheat during the exam. If each participant sees questions in a different order, it becomes more difficult for them to share specific questions or answers in real-time. This contributes to the overall fairness of the examination process. Consider bellow set of question to be displayed on a webpage

```
function shuffleArray(arr) {  
  return arr.sort(() => Math.random() - 0.5);  
}  
  
// Array of participants  
const participants = ['Participant1', 'Participant2', 'Participant3', 'Participant4', 'Participant5'];  
// Array of questions  
const questions = [  
  'Question1',  
  'Question2',  
  'Question3',  
  'Question4',  
  'Question5'  
];  
  
// Create a map to store the shuffled order of questions for each participant  
const shuffledOrders = {};  
// Shuffle questions for each participant  
participants.forEach(participant => {  
  const shuffledQuestions = [...questions]; // Create a copy of the original questions array  
  shuffleArray(shuffledQuestions);  
  shuffledOrders[participant] = shuffledQuestions;  
});  
  
// Display the shuffled orders for each participant  
participants.forEach(participant => {  
  console.log(` ${participant}'s Shuffled Order:`, shuffledOrders[participant]);  
});
```

# JavaScript Group Discussion

13. Given date of birth from a given API, the data bellow was extracted from a JSON object from the said API. Make a function to return people born before 2000

```
const yearOfBirths = [1990, 2002, 1998, 2014, 2005, 2023];
const bornBefore2000 = (arr) => arr.filter((value) => {
  return value < 2000;
}, {});
console.log(bornBefore2000(yearOfBirths));
```

---

## JavaScript Objects

---

14. By providing examples,
- discuss two ways to access Object Properties
  - discuss Object Methods
  - discuss how to access object property inside an Object Method

---

## JavaScript Control Statements

---

- Discuss on the usage of JavaScript Control Statements
- How can you exit from a loop prematurely in JavaScript?
- What is the purpose of the continue statement in JavaScript?
- Can you explain the purpose of the do...while loop in JavaScript?
- Explain the concept of the "conditional (ternary) operator" in JavaScript.
- How can you use the switch statement with regular expressions in JavaScript?
- What is a "switch" statement in JavaScript, and when should you use it?
- Explain the "break" and "continue" statements in JavaScript control statements.
- How does the "switch" statement perform its comparison?
- How can you explain the key differences between "if," "else if," and "else" statements in JavaScript?
- How do you use the "switch" statement in JavaScript, and when is it preferable over a series of "if" statements?
- What is the significance of loops in JavaScript, and can you describe the differences between "for," "while," and "do...while" loops?
- How do you break out of a loop prematurely using the "break" statement, and when might this be useful in a program?
- What is the purpose of the "continue" statement, and how does it affect loop execution?
- Explain the concept of nested control statements in JavaScript. Provide an example where nested control statements are necessary.
- What is the "return" statement in JavaScript, and how does it impact the flow of control in a function?
- How do you handle exceptions and errors using control statements like "try," "catch," and "finally" in JavaScript?

# JavaScript Group Discussion

32. What are "conditional (ternary) operators" in JavaScript, and how do they differ from traditional "if" statements?
33. Describe a scenario where you might use "break" and "continue" statements in a real-world JavaScript project.

To be continued..... [Remember to read and understand all exercises provided in each lecture]