Episode 6: `undefined` vs `not defined` in JS

Memory Allocation in JavaScript

- JavaScript allocates memory to variables and functions before executing any code.
- Even before a line of code is run, memory space is reserved for variables.
- The value of a variable that hasn't been assigned is `undefined`.
- `undefined` acts as a placeholder or default value in memory until a variable is assigned a different value.

Difference between `undefined` and `not defined`

- `undefined` means that memory has been allocated to a variable but no value has been assigned yet.
- `not defined` refers to a variable that has not been declared or allocated any memory.

`undefined` vs Empty

- undefined is not equivalent to empty or null.
- It is a special keyword in JavaScript that takes up its own memory space.
- undefined is a placeholder until a value is assigned to a variable.

Examples of `undefined`

```javascript
// Example 1
var a; // Memory is allocated for 'a', but no value is assigned yet
console.log(a); // Output: undefined

// Example 2
var x;
console.log(x); // Output: undefined

// Example 3
console.log(y); // Output: ReferenceError: y is not defined
```

- JS is a loosely typed / weakly typed language. It doesn't attach variables to any datatype. We can say var a = 5, and then change the value to boolean a = true or string a = 'hello' later on.

- Avoid assign undefined to a variable manually, Not a good practice. Watch Live On Youtube below: