

JavaScript Object Exercises

Basic

Khai báo một object dog mô tả về một con chó với các thông tin sau:

- name: Tên
- age: Tuổi
- breed: Giống chó
- gender: Giới tính
- color: Màu lông
- legs: Số chân
- favorite food: Thức ăn yêu thích

Và các phương thức:

- eat(): In ra console thông báo theo mẫu {name} is eating {favorite food}, yum...yum...
- run(): In ra console thông báo theo mẫu {name} is running on {legs} legs...
- bark(): In ra console thông báo theo mẫu Woof woof woof ...

Thực hiện các yêu cầu sau đây, với mỗi yêu cầu về thêm/sửa/xóa thông tin đều in ra console để xem kết quả:

- In dog ra console
- Viết câu lệnh để in name ra console
- Viết câu lệnh để in favorite food ra console
- Viết câu lệnh để thay đổi color thành Brown
- Viết câu lệnh để tăng age thêm 2 tuổi
- Viết câu lệnh gọi phương thức eat()
- Viết câu lệnh gọi phương thức run()
- Sử dụng vòng lặp để in ra tất cả tên thuộc tính (key) ra console
- Sử dụng vòng lặp để in ra tất cả giá trị (value) ra console
- Viết câu lệnh để thêm một thuộc tính isCrazy vào dog với giá trị boolean tùy ý
- Viết câu lệnh để thêm một phương thức crazy() vào dog, kiểm tra nếu isCrazy là false thì in ra theo mẫu {name} is not crazy, nếu isCrazy là true thì sử dụng vòng lặp để gọi 2 phương thức run() và bark() 5 lần (con chó bị điên), kết quả sẽ in ra console dạng {name} is running on {legs} legs ... Woof woof woof ... {name} is running ... Woof

woof woof

- Viết câu lệnh đảo ngược giá trị của `isCrazy`
- Viết câu lệnh gọi phương thức `crazy()`
- Viết câu lệnh xóa `isCrazy` khỏi `dog`
- Viết câu lệnh gọi phương thức `crazy()`, chú ý kết quả in ra và comment lý do vì sao lại in ra như vậy
- Sử dụng vòng lặp để in ra tất cả thuộc tính và giá trị tương ứng (không phải phương thức) trong `dog` ra console (sử dụng `typeof` để kiểm tra kiểu) theo mẫu `{key} có giá trị là {value}`
- Viết câu lệnh khai báo một biến mới `clone` và gán giá trị của `dog` cho `clone`
- Viết câu lệnh chỉnh sửa `name` trong `clone` thành một tên khác bất kỳ
- Viết câu lệnh in `clone` và `dog` ra console, chú ý kết quả
- Viết câu lệnh khai báo một object mới `copy` rỗng `{}`
- Sử dụng vòng lặp để sao chép từng thuộc tính và giá trị (bao gồm cả phương thức) trong `dog` sang `copy`
- Viết câu lệnh chỉnh sửa `name` trong `copy` thành tên khác bất kỳ
- Viết câu lệnh in `copy` và `dog` ra console, chú ý kết quả

Intermediate

1. Cho một object `guess_list` chứa thông tin sinh viên với `key` là tên và `value` là quốc gia của sinh viên đó

```
1 const guess_list = {  
2   Randy: "Germany",  
3   Karla: "France",  
4   Wendy: "Japan",  
5   Norman: "England",  
6   Sam: "Argentina"  
7 };
```

Viết hàm `greeting(name)` nhận một giá trị đầu vào `name` là một tên bất kỳ, kiểm tra nếu tên có trong danh sách hay không và in ra chuỗi dạng: `Hi, I'm [name], and I'm from [county]!`. Nếu tên không có trong danh sách thì in ra `Hi! I'm a guess.`

2. Viết hàm `afterNYears(family, years)` nhận hai giá trị đầu vào:
 - `family` là object chứa thông tin tên (`key`) và tuổi (`value`) của các thành viên trong một gia đình
 - `years` là một số (năm) nguyên dương bất kỳ

Tính và trả về kết quả là một object mới cho biết sau `years` năm thì những thành viên trong gia đình sẽ bao nhiêu tuổi

Ví dụ:

```
1 const family = {  
2   Susan: 32,  
3   Fred: 34,  
4 }  
5  
6 afterNYears(family, 5); // { Susan: 37, Fred: 39 }
```

3. Cho một object chứa key là các chữ cái trong bảng chữ cái và value là một số điểm tương ứng như sau:

```
1 const scores = { A: 100, B: 14, C: 9, D: 28, E: 145, F: 12, G: 3, H:  
10, I: 200, J: 100, K: 114,  
2   L: 100, M: 25, N: 450, O: 80, P: 2, Q: 12, R: 400,  
3   S: 113, T: 405, U: 11, V: 10,  
   W: 10, X: 3, Y: 210, Z: 23 };
```

Viết hàm `calcNameScore(name)` nhận một giá trị đầu vào `name` là một tên bất kỳ (không dấu, viết hoa), tính điểm và in ra kết quả ứng với từng ký tự trong tên theo quy tắc như sau:

- $score \leq 60$: NOT TOO GOOD
- $61 \leq score \leq 300$: PRETTY GOOD
- $301 \leq score \leq 600$: VERY GOOD
- $score > 600$: THE BEST

4. Viết hàm `invert(object)` nhận một giá trị đầu vào có key và value là các số bất kỳ (value là duy nhất), trả về kết quả là một object mới đảo ngược value và key của object ban đầu

Ví dụ:

```
1 const obj = { a: 1, b: 2, c: 3};  
2  
3 invert(obj); // {1: "a", 2: "b", 3: "c" }
```

5. Viết hàm `totalAmount(items)` nhận một giá trị đầu vào `items` là object có key là tên món đồ và value là giá trị tương ứng, tính và trả về kết quả là tổng giá trị của những món đồ

Ví dụ:

```
1 const items = {  
2   tv: 300,  
3   ipad: 1000  
4 };  
5  
6 totalAmount(items); // 1300
```

6. Viết hàm `expensiveItems(items)` nhận một giá trị đầu vào `object` có key là tên item và `value` là giá trị tương ứng, kiểm tra và trả về kết quả là một `object` mới chứa các item có giá trị lớn hơn hoặc bằng 1000

Ví dụ:

```
1 const items = {  
2   "Gucci Fur": 3000,  
3   "Icecream": 4,  
4   "Dolce Gabana Heels": 4000,  
5   "Humbergur": 10  
6 };  
7  
8 expensiveItems(items); // { "Gucci Fur": 3000, "Dolce Gabana Heels":  
   4000 }
```

Advanced
