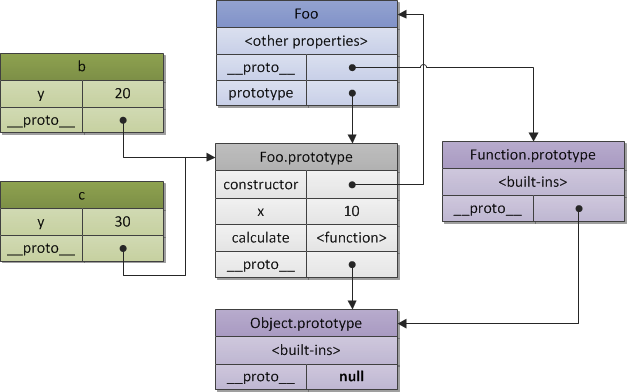
**http://dmitrysoshnikov.com/ecmascript/javascript-the-core/**

**General Idea :**

***In javascript a function can be used as a constructor.That means we can create objects out of them using new keyword.Every constructor function comes with a built in object chained with them.This built in object is called prototype .Instances of a constructor function use \_\_ proto \_\_ to access the prototype of it's constructor function.***

***Now i will explain the image which op attached with the question:***

[](https://i.stack.imgur.com/AGfN3.png)

**1.** First we created a constructor function **Foo(){}**

**2.** constructor function **Foo** has a prototype property which points to it's prototype which is**Foo.prototype**( see image).

**3.** Constructor functions are itself a function which is instance of a system constructor called **[[Function]]** constructor.So we can say function **Foo is constructed by [[Function]] constructor**.So, **\_\_ proto \_\_** of our **Foo** function will point to the prototype of it's consturctor which is **Function.prototype**.

**4.** **Function.prototype** is itself is nothing but an object which is constructed from another system constructor called **[[Object]]**.So, **[[Object]]** is the constructor of **Function.prototype**. So, we can say **Function.prototype** is an instance of **[[Object]]**. So **\_\_ proto \_\_** of **Function.prototype** points to **Object.prototype**.

**5.** **Object.prototype** is last man standing in prototype chain. I mean it has not been consturcted. It's already there in the system. So it's **\_\_ proto \_\_** points to **null**.

**6.** Now come to **instances of Foo**.When we create an instance using **new Foo()**, it creates a new object which is an instance **Foo**. That means **Foo** it the constructor of these instances.Here we created two instances (x and y). **\_\_ proto \_\_** of x and y thus points to **Foo.prototype**.