Day 4 – [24th June 2025]

TOPICS COVERED

JavaScript Coercion: Abstract vs Strict Equality

Today, I explored how JavaScript handles equality comparison differently when using == (abstract) and === (strict).

- == performs type coercion before comparing values:
  "5" == 5 // true – string converted to number
  null == undefined // true
  0 == false // true
- === checks both value and type without conversion:
  "5" === 5 // false – string vs number
  0 === false // false – number vs boolean
  5 === 5 // true – same type and value

Hoisting:

I also learned about hoisting in JavaScript, where declarations are conceptually moved to the top of their scope before code runs.

var is hoisted and initialized with undefined:

console.log(x); // undefined

var x = 10;

let and const are hoisted too but not initialized, which leads to a Temporal Dead Zone error if accessed before declaration.

Function declarations are fully hoisted and can be called before their definition.

TOOLS USED:

Visual Studio Code (VS Code)

Chrome Browser (JavaScript Console)

TASK:

Practice writing different types of loops:

1. for Loop

```
for (let i = 0; i < 5; i++) {
  console.log(i);
}
```

```
C:\Program Files\nodejs\node.exe .\script.js
0
1
2
3
4
```

2. while Loop

```
let i = 0;
while (i < 5) {
  console.log(i);
  i++;
}
```

```
C:\Program Files\nodejs\node.exe .\script.js
0
1
2
3
4
```

3. do-while Loop

```
let i = 0;
do {
  console.log(i);
  i++;
} while (i < 5);
```

```
C:\Program Files\nodejs\node.exe .\script.js
0
1
2
3
4
```

4. for-of Loop (for arrays, strings)

```
for (let i of "Navpreet") {
  console.log(i);
}
```

```
C:\Program Files\nodejs\node.exe .\script.js
N
a
v
p
r
2 e
t
```

5. for-in Loop (for objects)

```
let obj = { name: "Navpreet", age: 20 };
for (let key in obj) {
  console.log(key + ": " + obj[key]);
}
```

```
C:\Program Files\nodejs\node.exe .\script.js
name: Navpreet
age: 20
```