




Log In

Join

 **answers** Trusted answers to developer questions

# Dot vs. Bracket notation when accessing a JavaScript object

Programming Bytes 

We can access the property of an object by:

- Dot notation
- Bracket notation

## Dot notation

**Dot notation** makes code more readable. It is the most popular way to access the property of an object.

Syntax: `obj.property_name`

```
1 var user = {name : "Mark"};  
2 user.name ; // "Mark"
```



## When not to use dot notation

Consider that we have a property named `123` :

```
var obj = {  
  '123' : 123  
};
```



In the above case, we cannot access this using `obj.123`; because an identifier that begins with a number is not a valid dot notation identifier.

```
obj.123;
```

So, if the property name is not a valid identifier, we cannot access its value using `.` notation.

In this case, we can use bracket notation:

```
var obj = {  
  '123' : 123  
};  
obj['123']; // 123
```

In JavaScript, `$` and `_` are valid identifiers. Therefore, we can access them properties using `.` notation.

```
var obj = {  
  $ : 10,  
  _ : 20  
}  
obj.$; // 10  
obj._; // 20
```

## Bracket notation

**Bracket notation** is used when the property name is an invalid dot notation identifier.

Syntax: `obj[property_name]`

```
var obj = {  
  test-123 : "test"
```



```
}  
// in this case we cannot use dot notation  
obj['test-123']; // "test"
```

If the property name is a whole number, then we don't need to wrap the name inside single/double quotes. If the property name is a double, then we need to wrap the property name inside single/double quotes.

## Example 1: Whole number

```
var obj = {  
  123456 : 10  
}  
obj[123456]; // 10
```

## Example 2: Double

```
var obj = {  
  123.456 : 10  
}  
obj[123.456]; // undefined  
obj['123.456']; // 10
```

## Example 3: Using an invalid number

```
var obj = {  
  '123.123.123' : 10  
};  
obj['123.123.123']; // 10
```

## Example 4: Using special symbols

```
var obj = {  
  '123-test' : "test"
```



```
}  
obj[123-test]; // error (test is not defined)  
obj['123-test']; // "test"
```

## Using a variable as the property name

If the object's key value is only known at runtime, then we need to use bracket notation.

Example:

```
var obj = {  
  name : "Mark",  
  age : 20  
}  
var name = "age";  
obj[name]; // 20  
obj["name"]; // Mark
```

We can also use an object as the property name, but that will be converted into [object Object].

```
var a = {};  
var b = {};  
var c = {};  
c[a] = 10;  
c ; // {[object Object]: 10}  
c[b] = 20; // this will replace old [object Object] value  
c; // {[object Object]: 20}
```

We can also have an empty string as the property name.

```
var obj= {};  
var emptyString = "";  
obj[emptyString] = "10";  
obj[emptyString]; // "10"
```



## RELATED TAGS

javascript

communitycreator

## CONTRIBUTOR

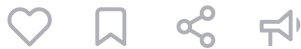
Programming Bytes



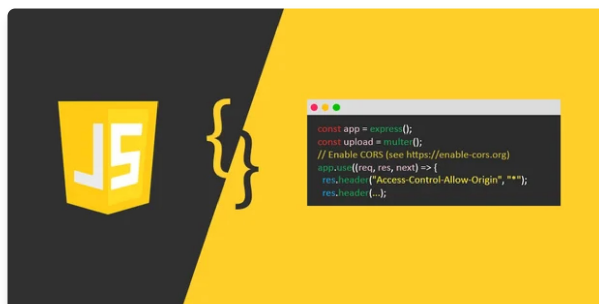
## Attributions:

1. Dot vs Bracket Notation in Accessing JavaScript Object. by Javascript Jeep

License: Creative Commons -Attribution -  
ShareAlike 4.0 (CC-BY-SA 4.0)



## Related Courses



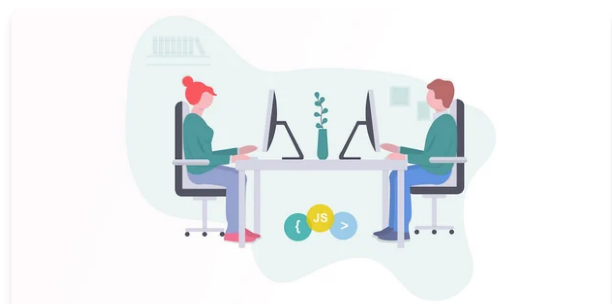
Baptiste Pesquet

## Complete JavaScript Course: Build a Real World App from Scratch



Beginner

Preview →



Zsolt Nagy

## JavaScript in Practice: Getting Started



Beginner

Preview →



## Keep Exploring

[How to disable a button in JavaScript](#)

[What is Object.values in JavaScript?](#)

[How to keep your screen awake using JavaScript](#)

[Learn in-demand tech skills in half the time](#)

---

### SOLUTIONS

[For Enterprise](#)

[For Individuals](#)

[For HR & Recruiting](#)

[For Bootcamps](#)

### PRODUCTS

[Educative Learning](#)

[Educative Onboarding](#)

[Educative Skill Assessments](#)

[Educative Projects](#)

---

### PRICING

[For Enterprise](#)

[For Individuals](#)

### LEGAL

[Privacy Policy](#)

[Terms of Service](#)



[Free Trial](#)[Business Terms of Service](#)

## CONTRIBUTE

[Become an Author](#)[Become an Affiliate](#)[Become a Contributor](#)

## RESOURCES

[Educative Blog](#)[EM Hub](#)[Educative Sessions](#)[Educative Answers](#)

## ABOUT US

[Our Team](#)[Careers](#) [Hiring](#)[Frequently Asked Questions](#)[Contact Us](#)[Press](#)

## MORE

[GitHub Students Scholarship](#)[Course Catalog](#)[Early Access Courses](#)[Earn Referral Credits](#)[CodingInterview.com](#)

Copyright ©2022 Educative, Inc. All rights reserved.

