01 . Into to javascrtipt

• Js is a plAnd it allows developers to add dynamic and interactive effects to any webpage.
• We also use it to manipulate the content or the CSS, load data from remote servers ,and really build entire applications in the browser,
• There has been a huge update to the language in 2015, which is officially called ES2015, but most people just call it ES6,

Conditional operator

```
syntax: condition:true
statement : false
statement
const age = 88;
age >= 18 ? consol e.l -
og( "you can") :conso -
le.l og ("you wont ");
```

Template literals

```
const firstname =
"Sathiya";
const firstname1 = " -
Sat hiy aPr iya n";
const $new = ${firs tname
} is a  guy  ;
consol e.l og( $new);
```

Template literals - use varible in a string , use multiple line is easy

p1-use stright mode

"use strict";
-> Use certian things and visible errors
-> and like that write more secure code. And when secure, I mean that strict mode makes it easier for us developers to avoid accidental error

Array methods

// Adds
push - add an element in a end
unshift(): Adds elements to the beginning of an array.
//Remove
pop - remove last element of an array
shift(): Removes and returns the first element of an array.
//Find index
indexOf(): Returns the index of the first occurrence of a specified element ; minus value mean does not exit
//Check array exit or not
includes(): Checks if an array contains a specified element. - immutable

Link js file

```
<script src="index.js">
</script>
```

Link from another page , Script in same page simple use script tag

Value and variable

Camal case

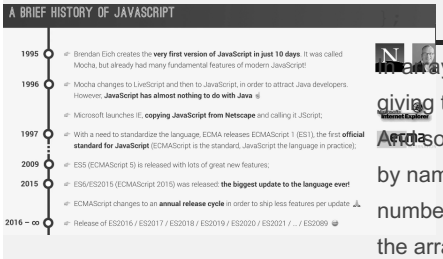
Type Conversion and Coercion

nan - not a valid number or not a number
Js - convert only string , boolen , number not other type

True valuse and false value

5 falsy values in js
0,'',undefined , null , nan

Version of javascript



p2-Function declaration and expression

```
"use strict";
// Function declar ation
function calAge 1(b -
irt hYear) {
    const age = 2029 -
birthYear;
    return age;
}
consol e.l og( cal Age -
1(2 003));
```

p2-Function declaration and expression (cont)

> // Anonymous function - expressions is a vaule

```
const me = function (birthYear) {
  const age = 2029 - birthYear;
  return age;
};
```

Object

```
// and it's called the
object literal Syntax
const spObject = {
    fir stName: " sat -
hiy a",
    las tName: " sp",
   age: 27,
    des tin ita tion:
" stu den t",
    habits: ["pr odu -
cti vit y", "time
manage men t"],
```

Arrays there is no way of giving these elements a name. And so we can't reference them by name, but only by their order number in which they appear in the array.

So we solve that problem we have another data structure in JavaScript, which is object

object is for unstructured data array is for ordered data

Sponsored by **Readable.com**
Measure your website readability!
https://readable.com

3 ways iof declaring a javascript

let , const is a modern way of declaring variable var is a old way of declarig of js

---

let age = 0 // Varaibale declare
age = 1 // it is a mutated variable or reasign variable

---

let vs const -> const is a best practice .
The reason for this is that it's a good practice
to have as little variable mutations or variable changes as possible because changing variables introduces a potential bug

---

var is completely avoid
or declare nothing - bad practice

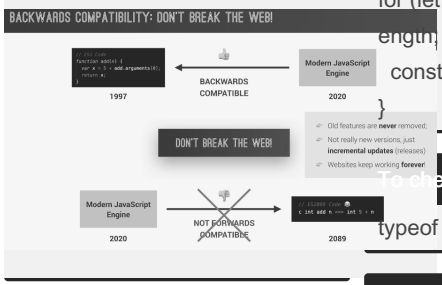
Equality operator

The strict equality operator (===) compares two values without type coercion

Equality operator (cont)

while the loose equality operator (==) compares values with type coercion in JavaScript.is full of really weird rules and behaviors. this can introduce many hard to find bugs into our code. clean code

Version-pic-2



p3-Arrow functions

```
syntax : let func =
(arg1, arg2, ..., argN)
=> expression
It is introuded in ES6
vesrion
```

Javascript dot vs bracket

https://codepen.io/pen/tour/welcome/start?editors=1010

Put into any via loops

```
const yearsfa = [1991,
2007, 1967, 2020];
const age = [];
const newArray = [];
function calAge ($var) {
    return 2023 - $var;
}
```

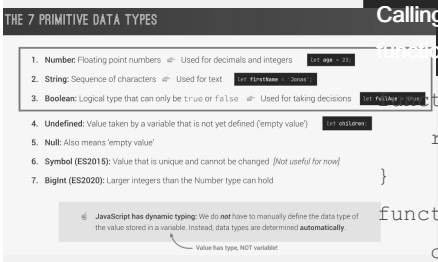
Put into any via loops (cont)

```
> for (let count = 0; count <=
years.length; count++) {
  console.log((age[count] =
calAge(years[count])));
  newArray.push(age);
}
console.log("--");
console.log(newArray);
console.log(calAge(2003));
for (let count = 0; count < array.l-
ength; count++) {
  const element = array[count];
}
```

To check what kind of data type

typeof

Data types



NULL is undefined data type

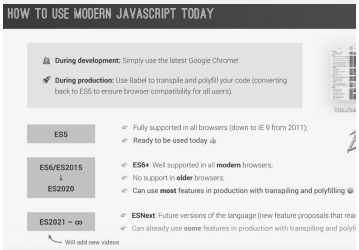
operator

Check c notes
Operator precedence So basically the order in which operators are executed.

Get input from webpage

```
const $colour =
prompt(`Tell your
most favourate colour`);
consol e.l og( $co -
lour);
```

Version-pic-3



https://kangax.github.io/compat-table/es6/

Calling function inside another functions

```
tion cutFruit(fruit) {
    return fruit * 4;
}
function fruitP roc ess or
    const appleP = cutFru
    const orangesP = cutFr
    const juice =Juice witl
{oranges} oranges;
    return juice;
}
```


Sponsored by **Readable.com**
Measure your website readability!
https://readable.com