JavaScript Numbers, NaN, -0 & Boxing (Notes)

These notes explain **Number conversion, NaN behavior, -0 edge cases, and boxing** in JavaScript.

1 `Number()` Conversion

```
console.log(Number('01254'));
```

☑ Output:

```
1254
```

Reason:

- `'01254'` is treated as a **decimal string** (not octal)

```
console.log(Number('abcd'));
```

✖ Output:

```
NaN
```

Reason:

- `'abcd'` cannot be converted into a number

---

2 What is NaN?

- NaN means **Not-a-Number**
- It represents an invalid numeric result

```
let x = NaN;
console.log(x == NaN);   // false
```

⚖ Rule:

> **NaN is never equal to anything, including itself**

---

3 `isNaN()` vs `Number.isNaN()`

◇ `isNaN()` (Global)

```
console.log(isNaN(x));        // true
console.log(isNaN('sanket')); // true
```

Why?

- `isNaN()` **converts input to a number first**
- `'sanket'` → `Number('sanket')` → `NaN`

◇ `Number.isNaN()` (Safe)

```
console.log(Number.isNaN('sanket')); // false
```

Why?

- Does **NO type coercion**
- Returns true **only if value is exactly NaN**

☑ Best practice: **Use** `Number.isNaN()`

---

4 `-0` in JavaScript

```
let n = -0;

console.log(n == 0);         // true
console.log(Object.is(n, 0)); // false
console.log(Object.is(n, -0));// true
```

Why this happens:

- `==` and `===` treat `0` and `-0` as equal
- `Object.is()` can **distinguish** between them

---

5 Checking Sign Using `Math.sign()`

```javascript
console.log(Math.sign(-3)); // -1
console.log(Math.sign(2));  // 1
```

Possible outputs:

Value	Result
positive	1
negative	-1
0	0
-0	-0
NaN	NaN

6 Boxing in JavaScript

✗ This does NOT work:

```javascript
// console.log(1.toString()); // SyntaxError
```

Reason:

- JS thinks `1.` is a decimal, not an object

☑ Correct Boxing Ways:

```javascript
console.log((1).toString());
console.log(Number(2).toString());
```

Explanation:

- Primitive numbers are **temporarily wrapped** into `Number` objects
- This is called **boxing**

🧠 Final Summary

Concept	Key Point
`Number()`	Converts string → number
`NaN == NaN`	Always false

Concept	Key Point
`isNaN()`	Converts input first
`Number.isNaN()`	No coercion
`-0`	Exists in JS
`Object.is()`	Detects `-0`
Boxing	Primitive → object temporarily

☑ Perfect for **Notion notes, revision, and interviews**