1. Every object in Javascript has a constructor property that returns the function that was used to construct or create that object
2. If you have a function, where you get constructor dynamically and you want to create an instance of an object based on that constructor then you have to reset the constructor property
3. Encapsulation: we encapsulate our variables and functions into objects
4. Polymorphism: when we encapsulate our variables and functions into objects and use Inheritance, we can execute many forms of a method by using a single line of code
5. Abstracion:
   * Hiding the details and complexity, and showing only the essential parts
   * Hiding some of these properties and methods from the outside
   * Hiding the details and comlexity inside of the module
6. Whe we use the new operator, three things happen:

* This new operator create a new object
* Sets this keyword to point to that object
* That new object will be return

**Note**: if we don’t use new keyword to instantiate a object => this will point to global object

1. In Classical object-oriented languages, we have two types of methods:
   * Instance methods
   * Static methods:
     1. are available on the class itself. Not the object instance. We often use them to create utility functions that are not specific to a given object
     2. To call a static method, we don’t have to create an instance class
2. Strict mode: by default body of classes are executed in the strict mode