Finish PHP Intro
Client-Side Programming

cs380

G Towell
Sep 28

Printf

$_SESSION

- persist information about a “session”
  - Session is usually
    - browser specific
      - but shared across tabs in browser
    - machine specific
    - of fairly long duration (but you can define it to be any length)
  - Implemented in 2 parts:
    - a cookie (which PHP automatically puts) on client
      - chrome, developer tools/application/cookies
        - PHPSESSID
    - a file on server

```
cat sessions/sess_9tahcsfq1uh0gr3rojg5filfq2

count|i:12;name|s:5:"Geoff";longname|s:12:"Geoff
Towell";double|d:3.1415926;
```

Session save is automatic

Override default location

file:sess.php

```
<?php
ini_set('session.save_path', '/etc/php/sessions');
session_start();
?>
<!DOCTYPE html><html><body>
<?php
$count = 1;
if (isset($_SESSION) &&
array_key_exists("count", $_SESSION)) {
   $count = $_SESSION["count"] + 1;
   }
else {
    echo("No session<br>");
}
$_SESSION[“count"]=$count;
$_SESSION["name"]="Geoff";
$_SESSION["longname"]="Geoff Towell";
$_SESSION["double"]=3.1415926;
print "<table border=\"1\">" ;
foreach ($_SESSION as $name => $value) {
    echo "<tr><td>" . $name . "</td><td>" .
$value . "</td></tr>" ;
}
echo "</table>";
?>
</body></html>
```

This MUST be close to first

Forms and Sessions

- html same except for action
  - now “sessf.php”
- PHP
  - copy all POST into session
  - Note dynamic creation of keys for session map
  - Print everything in session
- Stuff accumulates
  - This can get expensive

Files: form2.html, sessf.php

```
<?php
session_start();
?>
<!DOCTYPE html><html><body>
<?php
// Set session variables
$count = 1;
if (isset($_SESSION) &&
array_key_exists("count", $_SESSION)) {
   $count = $_SESSION["count"] + 1;
   }
$_SESSION["count"] = $count;
foreach ($_POST as $name => $value) {
    $_SESSION[$name . "_" . $count] = $value;
}
echo "<table border=\"1\">" ;
foreach ($_SESSION as $name => $value) {
    echo "<tr><td>" . $name . "</td><td>" .
$value . "</td></tr>" ;
}
echo "</table>";
?>
<a href="form2.html">Back to form</a>
</body></html>
```

PHP

Event-driven and preparation/presentation

- Is PHP event-driven?

- Separating preparation from presentation?
  - use php “include” command
    - put all of your functions in a separate file (myfunctions.php)
      - or at least at top of the page
      - <?php include(“myfunctions.php”) ?>
    - try to separate html and php
      - often an unachievable goal

Separation … functions to the top

file:separate.php

```
<?php
session_start();
?>
<!DOCTYPE html><html><body>
<?php
// Set session variables
$count = 1;
if (isset($_SESSION) &&
array_key_exists("count", $_SESSION)) {
   $count = $_SESSION["count"] + 1;
   }
$_SESSION["count"] = $count;
foreach ($_POST as $name => $value) {
    $_SESSION[$name . "_" . $count] = $value;
}
echo "<table border=\"1\">" ;
foreach ($_SESSION as $name => $value) {
    echo "<tr><td>" . $name . "</td><td>" .
$value . "</td></tr>" ;
}
echo "</table>";
?>
<a href="formers.html">Back to form</
</body></html>
```

```
<?php
function startAndUpdatedateSession($parray) {
    session_start();
    $count = 1;
    if (isset($_SESSION) &&
    array_key_exists("count", $_SESSION)) {
       $count = $_SESSION["count"] + 1;
       }
    $_SESSION["count"] = $count;
    foreach ($parray as $name => $value) {
        $_SESSION[$name . "_" . $count] = $value
    }
}
?>
<!DOCTYPE html><html><body>
<?php startAndUpdateSession($_POST); ?>
<table border=1>
<?php
foreach ($_SESSION as $name => $value) {
    echo "<tr><td>" . $name . "</td><td>" .
$value . "</td></tr>" ;
}
?>
</table>
<a href="form2.html">Back to form</a>
</body></html>
```

Maybe a function here also

Using Include

Just because include is within php block, does not imply that include is php. So you need <?php ... ?>

