Objects

Class	
Created	@Jan 16, 2021 4:42 PM
Materials	
Reviewed	☑
Type	

Property names can be any string.

Property values can be strings, numbers, booleans, null, undefined, objects, arrays, and functions.

When you access a property of an object that does not exist, the value that is returned is undefined.

IN OPERATOR

To test whether a property exists on an object, you can use the in operator.

```
var obj = {
  prop: undefined
};

obj.prop; //undefined
obj.prop2; //undefined

'prop' in obj; //true
'prop2' in obj; //false
```

DELETE OPERATOR

An expression with the delete operator will return true if the property can be deleted and false if it cannot.

```
var obj = {
  prop: 'whatever'
};

delete obj.prop;
```

CREATING OBJECTS

```
var obj1 = {};
var obj2 = Object();
var obj3 = new Object();

obj1; //{}
obj2; //{}
obj3; //{}
```

Another way to create objects is to use the `Object.create` method.The Object.create method accepts two parameters. The first is an object that is to serve as the prototype of the object you are creating **(if you do not want your object to have a prototype you can pass null**).

```
var obj = Object.create(null, {
  city: {
    value: 'Berlin'
  }
});

delete obj.city; //false (error in strict mode)

obj.city = 'Dallas'; //error in strict mode
```

CHECK PROPERTY AND KEY

```
console.log("city" in person);
console.log("all my secrets" in person);

console.log(person.hasOwnProperty("hatesRawTomatoes"));

// accessing properties/keys and values inside of objects
for (var key in person) {
    console.log("key:", key, "value:", person[key]);
}
```