

JavaScript

Test 11

grishaharutyunian@gmail.com [Сменить аккаунт](#)



Совместный доступ отсутствует

***Обязательный вопрос**

Name Surname *

Мой ответ

1. What is the `[[Prototype]]` in JavaScript?

5 баллов

- ☐ A reference to another object
- ☐ Prototype link
- ☐ Hidden prototype
- ☐ Private prototype

2. Which property allows you to access an object's prototype in JavaScript?

5 баллов

- ☐ prototypeOf
- ☐ __proto__
- ☐ inherit
- ☐ prototypeLink

3. Which of the following statements is true about the `__proto__` property?

5 баллов

- ☐ It is a standard property according to ECMAScript specification.
- ☐ It is deprecated and should not be used.
- ☐ It is a built-in property that cannot be changed.
- ☐ It is only used in Node.js.

4. Is it possible for an object to have a primitive type value as a `[[Prototype]]`?

5 баллов

- ☐ Yes
- ☐ No

5. What is the main difference between `'in'` operator and `'.hasOwnProperty()'` method?

***** 5 баллов

Мой ответ

6. What is the purpose of the `isPrototypeOf()` method in JavaScript?

5 баллов

- ☐ To check if a property exists in an object.
- ☐ To check if an object is in the prototype chain of another object.
- ☐ To set the prototype of an object.
- ☐ To create a new object with a specific prototype.

7. How is the `isPrototypeOf()` method used to check if one object is in the prototype chain of another?

5 баллов

- ☐ `object.isPrototypeOf(otherObject)`
- ☐ `otherObject.prototype.isPrototypeOf(object)`
- ☐ `Object.prototype.isPrototypeOf(object, otherObject)`
- ☐ `Object.isPrototypeOf(object, otherObject)`

8. Which of the following is an alternative way to check the prototype chain of an object, similar to `isPrototypeOf()`?

5 баллов

- ☐ `Object.assign()`
- ☐ `Object.setPrototypeOf()`
- ☐ `Object.getPrototypeOf()`
- ☐ `Object.defineProperty()`

9. Which of the following is an alternative way to set the prototype of an object, similar to `__proto__`?

5 баллов

- ☐ `Object.setPrototypeOf()`
- ☐ `Object.getPrototypeOf()`
- ☐ `Object.assign()`
- ☐ `Object.defineProperty()`

10. What will be the output of the following code snippet?

5 баллов

```
const obj1 = {
  name: 'Max',
};

const obj2 = Object.create(obj1);
obj2.greet = function() {
  console.log(this.name);
}
obj2.greet();
```

- ☐ Max
- ☐ undefined
- ☐ error
- ☐ `obj2.name`

11. What will be the output of the following code snippet?

5 баллов

```
const obj1 = {
  x: 10
};
const obj2 = Object.create(obj1);
console.log(obj1.isPrototypeOf(obj2));
```

- ☐ true
- ☐ false

12. What will be the output of the following code snippet?

5 баллов

```
const obj1 = {
  greet: function() {
    console.log("Hello from obj1");
  }
};
const obj2 = {
  __proto__: obj1
};
const obj3 = {
  __proto__: obj2
};
obj3.greet();
```

- ☐ Hello from obj1
- ☐ Hello from obj2
- ☐ Hello from obj3

13. What will be the output of the following code snippet?

5 баллов

```
const obj = Object.create(null);
console.log(obj.toString());
```

- ☐ undefined
- ☐ null
- ☐ `[Object object]`
- ☐ `[object Object]`

14. What will be the output of the following code snippet?

5 баллов

```
const obj1 = {
  x: 10,
};
const obj2 = {
  a: 5,
};
Object.setPrototypeOf(obj2, obj1);
console.log(obj2.hasOwnProperty('x'));
```

- ☐ undefined
- ☐ true
- ☐ false
- ☐ `x:10`

15. What will be the output of the following code snippet?

5 баллов

```
const obj1 = {
  x: 10,
};
const obj2 = {
  a: 5,
};
Object.setPrototypeOf(obj2, obj1);
console.log('x' in obj2);
```

- ☐ undefined
- ☐ true
- ☐ false
- ☐ `x:10`

16. What will be the output of the following code snippet?

5 баллов

```
const obj1 = {
  x: 10,
};
const obj2 = {
  a: 5,
};
Object.setPrototypeOf(obj2, obj1);
Object.defineProperty(obj1, 'x', {writable: false});
obj2.x = 7;
console.log(obj2.x);
```

- ☐ 7
- ☐ 10
- ☐ 5
- ☐ error in use strict mode

17. What will be the output of the following code snippet?

5 баллов

```
const obj1 = {
  x: 10,
};
const obj2 = {
  a: 5,
};
Object.setPrototypeOf(obj2, obj1);
Object.defineProperty(obj1, 'x', {writable: false});
Object.defineProperty(obj2, 'x', {value: 67});
console.log(obj2.x);
```

- ☐ 67
- ☐ 10
- ☐ 5
- ☐ error in use strict mode

18. What will be the output of the following code snippet?

5 баллов

```
const obj1 = {
  x: 10
};
const obj2 = {
  x: 42
};
Object.setPrototypeOf(obj1, obj2);
console.log(Object.getPrototypeOf(obj1).x);
```

- ☐ obj1
- ☐ 42
- ☐ obj2
- ☐ 10

19. What will be the output of the following code snippet?

5 баллов

```
const obj1 = {
  x: 10
};
const obj2 = {
  y: 20
};
obj2.__proto__ = obj1;
delete obj2.x;
console.log(obj1.x);
```

- ☐ undefined
- ☐ 20
- ☐ 10

20. What will be the output of the following code snippet?

5 баллов

```
const obj1 = {
  a: 2,
}
const obj2 = Object.create(obj1)

obj2.a++;
console.log(obj1.a);
console.log(obj2.a);
```

- ☐ 2,2
- ☐ 3,2
- ☐ 2,3
- ☐ 3,3

[Отправить](#)

[Очистить форму](#)

Никогда не используйте формы Google для передачи паролей.

Компания Google не имеет никакого отношения к этому контенту. [Сообщение о нарушении](#) - [Условия использования](#) - [Политика конфиденциальности](#)

Google Формы