You could use included to put in html, Or javascript libraries

File: includer.php

```
<?php
include 'included.php';
?>
<!DOCTYPE html><html><body>
<?php updateSession($_POST); ?>
<table border=1>
<?php
foreach ($_SESSION as $name => $value) {
    echo "<tr><td>" . $name . "</td><td>" .
$value . "</td></tr>" ;
}
?>
</table>
<a href="formers.html">Back to form</a>
</body></html>
```

File: included.php

```
<?php
function updateSession($parray) {
    session_start();
    $count = 1;
    if (isset($_SESSION) &&
    array_key_exists("count", $_SESSION)) {
       $count = $_SESSION["count"] + 1;
       }
    $_SESSION["count"] = $count;
    foreach ($parray as $name => $value) {
        $_SESSION[$name . "_" . $count] = $value;
    }
}
?>
```

SQL and PHP

- mysqli gives and interface that is almost identical to mysql UI.
  - Especially for queries
- Steps:
  - login and connect to database
  - do a query
  - check if there are results

```
$conn = new mysqli("localhost", "db_user",
“", "shakespeare");

$result = $conn->query($q);

if ($result->num_rows > 0) {
…
}
```

The shakespeare database

```
use shakespeare;
Database changed
MariaDB [shakespeare]> show tables;
+-----------------------+
| Tables_in_shakespeare |
+-----------------------+
| plays                 |
| slines                |
+-----------------------+
2 rows in set (0.000 sec)

MariaDB [shakespeare]> describe plays;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| playid | char(5)     | NO   | PRI | NULL    |       |
| pname  | varchar(64) | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
2 rows in set (0.001 sec)

MariaDB [shakespeare]> describe slines;
+------------+---------------+------+-----+---------+-------+
| Field      | Type          | Null | Key | Default | Extra |
+------------+---------------+------+-----+---------+-------+
| pid        | char(5)       | NO   | PRI | NULL    |       |
| speaker    | varchar(64)   | NO   |     | NULL    |       |
| lineinplay | int(11)       | NO   | PRI | NULL    |       |
| theline    | varchar(2048) | NO   |     | NULL    |       |
+------------+---------------+------+-----+---------+-------+
4 rows in set (0.001 sec)
```

One random line

```
<html><body>
    <?php
$conn = new mysqli("localhost", "gtstudent", "", "shakespeare");
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
echo "<h2>Query</h2>";
$q = "with pp as (select playid,pname from plays as pl order by rand() limit 1) select 
pp.pname, speaker, lineinplay, theline from slines as sl inner join pp on pp.playid=sl.pid 
order by rand() limit 1;";
echo $q;
echo "<br><h2>Result</h2>";
$result = $conn->query($q);
if ($result->num_rows > 0) {
   while ($row = $result->fetch_assoc()) {
   foreach(  $row as $kk=>$vv ) {
       echo $kk . ”  " . $vv;
       }
   echo "\n";
   }
   }
else {
    echo "Got nothing";
}
?>
</body></html>
```

UGLY!
Needs your SQL password in clear text
All SQL accounts have no password

file: oneline.php

Client-Side Programming

- Make content fit:
  - The server gives the client a bunch of content. Once it is received, programs on the client side figure out how to make it look good
    - JQuery / JQuery Mobile
      - class home page on devices vs my personal page
- Create new content:
  - Classic Example: a stopwatch

- Both reformatting and new content use programs that run on the client-side
  - Most commonly, these programs are written using Javascript
    - NOT Java, just a bad marketing decision.

A Stopwatch

```
<head>
<style>
.button {
  display: block; width: 100%; height: 100pt;
  border: none; padding: 14px 28px; font-size: 32px;
  cursor: pointer; text-align: center;
}
.buttonstart { background-color: green;
    color:black; }
.buttonstart:hover { background-color: lightgreen; }

.buttonstop { background-color: red;
    color:white; }
.buttonstop:hover {  background-color: #660000; }
</style>
</head>

<body onload=initialize()>
<center>
<span style="font-size:96px" id="swtime"></span>
</center>
<button class="button" id="start" onclick="doStartStop();"></button> <br/>
<button id="clear" onclick="resetTime();">reset</button>
```

