



链滴

# 给定的多层键是否存在 JSON 对象中

作者: [Vanessa](#)

原文链接: <https://ld246.com/article/1595648916763>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

## 描述

如果一个 JSON 对象中存在给定的目标键值就返回 `true`，否则的话返回 `false`。

## 提示

- 首先检查给定的 `keys` 是否为一个非空数组，如果为空的话，则提前返回 `false`
- 使用 `Array.prototype.every()` 按顺序依次检查给定的每一个键是否都存在于 `obj` 对象的深层次中
- 使用 `Object.prototype.hasOwnProperty()` 检查 `obj` 是否没有拥有当前的键，使用 `typeof` 检测 `obj` 是否不是一个对象。如果满足其中任何一种情况，则停止解析，返回 `false`
- 否则将键值赋值给 `obj`，以便在下一次迭代中继续使用

## 代码

```
const hasKey = (obj, keys) => {
  return (
    keys.length > 0 &&
    keys.every(key => {
      if (typeof obj !== 'object' || !obj.hasOwnProperty(key)) return false;
      obj = obj[key];
      return true;
    })
  );
};
```

## 示例

检测 JSON 对象中是否存在指定的多层键：

```
let obj = {
  a: 1,
  b: { c: 4 },
  'b.d': 5
};
hasKey(obj, ['a']); // true
hasKey(obj, ['b']); // true
hasKey(obj, ['b', 'c']); // true
hasKey(obj, ['b.d']); // true
hasKey(obj, ['d']); // false
hasKey(obj, ['c']); // false
hasKey(obj, ['b', 'f']); // false
```

## 返回总目录

[每天 30 秒系列之 JavaScript 代码](#)