

Training Day 4

24th June 2025

TOPICS COVERED

- **JavaScript Coercion: Abstract vs Strict Equality**

In JavaScript, coercion often shows up in the difference between `==` (abstract equality) and `===` (strict equality):

`==` (Abstract Equality):

Compares values after converting them to the same type (type coercion happens).

Example:

```
"5" == 5    // true (string is converted to number)
```

```
null == undefined // true
```

```
0 == false // true
```

`===` (Strict Equality):

Compares both value and type (no coercion).

Example:

```
"5" === 5    // false (string vs number)
```

```
0 === false // false (number vs boolean)
```

```
5 === 5      // true (number vs number)
```

- **Hoisting**

Hoisting means that variable and function declarations are moved to the top of their scope before code execution.

`var` declarations are hoisted and initialized as `undefined`

`let` and `const` are hoisted but not initialized (in Temporal Dead Zone)

Function declarations are fully hoisted

Example:

```
console.log(x); // undefined
```

```
var x = 10;
```

TOOLS USED

Visual Studio Code (VS Code)

Chrome Browser (JavaScript Console)

TASK

Practice writing different types of loops.

❖ for Loop

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

Output: 0 1 2 3 4

❖ while Loop

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

Output: 0 1 2 3 4

❖ do-while Loop

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

Output: 0 1 2 3 4

❖ for-of Loop (for arrays, strings)

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

Output: N a v n e e t

❖ for-in Loop (for objects)

```
let obj = { name: "Navneet", age: 19 };  
for (let key in obj) {  
  console.log(key + ": " + obj[key]);  
}
```

Output: name: Navneet

age: 19