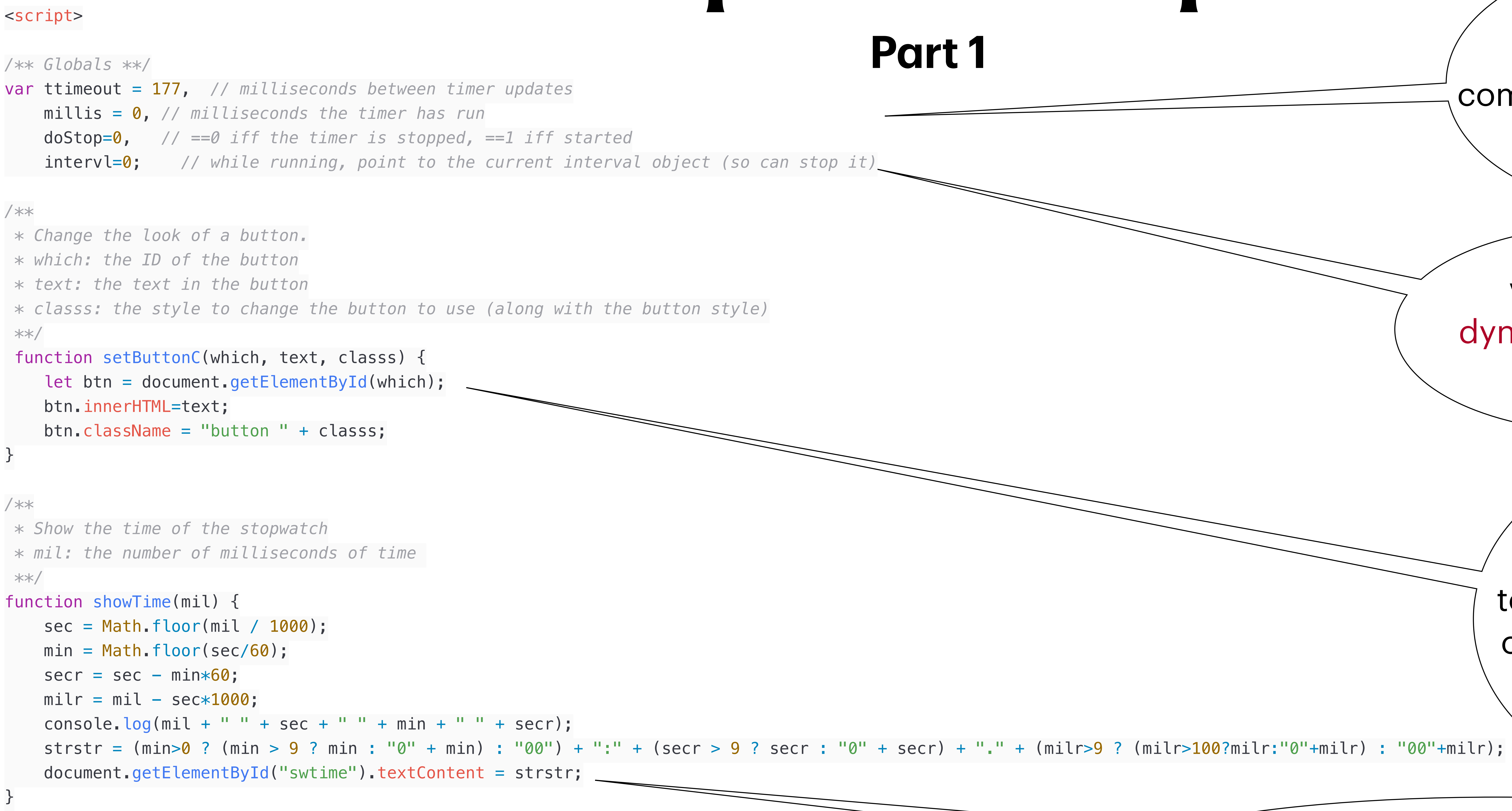
CSS
Why “cascading”, examples…

Generally not a good idea to use. Load does not mean ready

The entire stopwatch layout.
Everything else is Javascript

Javascript for the stopwatch

Part 1



```
<script>

/** Globals **/
var ttimeout = 177,  // milliseconds between timer updates
    millis = 0, // milliseconds the timer has run
    doStop=0,   // ==0 iff the timer is stopped, ==1 iff started
    intervl=0;    // while running, point to the current interval object (so can stop it)

/**
 * Change the look of a button.
 * which: the ID of the button
 * text: the text in the button
 * classs: the style to change the button to use (along with the button style)
 **/
 function setButtonC(which, text, classs) {
    let btn = document.getElementById(which);
    btn.innerHTML=text;
    btn.className = "button " + classs;
}

/**
 * Show the time of the stopwatch
 * mil: the number of milliseconds of time
 **/
function showTime(mil) {
    sec = Math.floor(mil / 1000);
    min = Math.floor(sec/60);
    secr = sec - min*60;
    milr = mil - sec*1000;
    console.log(mil + " " + sec + " " + min + " " + secr);
    strstr = (min>0 ? (min > 9 ? min : "0" + min) : "00") + ":" + (secr > 9 ? secr : "0" + secr) + "." + (milr>9 ? (milr>100?milr:"0"+milr) : "00"+milr);
