LETSUPGARDE- JAVASCRIPT ESSTENTIALS

MINI PROJECT

NAME : K KAVYA

EMAIL : [kavya.kguru@gmail.com](mailto:kavya.kguru@gmail.com)

QUESTION 2:

Write a program to make a Stopwatch. Which takes the timer to be set from the user

HTMLCODE:

<!doctype html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <title>Stopwatch</title>

</head>

<body onload="show();">

  <div>Time: <span id="time"></span></div>

  <input type="button" value="start" onclick="start();">

  <input type="button" value="stop" onclick="stop();">

  <input type="button" value="reset" onclick="reset()">

</body>

</html>

JSCODE:

|  |
| --- |
|  |

*//  Simple example of using private variables*

*//*

*//  To start the stopwatch:*

*//    obj.start();*

*//*

*//  To get the duration in milliseconds without pausing / resuming:*

*//    var x = obj.time();*

*//*

*//  To pause the stopwatch:*

*//    var x = obj.stop(); // Result is duration in milliseconds*

*//*

*//  To resume a paused stopwatch*

*//    var x = obj.start();  // Result is duration in milliseconds*

*//*

*//  To reset a paused stopwatch*

*//    obj.stop();*

*//*

var clsStopwatch = function() {

*// Private vars*

    var startAt = 0;  *// Time of last start / resume. (0 if not running)*

    var lapTime = 0;  *// Time on the clock when last stopped in milliseconds*

    var now = function() {

        return (new Date()).getTime();

      };

*// Public methods*

*// Start or resume*

    this.start = function() {

        startAt = startAt ? startAt : now();

      };

*// Stop or pause*

    this.stop = function() {

*// If running, update elapsed time otherwise keep it*

        lapTime = startAt ? lapTime + now() - startAt : lapTime;

        startAt = 0; *// Paused*

      };

*// Reset*

    this.reset = function() {

        lapTime = startAt = 0;

      };

*// Duration*

    this.time = function() {

        return lapTime + (startAt ? now() - startAt : 0);

      };

  };

var x = new clsStopwatch();

var $time;

var clocktimer;

function pad(*num*, *size*) {

  var s = "0000" + num;

  return s.substr(s.length - size);

}

function formatTime(*time*) {

  var h = m = s = ms = 0;

  var newTime = '';

  h = Math.floor( time / (60 \* 60 \* 1000) );

  time = time % (60 \* 60 \* 1000);

  m = Math.floor( time / (60 \* 1000) );

  time = time % (60 \* 1000);

  s = Math.floor( time / 1000 );

  ms = time % 1000;

  newTime = pad(h, 2) + ':' + pad(m, 2) + ':' + pad(s, 2) + ':' + pad(ms, 3);

  return newTime;

}

function show() {

  $time = document.getElementById('time');

  update();

}

function update() {

  $time.innerHTML = formatTime(x.time());

}

function start() {

  clocktimer = setInterval("update()", 1);

  x.start();

}

function stop() {

  x.stop();

  clearInterval(clocktimer);

}

function reset() {

  stop();

  x.reset();

  update();

}

OUPUT:



