

# Objects and Classes

08 March 2023 14:03

## I. Objects(definition, properties, and methods)

**Objects** -> Structure of related data or functionality, contains values accessed by string keys, data values are called properties, function values are called methods

Object	
'name'	'Peter'
'age'	20

- We can create an object with an object literal

```
let person = { name:'Peter', age:20, height:183 };
```

- We can define an empty object and add properties later

```
let person = {};  
person.name = 'Peter';  
person.age = 20;  
person.hairColor = 'black';
```

- Functions within a JavaScript object are called **methods**.

```
let person = {  
  sayHello: function() {  
    console.log('Hi, guys');  
  }  
};
```

```
let person = {  
  sayHello() {  
    console.log('Hi, guys');  
  }  
};
```

```
Object.entries(cat); // [['name', 'Tom'], ['age', 5]]
```

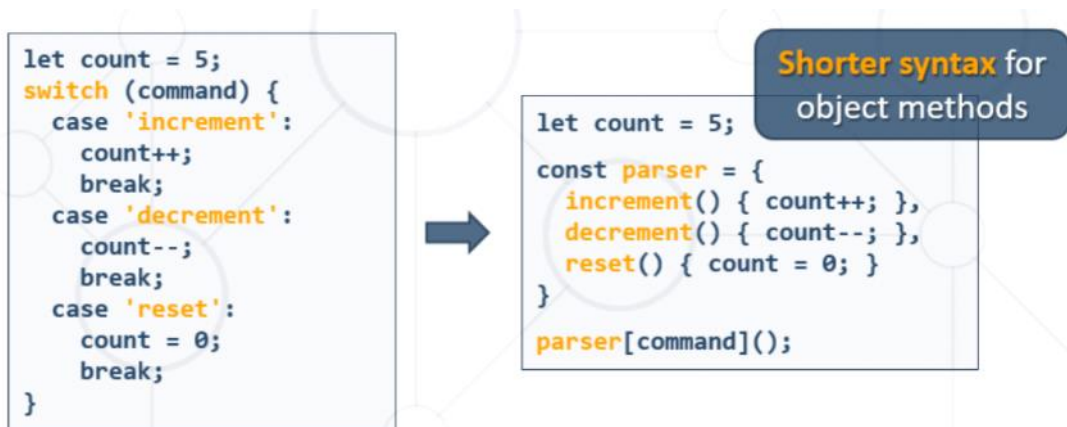
```
Object.values(cat); // ['Tom', 5]
```

```
Object.keys(cat); // ['name', 'age']
```

# Reference vs. Value Types

Stack -> Value -> Boolean, String, Number, null, undefined, Symbol, BigInt

Heap -> Reference -> Array, Objects and Function



## JSON - > Java Script Object Notation

- Open-standard file format that uses text to transmit data objects
- Exchange data between browser and server
- JSON is a lightweight format compared to XML

Brackets define a JSON

Keys are in double quotes

Keys and values separated by :

```
{
  "name": "Ivan",
  "age": 25,
  "grades": {
    "Math": [2.50, 3.50],
    "Chemistry": [4.50]
  }
}
```

It is possible to have nested objects

In JSON we can have arrays

- We can convert object into **JSON** string using **JSON.stringify(object)** method

```
let text = JSON.stringify(obj);
```

- We can convert JSON string into object using **JSON.parse(text)** method

```
let obj = JSON.parse(text);
```