    document.getElementById("swtime").textContent = strstr;
}
```

global variables, commented appropriately

variables are dynamically typed/ Typescript

javascript function to change color and text of button. Color change done by changing css

Time is shown by changing the text of element of page

Part 2

```
/**
 * Add time to the stopwatch, and show the time after adding
 * Also set timeout so that the clock keeps incrementing
 **/
function add() {
    millis=millis+ttimeout;
    console.log("ADD " + millis);
    showTime(millis);
}

/**
 * Start or stop the timer
 **/
 function doStartStop() {
    console.log("hello");
    if (doStop==1) {
        doStop=0;
        clearInterval(intervl);
        setButtonC("start", "start", "buttonstart");
        //setButton("start", "start", startText, startcolor, starthover);
    } else {
        intervl = setInterval(add, ttimeout);
        setButtonC("start", "stop", "buttonstop");
        //setButton("start", "stop", stopText, stopcolor, stophover);
        doStop=1;
    }
}
```

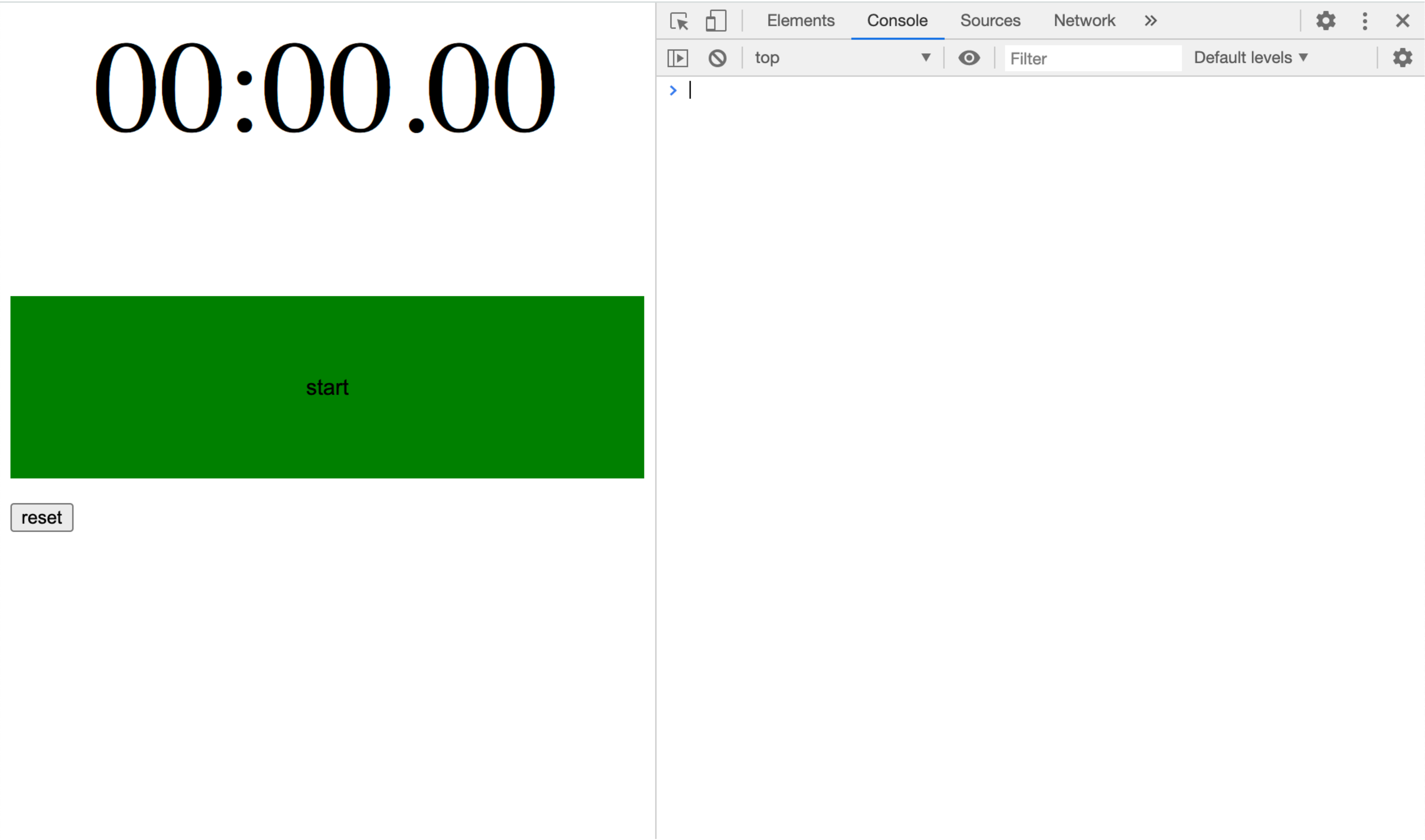
Part 3

```
/**
 * Reset the timer.
 **/
function resetTime() {
    millis=0;
    showTime(millis);
}

/**
 * Initialize the start/stop button, etc. Called onload.
 **/
function initialize() {
    setButtonC("start", "START", "buttonstart");
    resetTime();
}
```

