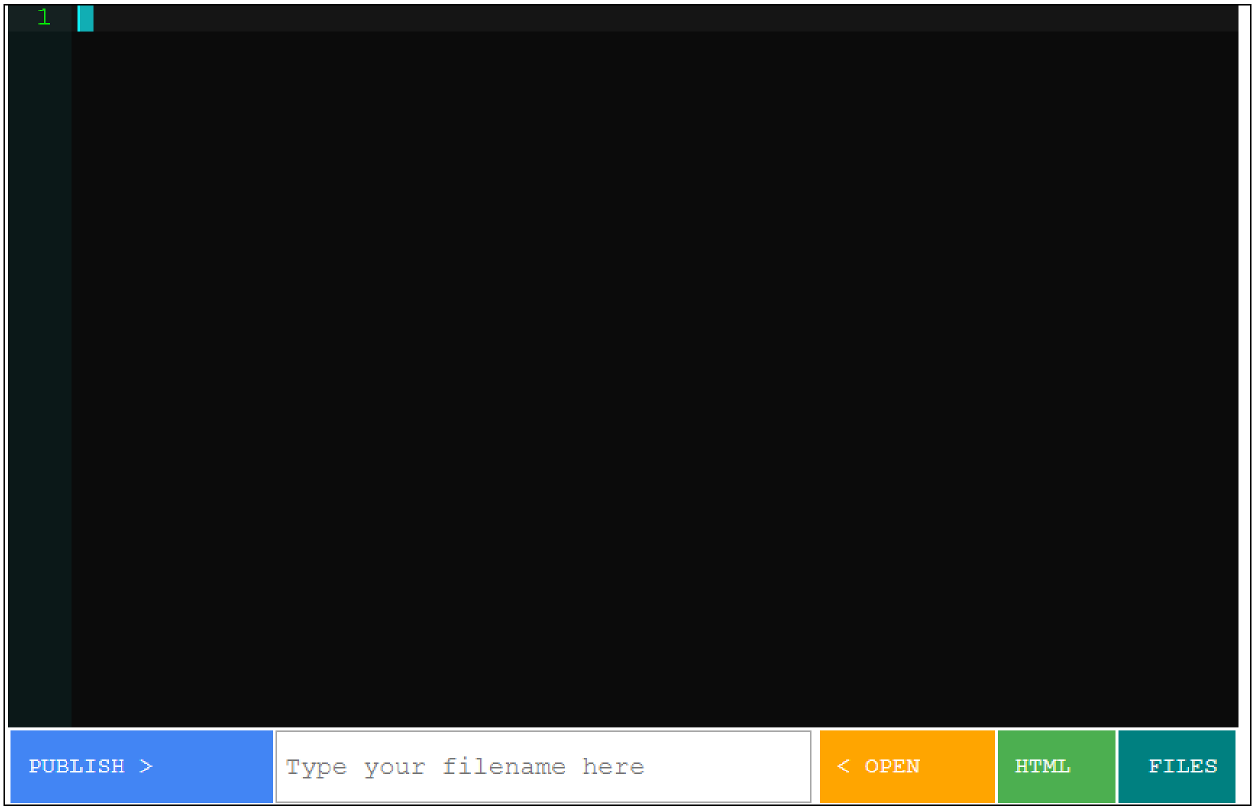
We’re going to learn a bit about programming games in HTML. This week, we’ll be reviewing HTML and creating our first HTML page, and including a clock in it. The clock is important to all kinds of games – when you play ‘whack a mole’ on the computer, the clock controls how long the game lasts. This week we’ll see how to include a clock on an HTML page, and then use it, to control a simple game.

To start, click the CoderDojo icon along the bottom of the screen. This should bring you to the DingleDojo website. Click the button, to navigate to https://quickdojo.com/. Here you should see the editor window we’ll be using for the Dojo:

