

Prototype In Javascript

Prototype is one of the most confusing concepts

Javascript is a prototype based language, so whenever we are creating a function using JS,

JS Engine adds a prototype property inside a function.

There are 2 prototypes

1. prototypes This is a special object which is assigned as property of any function you make in JS.

It is already present on any function you make, but not mandatory for internal function provided by Javascript and function returned by bind

Date: / /

2. [[Prototype]] → This is **hidden property** on every object which is accessed by running context.

This property simply is a reference to prototype of function from which object was made

It can be accessed in script using special getter - setter, called `--proto--`.

Function's prototype property is accessible and modifiable

Object's prototype property is not visible.

You can use `Object.getPrototypeOf(obj)` method instead of `--proto--` to access prototype object

Prototype is useful in keeping a copy of functions for all objects.

Eg \Rightarrow var obj = {}

When I make new object, it is empty.

However It is not empty because it is instance of an object constructor, and It inherently gets a reference of prototype of object, which is pointed to —proto— of newly created object.

Eg \Rightarrow var obj1 = new Object()

It also creates same — proto — as earlier obj.

In fact they are two pointers referring to same object

obj. — proto — === Object.prototype

Every prototype of function has an inherent property called constructor which is a pointer to function itself.

In case of Object function, the prototype has constructor which points back to object

Object.prototype.constructor === Object

— proto — of object looks same as prototype of function

Date: / /

If you check

`Object == proto == Function.prototype`

It returns true.

Function itself has `--proto--` which means function constructor is called from constructor function which has a prototype.

Why do we Use Prototype

- * To find properties and methods of an object.
- * To implement Inheritance in Javascript

Date: / /

Difference between Prototype and --proto--

Prototype

property of
class constructor

visible, can be
~~used to~~ access
and modify.

--proto--

property of class
Instance.

Invisible, It returns
prototype object of
fn it links to

Instance of Constructor
function use - proto-