# Question 1: Clock

## Clock Constructor Function

|  |  |
| --- | --- |
| 1. Also add a line of code that will stop the clock after 10 ticks | setTimeout(()=>clock.stop(), 10000); |
| 2. What is the inner function of the constructor function? | render |
| 3. What is the local variable of the constructor function? | timer |
| 4. What is the clock “interface” returned by the constructor function? | An object of Clock |
| 5.What are the closures? | No |
| 6.What are the private variables and functions? | timer, render |
| 7. What are the public methods? | stop,start |
| 8. How does this example illustrate that a Javascript class is really a function and not an object? |  |

## Clock Class

|  |  |
| --- | --- |
| 1. Also add a line of code that will stop the clock after 10 ticks | setTimeout(()=>clock.stop(), 10000); |
| 2. What is the inner function of the constructor function? | No |
| 3. What is the local variable of the constructor function? | No |
| 4. What is the clock “interface” returned by the constructor function? | An object of Clock |
| 5.What are the closures? | No |
| 6.What are the private variables and functions? | No |
| 7. What are the public methods? | stop, start, render |
| 8. How does this example illustrate that a Javascript class is really a function and not an object? | As you must already be aware by now there are no classes in JavaScript. Instead functions in JavaScript may be made to behave like constructors by preceding a function call with the new keyword.This is known as the constructor pattern.  In JavaScript everything is an object except for the primitive data types (boolean, number and string), and undefined.  On the other hand null is actually an object reference even though you may at first believe otherwise. This is the reason typeof null returns "object".  Functions in JavaScript are like functables in Lua (i.e., they are callable objects).  Hence a function can be used in place of an object. Similarly, arrays are also objects in JavaScript. On the other hand, objects can be thought of as associative arrays.  The most important point however is that there are no classes in JavaScript because JavaScript is a prototypal object-oriented language.  This means that objects in JavaScript directly inherit from other objects. Hence, we don't need classes. All we need is a way to create and extend objects. |

# Question 3: quiz system