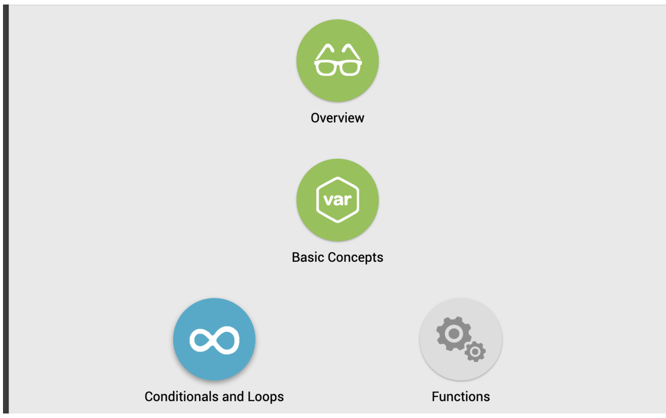
basic concepts(2/8)

https://www.w3schools.com/js/js_datatypes.asp



var

- 변수는 data 값들을 저장하는 컨테이너
- 변수의 값은 프로그램 전역에서 바뀔 수 o
- var 키워드를 사용한다. 쩐에 봤던 스칼라랑 같음 → 변수 타입 지정할 필요 ㄴ~~
- JavaScript statements are separated by semicolons.

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript</h2>

<p>JavaScript evaluates expressions from left to right. Different sequences can produce different results:</p>

<p id="demo"></p>

<script>
var x = "Volvo" + 16 + 4;
document.getElementById("demo").innerHTML = x;
</script>

</body>
</html>
```

JavaScript

JavaScript evaluates expressions from left to right. Different sequences can produce different results:

Volvo164

변수 이름 붙이기 규칙

- 첫번째 글자는 무조건 글자거나 _ 거나 $여야함. 숫자놉
- 다른 특수문자나 연산자 놉 하이픈도 놉
- 띄어쓰기 놉→ 뭐 알아서 빨간줄 그어주겠지

number operations

> JavaScript numbers are always stored as double precision floating point numbers.

Essentially, floating-point makes it easier for the coder and the code to understand very large and very small numbers. For example, when you have a number like 1,000,000,000.001, it's so big that the .001 doesn't really matter. So, instead of clogging up the code, the .001 "floats" off so you only have 1,000,000,000.

→엥 근데 저 숫자 돌려봐도 소숫점까지 나오는디... 여튼 숫자를 64bit 부동소숫점으로 저장한다고 한다.

You can get the result of a string expression using the eval() function, which takes a string expression argument like eval("10 * 20 + 8") and returns the result. If the argument is empty, it returns undefined.

```
var a = '2';
var n = eval("5*a"); //a가 '2' 여도 된다! 당연한건가?
document.write(n);
//output: 10
```

10 * '5' or '10' * '5' gives the same result. Multiplying a number with string values like 'sololearn' * 5 returns NaN (Not a Number).

```
var a = 'aa';
var n = eval("5*a");
document.write(n);
//ouput: NaN
```

' ' & " "

> "this is an 'apple'" is right
> "this is an ' apple" is right
> "this is an "apple"" is wrong

> You can also
> var text = "My name is \"John\" ";
> or
> var text = 'My name is \'John\' ';
> To Get:
> My name is "John"
> My name is 'John'

→역슬래시 이다!!!

boolean type

- The Boolean value of 0 (zero), null, undefined, empty string is **false**
- Everything with a "real" value is **true**

신기한거

you can use multiple assignment operators in one line, such as x -= y += 9.

var a_number = 5;
var a_string = "5";
a_number == a_string; → true
a_number === a_string; → false

= : assignment.
== : only compares values.
=== : compares values and type.

logical operations - Conditional Operator - 삼항연산자 같은거

Another JavaScript conditional operator assigns a value to a variable, based on some condition.

Syntax:

variable = (condition) ? value1: value2

For example:

var isAdult = (age < 18) ? "Too young": "Old enough";

condition : true → isAdult = Too young / false → Old enough

string operations

> Numbers in quotes are treated as strings: "42" is not the number 42, it is a string that includes two characters, 4 and 2.

```
//The str.repeat() method is handy too:

var mystring = "Ha! ";
document.write(mystring.repeat(3));

// output: Ha! Ha! Ha!
```

```
document.write(1+2+"Hello"+1+2)

//Output: 3Hello12
```