❓ solution ❓

1️⃣ what is hoisting in JavaScript

Hoisting in JavaScript is a behavior where variable and function declarations are moved to the top of their containing scope during the compilation phase, regardless of where they are declared in the code. This allows variables and functions to be accessed before they are actually declared in the code. However, only the declarations are hoisted, not the initializations or assignments.

2️⃣ what is the difference between function declaration, function expression, constructor function, and arrow function along with their advantage and disadvantage

```
// *Function Declaration
function evangadi(){
    console.log("evangdi Tech")
}

// *function expression
const evangadi = function(){
    console.log("evangadi Tech")
}

const evangadi = ()=>{
    console.log("evangadi Tech")
}

//* constructor function
function personalInfo(name,age,address){
    this.name = name
    this.age = age
    this.address = address

 }

 let personOne = new personalInfo('a',2,"aa")
 console.log(personOne.name="tasew")
 console.log(personOne)
```

The advantage of function declaration is it follows hoisting principle of javaScript but others doenst

3️⃣ what is the difference between, function, callback function, and higher order function

```
//* Function
function add(a,b) {
    return a + b
}
//* Function
function divide(a,b){
    return a/b
}

function calculate(x,y,operation){  //* Calculate  higher-order function because
it takes another function
    return operation(x,y)  // operation is a call back function (since it passed
and invoked within a function )
}

console.log(calculate(2,2,add))
console.log(calculate(2,2,divide))
```

4️⃣ what is a ternary operator

```
let amount = 3;
 amount > 300 ? 'expensive' : 'affordable'; // depending on amount , if true it
will give the “expensive” value if not it will give “affordable” .
in general it’s a simplest form of if else condition
```

5️⃣ why post increment doesn't work on return

Because return will get first the value and will not address further operations

```
function n(a) {
    if(a>5){
        return  a++
    }
}

console.log(n(7))  will result 7 it self but for pre increment(++a) it gives 8
```

6️⃣ what is the disadvantage of using isNaN while checking a value

```
 console.log(isNaN('') will give false meaning ‘’ string is considered as 0
```