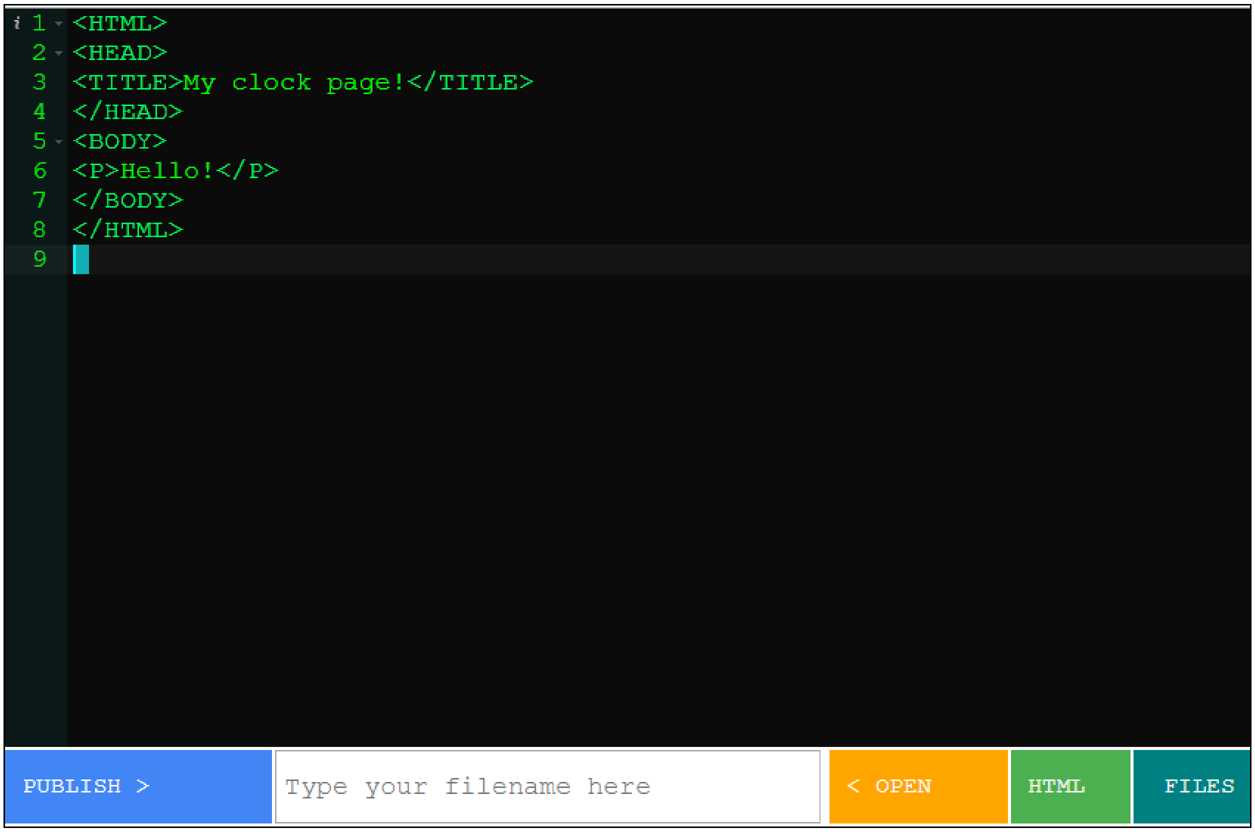


Now, let’s create a very simple HTML web page. HTML (HyperText Markup Language) is the language we use to tell the web browser what we want to do on our web page. Web pages can be sites to buy things, like Amazon.co.uk, to watch progames like RTE.ie, to read and write e-mail like gmail.com, and many more. All these web pages are written in HTML.

Type in the following into the editor.

```
<HTML>

<HEAD>

<TITLE>My clock page!</TITLE>

</HEAD>

<BODY>

<P>Hello! </P>

</BODY>

</HTML>
```

You should see the following in the editor window:



```
<HTML>
<HEAD>
<TITLE>My clock page!</TITLE>
</HEAD>
<BODY>
<P>Hello!</P>
</BODY>
</HTML>
```

Type in a filename like ‘SeamusFile1.html’ and press PUBLISH. You should see your HTML page on your web browser.

HTML files can contains all kinds of instructions in them. Each set of instructions to the browser, are controlled by tags, which are the words contained in angle-brackets '<' and '>'. All tags follow the same format – the starting tag, and the end of the tag. So, for example, <P>A paragraph!</P> is a paragraph of text in HTML.

Every HTML file starts with an <HTML> tag and ends with a closing </HTML> tag.

The <TITLE> tag tells the browser what title to display for your HTML page. <HEAD> contains the first section of your file, while <BODY> is the main part of the file, Tags can be on a single line, or span several lines as shown in the example.

For this Dojo, we'll be concerned mostly with two tags - <script> and <style>.

We will write our code under the <script> tag, in a language called Javascript. Javascript is widely used to program HTML pages to perform tasks, like clicking a clock or keeping a score in a game like 'whack-a-mole.' Style controls the colors, shapes and text properties in pages.

Let's start with a web page that shows a clock, that you control with a 'start' button:

```
<html>
    <body>
    <div id="txt"></div>
    <p>Click "Try it". Some number of seconds passes, and you'll see 
an alert.</p>

    <button onclick="myFunction();">Try it</button>

    <script>
    var elapsedTime = 0;
    var t;
    function myFunction(){
        t = setInterval(moveClockTime, 25);
        setTimeout(myAlert, Math.floor(3001*Math.random()));
    }
    function myAlert() {
        alert('Hello');
        clearInterval(t);
    }

    function moveClockTime() {
        elapsedTime +=25;
        document.getElementById('txt').innerHTML = 
        Math.floor(elapsedTime / 1000) + ":" + Math.floor(elapsedTime 
% 1000);

    }

    </script>
    </body>
    </html>
```

We've seen <html><body> etc. already. New in this file, is the <div> tag

```
<div id="txt"></div>
```

The <div> tag indicates a point of reference on the web page. It's a way to divide up web pages into different sections.

