




LEARN JAVASCRIPT

 Dive Deep into JavaScript Objects!



Exercise JavaScript Objects!

Exercise: Creating and Accessing Object Properties	2
Exercise: Modifying Object Properties	3
Exercise: Adding New Properties to an Object	3
Exercise: Deleting a Property from an Object	3
Exercise: Iterating Over Object Properties	4
Exercise: Checking Property Existence	4
Exercise: Nested Objects	4
Exercise: Copying an Object	5
Exercise: Merging Objects	6
Exercise: Object Keys and Values	6



Below are some interactive coding exercises focused exclusively on JavaScript Objects. Whether you're just starting out or refining your existing skills, these exercises are designed to strengthen your understanding and mastery of JavaScript objects. 

We've explored various aspects of working with objects:

- Creating and Accessing Properties
- Modifying Properties
- Adding New Properties

Learn more about JavaScript with Examples and Source Code Laurence Svekis
Courses <https://basescripts.com/>

- Deleting Properties
- Iterating Over Properties
- Property Existence Check
- Working with Nested Objects
- Copying Objects
- Merging Objects
- Extracting Object Keys and Values

Each exercise is accompanied by a solution and a detailed explanation, making it perfect for both learning and teaching.  

Exercise: Creating and Accessing Object Properties

Problem: Create an object named car with properties make, model, and year. Then, access and log the model property.

Solution:

```
let car = {  
  make: 'Toyota',  
  model: 'Camry',  
  year: 2020  
};  
console.log(car.model); // Output: Camry
```

Explanation: This exercise demonstrates the creation of an object and accessing one of its properties using dot notation.

Learn more about JavaScript with Examples and Source Code Laurence Svekis
Courses <https://basescripts.com/>

Exercise: Modifying Object Properties

Problem: Modify the year property of the car object to 2021.

Solution:

```
car.year = 2021;
```

```
console.log(car.year); // Output: 2021
```

Explanation: Object properties can be modified directly using dot notation.

Exercise: Adding New Properties to an Object

Problem: Add a new property color to the car object.

Solution:

```
car.color = 'black';
```

```
console.log(car.color); // Output: black
```

Explanation: New properties can be added to an object simply by assigning a value to them using dot notation.

Exercise: Deleting a Property from an Object

Problem: Remove the make property from the car object.

Solution:

```
delete car.make;
```

```
console.log(car.make); // Output: undefined
```

Explanation: The delete operator removes a property from an object.

Exercise: Iterating Over Object Properties

Problem: Write a loop that logs all properties and their values of the car object.

Solution:

```
for (let key in car) {  
  console.log(key + ': ' + car[key]);  
}
```

Explanation: The for...in loop is used to iterate over all enumerable properties of an object.

Exercise: Checking Property Existence

Problem: Check if the car object has a property named model.

Solution:

```
console.log('model' in car); // Output: true
```

Explanation: The in operator checks whether an object has a specified property.

Exercise: Nested Objects

Problem: Create an object person with properties name and address, where address is an object with properties street, city, and zipCode.

Solution:

```
let person = {  
  name: 'Alice',
```

Learn more about JavaScript with Examples and Source Code Laurence Svekis
Courses <https://basescripts.com/>

```
address: {
  street: '123 Main St',
  city: 'Anytown',
  zipCode: '12345'
}
};
console.log(person.address.city); // Output: Anytown
```

Explanation: This demonstrates how to create nested objects and access properties within them.

Exercise: Copying an Object

Problem: Create a shallow copy of the person object.

Solution:

```
let personCopy = Object.assign({}, person);
console.log(personCopy);
```

Explanation: `Object.assign` is used to create a shallow copy of an object.

Exercise: Merging Objects

Problem: Merge two objects, `person` and `contactDetails` (with properties `email` and `phone`), into a new object.

Solution:

```
let contactDetails = {
```

```
email: 'alice@example.com',  
phone: '1234567890'  
};  
let mergedObject = Object.assign({}, person, contactDetails);  
console.log(mergedObject);
```

Explanation: `Object.assign` can also be used to merge multiple objects into a new object.

Exercise: Object Keys and Values

Problem: Get all keys and values from the person object separately and log them.

Solution:

```
let keys = Object.keys(person);  
let values = Object.values(person);  
console.log('Keys:', keys); // Output: Keys: ['name', 'address']  
console.log('Values:', values); // Output: Values: ['Alice', {...}]
```

Explanation: `Object.keys` and `Object.values` are used to get an array of an object's keys and values, respectively.