Objects

Overview

```
- what is an object?
- why are objects useful?
- typeof object
- accessing, adding, changing, deleting values
- in operator
- for...in loop
- Object.keys()
- nested arrays and objects
```

What is an object?

```
/* an object is a collection of key-value pairs */
   /* like arrays, objects store values, but instead of storing them in
      numeric "indices", objects store values in string "keys" */
   let myArray = ['value1', 'value2'];
   let myObject = {
    'key1': 'value1',
   'key2': 'value2'
11 };
13 console.log(myArray[0]);
14 console.log(myObject['key1']);
```

Why are objects useful?

```
/* consider representing Pusheen the Cat as an array */
   let pusheen = ['Pusheen', 7, 'gray and tabby'];
   /* an array is a good place to hold an ordered list of values, but it
      doesn't store any information about what those values represent */
   /* an object's string keys allows objects to store more information about
      the values within it */
   let pusheen = {
    'name': 'Pusheen',
   'age': 7,
'color': 'gray and tabby'
14 };
```

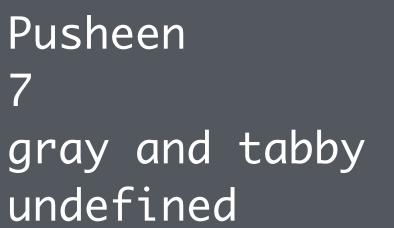


typeof object

```
console.log(typeof {});
```

Creating an object

```
/* create a new object using curly braces */
   /* an object's keys are always strings; you can omit the quotation
      marks */
   let pusheen = {
     name: 'Pusheen',
     age: 7,
     color: 'gray and tabby'
10 };
```



Accessing a value

```
/* use bracket notation to access a value */
   /* pass a string into the brackets that corresponds with a key in the
   object */
   let pusheen = {
   name: 'Pusheen',
     age: 7,
     color: 'gray and tabby'
  };
10
   console.log(pusheen['name']);
   console.log(pusheen['age']);
13 console.log(pusheen['color']);
14 console.log(pusheen['notAKeyInTheObject']);
```

Accessing a value

```
/* any variable or expression that evaluates to a string can be passed
      into the brackets */
   let pusheen = {
     name: 'Pusheen',
    age: 7,
     color: 'gray and tabby'
  };
  let keyToCheck = 'name';
   console.log(pusheen[keyToCheck]);
13 console.log(pusheen['col' + 'or']);
```

Pusheen 7 gray and tabby undefined

Accessing a value

```
/* you can also use dot notation to access values */
   let pusheen = {
     name: 'Pusheen',
     age: 7,
     color: 'gray and tabby'
   };
   let keyToCheck = 'name';
10
   console.log(pusheen.name); // no quotes needed with dot notation
   console.log(pusheen.age);
13 console.log(pusheen.color);
14 console.log(pusheen.keyToCheck);
```



```
/* use bracket notation or dot notation to add a
      key/value pair */
   let pusheen = {
     name: 'Pusheen',
     age: 7,
     color: 'gray and tabby'
   };
   pusheen['sister'] = 'Stormy';
   pusheen.brother = 'Pip';
13 console.log(pusheen);
```

```
name: Pusheen,
age: 7,
color: gray and tabby,
sister: Stormy,
brother: Pip
}
```

Changing a value

```
/* use bracket notation or dot notation to change a value */
let pusheen = {
  name: 'Pusheen',
 age: 7,
color: 'gray and tabby'
};
pusheen['age'] = 8;
pusheen.age++
console.log(pusheen.age);
```

Deleting a key/value pair

```
/* use the delete keyword to delete a key/value pair */
let pusheen = {
  name: 'Pusheen',
 age: 7,
color: 'gray and tabby'
};
delete pusheen['age'];
delete pusheen.color;
console.log(pusheen);
```

in operator

```
/* use the in operator to check if a key is in the object */
let pusheen = {
  name: 'Pusheen',
 age: 7,
 color: 'gray and tabby'
};
console.log('name' in pusheen);
console.log('sadness' in pusheen);
```

for...in loop

```
/* use the for…in loop to loop through all of the keys in an object */
   let pusheen = {
     name: 'Pusheen',
     age: 7,
    color: 'gray and tabby'
   };
   for (let key in pusheen) {
     console.log(key);
10
```





Pusheen's name is Pusheen
Pusheen's age is 7
Pusheen's color is gray and tabby

```
/* use the for…in loop to loop through all of the keys in an object */
   let pusheen = {
     name: 'Pusheen',
    age: 7,
   color: 'gray and tabby'
   };
   for (let key in pusheen) {
     console.log("Pusheen's", key, 'is', pusheen[key]);
10
```

for...in loop

```
/* use the for…in loop to loop through all of the keys in an object */
   let pusheen = {
     name: 'Pusheen',
    age: 7,
   color: 'gray and tabby'
   };
   for (let key in pusheen) {
     console.log("Pusheen's", key, 'is', pusheen.key);
10
```

Object.keys()

```
/* use Object.keys() to get an array of the keys in the object */
   let pusheen = {
     name: 'Pusheen',
    age: 7,
    color: 'gray and tabby'
   };
   console.log(Object.keys(pusheen));
10
```

nested arrays

```
/* objects can store any type of value, including arrays and other
   objects */
let pusheen = {
  name: 'Pusheen',
 age: 7,
  colors: ['gray', 'tabby']
};
console.log(pusheen.colors[0]);
console.log(pusheen.colors[1]);
```

nested objects

```
/* objects can store any type of value, including arrays and other
      objects */
   let pusheen = {
     name: 'Pusheen',
     age: 7,
     siblings: {
       sister: 'Stormy',
       brother: 'Pip'
13 console.log(pusheen.siblings.sister);
14 console.log(pusheen.siblings.brother);
```

Recap

```
- what is an object?
    - why are objects useful?

    typeof object

    - accessing, adding, changing, deleting values
   - in operator
   - for...in loop
   - Object.keys()
    - nested arrays and objects
10 */
```