We also see

```
<button onclick="myFunction();">Try it</button>
```

This <button> tag says to the browser, that if this button is clicked, invoke the function 'myFunction', which is a Javascript function we wrote.

<script> is the Javascript tag.

Javascript is a programming language. You write programs in it to make the computer do something you want

- Javascript is widely used to program web pages

Javascript code is a sequence of statements

Statements come in two forms

- Function definitions
- Ordinary statements

Function definitions

- Functions are identified by the keyword `function`:
- `function` myFunction() is one we've written for our clock program

The myFunction function, has a lot of code in it, but it's not that scary! It contains some variable definitions, and some executable code.

Variable definitions tell the browser to set aside some small amounts of the computer's memory, to hold data, like numbers. In this web page, we see

```
var elapsedTime = 0;
```

The variable 'elapsedTime' will be used to hold the clock time from when the button labeled 'Try it!' is pressed. What's important to notice here, is that this instruction to the browser, called a 'statement' in Javascript, ends with a semi-colon ';'. Don't forget your semi-colons!

Variables can be modified through what's known as 'operations', which change the contents of a variable from one value to another. Mathematical operations are pretty common. Here, we see a bunch of ways to combine the variable 'x' and 'y' to put a value into the variable 'Z'.

- Z = x + y;
- Z = x - y
- Z = x * y;

- Z = x / y;
- Z = x % y; //remainder when x is divided by y

Sometimes, variables contain strings (though, not in the clock program we’re looking at.) You can tie two strings together as if they were simple numbers, using the ‘+’ sign, which we call ‘concatenation:’

So here, the variable “S” is set to the value of ‘my hovercraft is full of eels’

- S = “my hovercraft “ + “is full of eels!”

Another way we can assign a value, to a variable, is by invoking another function.

Javascript has lots of builtin functions we’ll be referencing throughout the Dojo. One group of them, is more advanced mathematical functions. In our example, we use the ‘floor’ function, which truncates the values after the decimal point, on the value returned by the function ‘random’ (which is a random number between 0 and 1. We multiple the value by 11, to get us a random number between 0 and 10:

```
Z = Math.floor(Math.random() * 11);
```

Most of the interesting ‘stuff’ in our program, is controlled by the builtin function setInterval, and the function we wrote called moveClockTime.

The function invocation

```
setInterval(moveClockTime,25)
```

Tells the web page to run the code in the function ‘moveClockTime’ every 25 milliseconds.

The function moveClockTime looks like this:

```
function moveClockTime() {
        elapsedTime +=25;
        document.getElementById('txt').innerHTML =
        Math.floor(elapsedTime / 1000) + ":" + Math.floor(elapsedTime
% 1000);

    }
```

The function ‘document.getElementById(‘txt’).innerHTML’ tells the browser to replace the contents of the web page contained in the element ‘txt’, with some value. In this case, the value is an elapsed time string in seconds, such as 1:5.

So, what the moveClockTime function is doing, is incremented the elapsed time since it was last run by the browser by 25 milliseconds, and turning that value into a string, putting it on the web page at the location of the <div> named 'txt'.

Other builtin functions used in this program, are setTimeout and alert. setTimeout clears a timer that was created by setInterval. alert shows a box at the top of the web page with a string you supply in it.

Challenges

Easy: Make the program run for 5 seconds every time you run it.

Hard: Add a second button, label it 'Stop! Stop!', and program it so that when pressed, the program stops running, and a different alert shows saying you made the program stop running.

