

The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern and dynamic visual effect.

# Изучение ВОЗМОЖНОСТЕЙ языка JavaScript

# Функции

Функция в программировании — фрагмент программного кода (подпрограмма), к которому можно обратиться из другого места программы.

Вначале идет ключевое слово **function**, после него **имя функции**, затем **список параметров в скобках** (параметров может не быть => оставляем пустые скобки) и **тело функции в фигурных скобках** — код, который выполняется при её вызове.

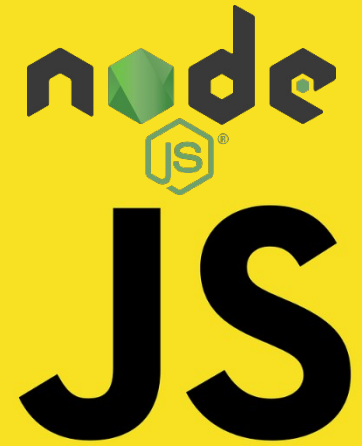
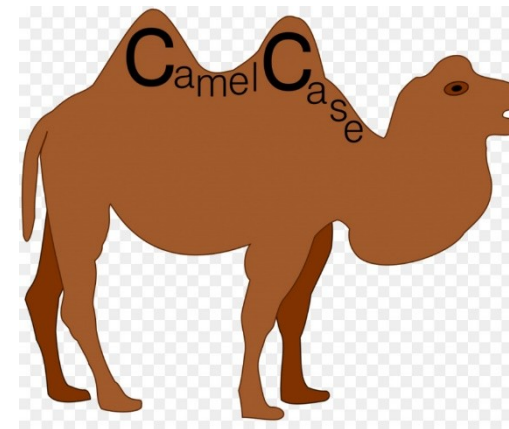
```
function nameOfFunction(param1, param2) {  
    let answer = param1 + param2;  
    return answer;  
}  
  
nameOfFunction(a, b);
```

return — возвращает ответ

Функция может не возвращать ответ, например функция распечатки

## Как оформляется код

CamelCase (с англ. — «ВерблюжийРегистр», также «ГорбатыйРегистр», «СтильВерблюда») — стиль написания составных слов, при котором несколько слов пишутся слитно без пробелов, при этом каждое слово внутри фразы пишется с прописной буквы.



# Объекты

Объект — это **ассоциативный массив**, а именно, **набор свойств**, и каждое свойство состоит из имени и значения, ассоциированного с этим именем.

**Ассоциативный массив** – структура данных, в которой можно хранить любые данные в формате ключ-значение.

```
let obj = { property_1: value_1, // property(свойство) # обозначается именем и именем...
            2: value_2, // или номером...
            // ...,
            "property n": value_n }; // или строкой...
```

здесь obj – это имя нового объекта, каждое property\_i – это идентификатор

Значением свойства может быть **функция**, которую можно назвать **методом** объекта.

Методы определяются так же, как и обычные функции, за тем исключением, что **они присваиваются свойству объекта**

```
let myObj = {
  myMethod: function(params) {
    // ...do something
  }
};
```

Пробегаемся по всем полям объекта

```
1 | var obj = {a:1, b:2, c:3};
2 |
3 | for (var prop in obj) {
4 |   console.log("obj." + prop + " = " + obj[prop]);
5 | }
```

# Объекты

- Доступ к элементам осуществляется:
  - Напрямую по ключу `obj.prop = 5`
  - Через переменную, в которой хранится ключ:

```
1 var key = "prop";  
2 obj[key] = 5
```

- Удаление ключей: `delete obj.name`.
- Существование свойства может проверять оператор `in`: `if ("prop" in obj)`, как правило, работает и просто сравнение `if (obj.prop !== undefined)`.

# Спасибо за внимание :)

Примеры с урока, установки и ДЗ по ссылке:

<https://github.com/ing-prog/ing-prog-web-2>