Debugging/ running

- Only possible within the browser
- I like the tools that come with chrome. Firefox works also.
  - TONs of developer add ons
- View/Developer/Javascript console
  - console.log(“xxxx”);
  - view sources
    - breakpoints!



Javascript is event driven

- Events in the stopwatch example
  - onload
  - onclick
  - timeout
  - onmouseover
  - onmouseout
- Other events:
  - onkeydown
  - etc (see https://www.w3schools.com/jsref/dom_obj_event.asp)

An RPN Calculator

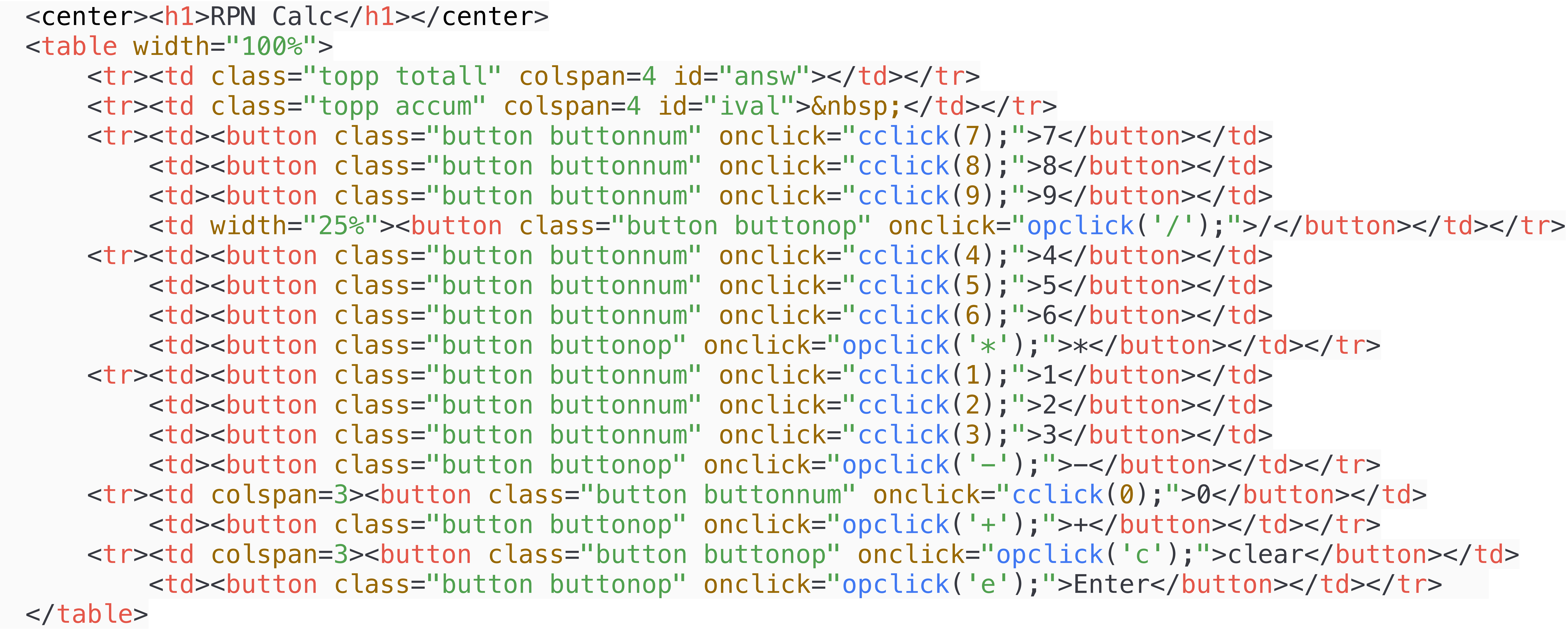
CSS

```
<style>
.topp {
    height: 50pt;
    border: none;
    font-size: 24pt;
    text-align: right;
}
.accum {
    background-color:wheat;
    color:black;
}
.totall {
    background-color: magenta;
    color:black
}

.button {
  display: block; width: 100%;
  height: 50pt; border: none;
  font-size: 24pt; cursor: pointer; text-align: center;
}
.buttonnum { background-color: green;
    color:black;
}
.buttonnum:hover {    background-color: lightgreen; }
.buttonop { background-color: red;
    color:white;
}
.buttonop:hover {    background-color: #660000; }
</style>
```

