

1. Write a function called "addNumbers" that takes two numbers as arguments and returns their sum. Call the function before it is declared to demonstrate hoisting.

```
console.log(addNumber(11,22)); // 33

function addNumber(num1,num2){
  return num1+num2;
}
```

2. Write a function called "multiplyNumbers" that takes two numbers as arguments and returns their product. Use function expressions to define the function and call the function before it is declared to demonstrate hoisting.

```
console.log(multiply(11,22)); // o/p -> Cannot access 'multiply' before initialization due to reference error

const multiply=function(num1,num2){
  return num1*num2;
}
```

3. Write a function that takes two numbers as arguments and returns their sum. Declare a variable inside the function using the var keyword and log its value to the console before it is assigned a value to demonstrate variable hoisting.

```
function addtwoNumber(num1,num2){
  console.log("sum before assignment value => ",sum);
  var sum=num1+num2;
  return sum;
}

console.log(addtwoNumber(10,20));
```

Console output :-

```
sum before assignment value => undefined
30
```

4. Declare three variables, one using let, one using var, and one using const, all inside a block scope. Assign them values and log their values to the console before and after they are declared to demonstrate variable hoisting.

```
console.log(var1); // o/p -> from var , due to hoisting
console.log(var2); // o/p -> Cannot access 'var2' before initialization due to reference error
console.log(var3); // o/p -> Cannot access 'var2' before initialization due to reference error

var var1="from var";
let var2="from let";
const var3="from const";

console.log(var1); // o/p -> from var
console.log(var2); // o/p -> from let
console.log(var3); // o/p -> from const
```

5. Declare a variable using let inside a block scope and attempt to log its value to the console before it is assigned a value to demonstrate the temporal dead zone.

```
console.log(tempvalue); // o/p -> throws error due to reference error and it is called temporal zone
let tempvalue="hello from temp"; // o/p -> here the variable is declared and assigned a value

console.log(tempvalue); // o/p -> hello from temp
```