💡 \*\*Q.1\*\* Explain Hoisting in JavaScript

💡 \*\*Q.2\*\* Explain Temporal Dead Zone?

💡 \*\*Q.3\*\* Difference between var & let?

💡 \*\*Q.4\*\* What are the major features introduced in ECMAScript 6?

💡 \*\*Q.5\*\* What is the difference between \*\*`let`\*\* and \*\*`const`\*\* ?

</aside>

<aside>

💡 \*\*Q.6\*\* What is template literals in ES6 and how do you use them?

</aside>

<aside>

💡 \*\*Q.7\*\* What’s difference between map & forEach?

</aside>

<aside>

💡 \*\*Q.8\*\* How can you destructure objects and arrays in ES6?

</aside>

<aside>

💡 \*\*Q.9\*\* How can you define default parameter values in ES6 functions?4

Q10 What is the purpose of the spread operator (...) in ES6?

</aside>

Hoisting is a behavior in JavaScript where variable and function declarations are moved to the top of their respective scopes during the compilation phase before the code is executed. This means that regardless of where the declarations are placed in the code, they are treated as if they were at the top of their scope. However, it's important to note that only the declarations are hoisted, not the initializations or assignments.