

Objects and Composition

[Назад към каталога](#)[Частичен тест](#)

1. Обектите в JavaScript са:

- ☐ Линейна структура от данни
- ☐ Друго име за клас
- ☒ Сбор от двойки ключ-стойност, наречени свойства
- ☐ Наименувана поредица от операции

2. Какъв тип данни са обектите?

- ☐ Примитивен
- ☐ Сложен
- ☒ Референтен

3. Посочете правилната декларация на обект:

- ☒ `const myObj = {a: 1, b: 2, c:3}`
- ☐ `const myObj = [1, 2, 3]`
- ☐ `const myObj = {"a", "b", "c"}`

4. Ключа на свойството може да бъде string, съдържащ всякакви символи.

- ☒ Вярно
- ☐ Невярно

5. Структурата на обектите в JavaScript трябва да се дефинира предварително и не може да бъде променяна.

- ☐ Вярно

☒ Невярно

6. **Операцията** `const {name,age} = person` **се нарича:**

☒ Деструктуриране

☐ Дерекфериране

☐ Няма такава операция в JavaScript

☐ Експлозия

7. **Метода** `Object.keys()` **върща:**

☐ Масив от стойностите на обекта

☒ Масив от ключовете на обекта

☐ Масив от двойки ключ-стойност на обекта, оформени като масиви

8. **Кой оператор е предназначен за обхождане ключовете на обект?**

☒ `for-in`

☐ `for-key`

☐ `for-of`

9. **Асоциативните масиви в JavaScript са обикновени обект.**

☒ Вярно

☐ Невярно

10. **Попълнете липсващия фрагмент, за да се отпечата описания в коментара резултат:**

```
1. function spawn(name) {  
2.   return {  
3.     name,  
4.     greet() {
```

```
5.     console.log(`${   .name} says hello`);
6.   }
7.   };
8. }
9.
10. function start() {
11.   const person = spawn('Jonny');
12.   person.greet();
13. }
14.
15. start();
16. // Jonny says hello
```

(въведете верния отговор)

Правилен отговор: this

11. Към какво сочи референцията this при изпълнение на метод на обект?

- ☒ Към обекта, от който е изпълнен метода
- ☐ Не съществува такава референция
- ☐ Към изпъления метод
- ☐ Към родителя на обекта

12. Функциите в JavaScript могат да бъдат свободно добавяни като методи към обекти.

- ☒ Вярно
- ☐ Невярно

13. Какво ще се отпечата при изпълнението на следния код:

```
1. const car1 = {
2.   model: 'Race Car',
3.   start() {
4.     console.log(`${this.model} goes vroom!`);
5.   }
6. };
```

```
7.  
8. const car2 = {  
9.   model: 'Station Wagon',  
10.  start: car1.start  
11. };  
12.  
13. car2.start();
```

- ☐ Race Car goes vroom!
- ☒ Station Wagon goes vroom!
- ☐ Ще получим грешка при изпълнение
- ☐ undefined goes vroom!

14. Всяка функция в JavaScript запазва първоначалния си контекст, дори когато е извадена и изпълнена извън него.

- ☐ Вярно
- ☒ Невярно

15. Обектите в JavaScript могат да съдържат в себе си други обекти.

- ☒ Вярно
- ☐ Невярно

16. Как наричаме техниката на комбиниране на прости обекти и методи в по-сложни обекти?

- ☐ Асемблиране
- ☐ Симфония
- ☐ Инстанциране
- ☒ Композиция

17. Factory функция в JavaScript наричаме функция, която:

- ☒ създава и връща обекти

- ☐ добавя функционалност към съществуващи обекти
- ☐ осъществява наследяване между обекти

18. С какъв термин можем да наречен следната функция:

```
1. function canPrint(device) {  
2.   device.print = function print() { /*...*/ };  
3.   return device;  
4. }
```

- ☐ Конструктор
- ☐ Деструктор
- ☐ Принтер
- ☒ Декоратор

19. Какво е едно от основните приложения на формата JSON?

- ☐ Участие в древногръцки митове
- ☐ Дълбоко копиране на референтни типове
- ☒ Обмен на данни в интернет мрежата

0 / 19 верни отговора