**util : -The term ‘util’ is a short form of Utility, it is commonly used in programming to refer to module that provides utility functions or methods. These functions typically perform common tasks that are not specific to a particular domain or application, but are useful across different parts of program.Important Methods/ Properties Of Util Module Are As Follows – 1. debuglog() – it is used for writing debug messages to the error object . 2.deprecate () – the method is used to mark functions as deprecated, it means that the functions are no longer recommended for use and might be removed in future versions. 3. format() – This method allows to create formatted strings by replacing placeholders in a template string with values. 4. inherits() - the method is used for inheritance by linking the prototype of a constructor function to another constructor function prototype. 5. inspect() – this method is used to convert an object into a string representation of debugging purpose. It provides a way to inspect the properties and values of an object, making it easier to understand its structure.**

**Timers : -A timer is a module that enables us to execute a function at a particular time. it helps to delay the execution of a code or to execute something at regular intervals.**

**Timers are used to create animations using JavaScript.**

**Timers are also used to delay a function execution and helps create functionality to send emails after specified intervals. Timers are also used to refresh data and specify schedules from time to time.**

**Timer module methods setInterval() -**

**While working on a website , setInterval()() can be used to trigger an alert in regular intervals. \* The setInterval()() method helps us to repeatedly execute a function after a fixed delay. Takes a variable and time interval as arguments.**

**clearInterval()- The clearInterval() method clears a timer set with the setInterval() method. SetTimeout()- setTimeout method helps in a delay to execute a function. setTimeout will be executed only once.**

**clearTimeout()- The clearTimeout() method clears a timer set with the setTimeout() method.it prevents the function from execution.**

**setImmediate() and clearImmediate() :-The setImmediate() function in JavaScript is used to execute a function at the end of the current loop cycle. It is similar to the setTimeout() function, but it has a few key differences.**

**setImmediate()** **always executes after all I/O callbacks have been processed. This means that it can be used to ensure that your code is executed in the correct order, even if it depends on asynchronous operations.**