Search for “Session 4”

Topics: Unary Plus - NaN Type – Null Type

- ***Unary Plus***

The unary plus (+) operator is a unary operation that precedes its operand and attempts to convert it into a number, if it isn’t already. In JavaScript, for example, applying the unary plus to a string representing a number will convert the string to a numeric type. If the conversion isn’t possible, it results in NaN (Not-a-Number).

Example JS:

⇒ Code:

```
<script>
    let str = "42";
    let num = +str; // num is now 42 (number)
</script>
```

- ***NaN Type***

NaN stands for “Not-a-Number” and is a numeric value representing an undefined or unrepresentable value, especially in floating-point calculations. It’s part of the IEEE 754 floating-point standard. Operations that result in undefined numerical results, such as dividing zero by zero or taking the square root of a negative number, yield NaN.

Example JS:

⇒ code :

```
<script>
    let result = 0 / 0; // result is NaN
</script>
```

- ***Null Type***

  The null type represents the intentional absence of any object value. It's a primitive value in languages like JavaScript, where it signifies that a variable has been explicitly set to have no value. In Python, the equivalent is None, which is the sole instance of the NoneType and is used to signify 'no value' or 'nothing'.

Example JS:

⇒ code :

```
<script>
    let obj = null; // obj is explicitly set to have no val
</script>
```