Another Javascript example

Web pages typically contain lots of text, in different typefaces or fonts. *Slanted,* underlined, or **boldfaced** letters are all features of different fonts. In the next example, we can see how to change text in different parts of a web page, by using a 'Cascading Style Sheet' (known as a 'css') with the HTML page. CSS is usually contained in a separate file whose filename ends in '.css'. In this example, we've embedded the css within the HTML page itself, in a section starting with <style> and ending with </style>. Type in the following code, save it to a new file and see what it does.

```
<!DOCTYPE HTML>

<html lang="en">

<head>
    <title>Dojo Javascript</title>
    <meta name="author" content="">
    <meta http-equiv="content-type" content="text/html; charset=utf-8"
/>
    <meta name="description" content="Javascript and Swahili" />
    <meta name="keywords" content="JS, Swahaili, Dates, Times,
Different time Zones, Different Cultures" />
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <link href="https://quickdojo.com/lessonplans/HTMLWeek1Style.css"
rel="stylesheet" type="text/css">
</head>

<body>
    <h1 class="kells">Javascript Dates and Times</h1>
    <hr />
    <div class="main">
        <p>Welcome to Javascript <strong><em>'Dates and
Times'</em></strong>;
            <p>It is now <span class='time'>

<!--This script gets the time from your own computer, it obviously
assumes you have your system time to the correct location-->

            <script>
                var now = new Date(),
                  hours = now.getHours();
                    if (hours < 10) {
                 hours = "0" + hours;
                }
                  minutes = now.getMinutes();
                    if (minutes < 10) {
                 minutes = "0" + minutes;
                }
                document.write(hours + ":" + minutes )
            </script>
            </span> in Dingle. In Zanzibar, the clocks are 3 hours
ahead of us.</p>
            <p>
                It is interesting to note that in the Swahili culture
the day starts at sunrise (unlike in the Arab world where the day
starts at sunset, and in the Western world where the day starts at
midnight).
                <br /><br /> Sunrise in Zanzibar, Tanzania and other
other African countries being very near the Equator, happens every day
at approximately 6:00 a.m. And for that reason, 6:00 a.m. is '0:00 in
the morning' Swahili time.
                <br /><br />By "Swahili time" we mean the time as
'Spoken' in Swahili.
                <br /><br />So, the hands of a watch or clock meant to
read Swahili time would always point to a number opposite to the
number for the actual time as spoken in English. That is, the Swahili
time (anywhere in the world, not just East Africa)
                is delayed by 6 hours
                <ul>
                    <li>Therefore 7:00 a.m. is '1:00 a.m. in the
morning' (saa moja asubuhi) Swahili time;</li>
                    <li>midnight is '6:00 p.m. in the evening' (saa
sita usiku) Swahili time.</li>
                    <li>5:00 a.m. is '11:00 p.m. at night' (saa kumi
na moja alfajiri) Swahili time.</li>
                </ul>
                Note also that the Swahili time doesn't use 'noon' as
the reference as in a.m. (before noon) and p.m. (after noon).
                <br /> The time is spoken using 'alfajiri' which is
the early morning time during which the light has started to shine but
the sun has not yet risen ; 'asubuhi' which is the time between
sunrise and a little before noon; 'mchana' which
                is from around noon to around 3:00 p.m.; 'alasiri'
which is from around 3:00 p.m. to sunset; 'jioni' which is the entire
time period from around 3:00 p.m. up to a little before 7:00 p.m.; and
'usiku' which is the entire time period from
                around 7:00 p.m. to early morning.
            </p>
            So,
            <!--This script gets the time in another location, namely,
Zanzibar-->

            <script>
                // function to calculate local time in a different
city given the city's UTC offset
```

```
                function calcTime(city, offset) {
                    // create Date object for current location
                    now = new Date();

                    // convert to msec, add local time zone offset & 
get UTC time in msec
                    utc = now.getTime() + (now.getTimezoneOffset() * 
60000);

                    // create new Date object for different city using 
supplied offset
                    zt = new Date(utc + (3600000 * offset));
                    st = new Date(utc - (3600000 * offset));
                    // return time in Zanzibar

                    return "The local time in " + city + " is now 
<span class='time'>" + zt.getHours() + ":" + zt.getMinutes() + 
"</span> (by Western clocks) and in Swhali spoken time it is <span 
class='time'>" + st.getHours() + ":" + st.getMinutes() + "</span>";
                }
                // Print out result
                document.write(calcTime('Zanzibar', '+3'));
            </script>

            <p>See if you can change the times and greetings in the 
javascript code to the Swahili words for that period of the day. When 
completed, you can fill in your name in the copyright notice with the 
full date (Day and Month).</p>

            <div class="afrika">

                <script type="text/javascript">
                    var now = new Date();
                    if (now.getHours() >= 0 && now.getHours() < 2) {
                        document.write("<strong>Asubuhi nzuri sana!
</strong>");
                    } else
                    if (now.getHours() >= 2 && now.getHours() < 8) {
                        document.write("<strong>Habari za 
asubuhi</strong>");
                    } else
                    if (now.getHours() >= 8 && now.getHours() < 14) {
                        document.write("<strong>Alasiri 
nzuri</strong>");
                    } else
                    if (now.getHours() >= 14 && now.getHours() < 24) {
                        document.write("<strong>Habari za 
jioni</strong>");
                    }
                </script>
            </div>
    </div>
    <hr />
    <div class="main">
        <div class="cent">&copy; CoderDojo Ninga 'YourName'
            <script>
                document.write(new Date().getFullYear())
            </script>
        </div>
    </div>

</body>

</html>
```