RPN Calculator

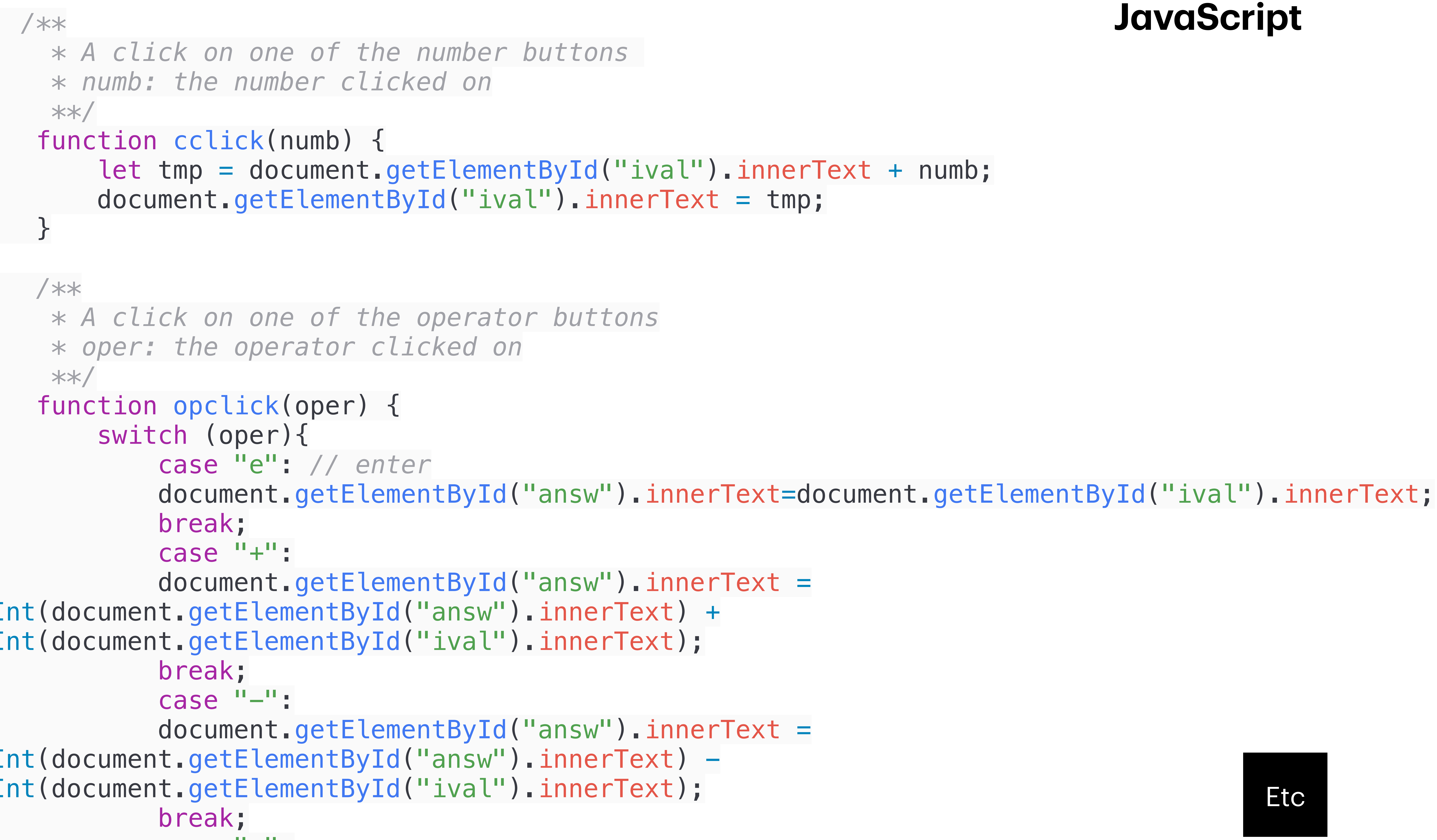
HTML



```
<center><h1>RPN Calc</h1></center>
<table width="100%">
    <tr><td class="topp totall" colspan=4 id="answ"></td></tr>
    <tr><td class="topp accum" colspan=4 id="ival">&nbsp;</td></tr>
    <tr><td><button class="button buttonnum" onclick="cclick(7);">7</button></td>
        <td><button class="button buttonnum" onclick="cclick(8);">8</button></td>
        <td><button class="button buttonnum" onclick="cclick(9);">9</button></td>
        <td width="25%"><button class="button buttonop" onclick="opclick('/');">/</button></td></tr>
    <tr><td><button class="button buttonnum" onclick="cclick(4);">4</button></td>
        <td><button class="button buttonnum" onclick="cclick(5);">5</button></td>
        <td><button class="button buttonnum" onclick="cclick(6);">6</button></td>
        <td><button class="button buttonop" onclick="opclick('*');">*</button></td></tr>
    <tr><td><button class="button buttonnum" onclick="cclick(1);">1</button></td>
        <td><button class="button buttonnum" onclick="cclick(2);">2</button></td>
        <td><button class="button buttonnum" onclick="cclick(3);">3</button></td>
        <td><button class="button buttonop" onclick="opclick('-');">-</button></td></tr>
    <tr><td colspan=3><button class="button buttonnum" onclick="cclick(0);">0</button></td>
        <td><button class="button buttonop" onclick="opclick('+');">+</button></td></tr>
    <tr><td colspan=3><button class="button buttonop" onclick="opclick('c');">clear</button></td>
        <td><button class="button buttonop" onclick="opclick('e');">Enter</button></td></tr>
</table>
```

RPN Calculator

JavaScript



```
/**
  * A click on one of the number buttons
  * numb: the number clicked on
  **/
 function cclick(numb) {
     let tmp = document.getElementById("ival").innerText + numb;
     document.getElementById("ival").innerText = tmp;
 }

 /**
  * A click on one of the operator buttons
  * oper: the operator clicked on
  **/
 function opclick(oper) {
     switch (oper){
         case "e": // enter
         document.getElementById("answ").innerText=document.getElementById("ival").innerText;
         break;
         case "+":
         document.getElementById("answ").innerText =
Int(document.getElementById("answ").innerText) +
Int(document.getElementById("ival").innerText);
         break;
         case "-":
         document.getElementById("answ").innerText =
Int(document.getElementById("answ").innerText) -
Int(document.getElementById("ival").innerText);
         break;
```

Etc

Next time

More javascript

Reading:

https://developer.mozilla.org/en-US/docs/Web/JavaScript/A_re-introduction_to_JavaScript