



SCOPE & OBJECTS

VAR, LET, & CONST

keyword	local scope	mutable	browsers
var	function only	yes	all
let	any block	yes	modern
const	any block	no	modern

FUNCTIONS & HOISTING

type	name hoisted	content hoisted
declaration	yes	yes
let/const expression	no	no
var expression	yes	no

GLOBAL, LOCAL, & BLOCK SCOPE

a variable declared outside of a function or block is in the **global scope**

```
let temp = 75;
```

```
function predict() {  
  let forecast = 'Sun';  
  console.log(temp + ' and ' + forecast);  
  // 75 and Sun  
}
```

a variable declared within a function is in the **function scope** of that function

```
if (temp > 70) {  
  let forecast = 'It's gonna be warm!';  
  console.log(temp + '! ' + forecast);  
  // 75! It's gonna be warm!  
}
```

a variable declared with let and within a block, such as an if statement, is in the **block scope** of that statement

OBJECT SYNTAX

```
let favorites = {  
  fruit: "apple",  
  vegetable: "carrot",  
  greeting: function() {  
    console.log("I like fruit and veg");  
  },  
  sayFaves: function() {  
    console.log("I like " + this.fruit +  
      "s and " + this.vegetable);  
  },  
};
```

property

method