WSUK – Test Project
17. Web Technologies
Training Pack Samples

1 - SPEED

Contents

Introduction **2**

Part A: Easy **2**

A1: Yellow Square (Easy) **2**

A2: Jumper Ball (Easy) **2**

A3: Loading Screen (Easy) **2**

A4: Smiley (Easy) **2**

A5: Grayscale image (Easy) **2**

Part B: Medium **3**

B1: Simple Image Filter (Medium) **3**

B2: Transition (Medium) **3**

B3 RGB Slider (Medium) **4**

B4 Image blur linear slider (Medium) **4**

B5 Sunrise and sunset (Medium) **4**

Part C: Difficult **5**

C1: Linear gradient (Difficult) **5**

C2: Digital Timer (Difficult) **5**

C3: Roman Converter (Difficult) **5**

C4: Analog Clock (Difficult) **5**

C5: Cube Rotation (Difficult) **6**

Introduction

In this module, you have to complete these 15 mini tasks. These mini tasks are divided in 3 parts as easy, medium and hard. As a time, guidance for the speed tasks, you should aim to solve the requirements as follow:

- Easy =< 5 min
- Medium <> 5 – 15 min
- Difficult <> 15 – 30 min

Some tests have specific restrictions (e.g., *no JS allowed*). Please adhere to these restrictions. Create a new folder for each test number as it is provided in the test project. *No frameworks can be used*.

Part A: Easy

A1: Yellow Square (Easy)

Make an animation that starts with a big yellow square, and this square turns in red circle, the animation needs to be cantered on the view and the animation needs to loop. A sample is provided as *sample.mp4*. (*CSS only – no JS is allowed*).

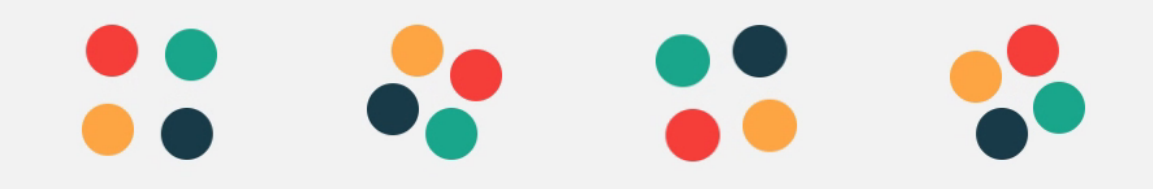
A2: Jumper Ball (Easy)

Make a loop animation of a jumper ball respecting all states from an animation (*CSS only – no JS is allowed*).

A3: Loading Screen (Easy)

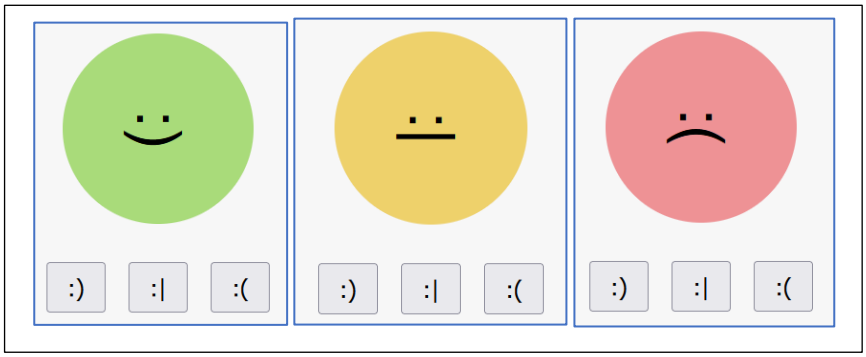
Create an animated loader using only HTML and CSS same as sample.mp4. The loader animation should be looped. 4 different colours (#19A68C, #F63D3A, #FDA543, #193B48) should be used. (*CSS only – no JS is allowed*).

Screen shots:



A4: Smiley (Easy)

Create a big smiley face on the centre of the page and it is 200px width and height. Below the smiley, there are 3 buttons with texts: **:)**, **:|**, and **:(**. By clicking the buttons, the smiley face changes to that state with a 0.5s animation. By default, the smiley is in **:|** state. The states have these designs:



A5: Grayscale image (Easy)

The imagine is open in a grayscale, and when the mouse is hovered over the imagine, the image is displayed with full colours.

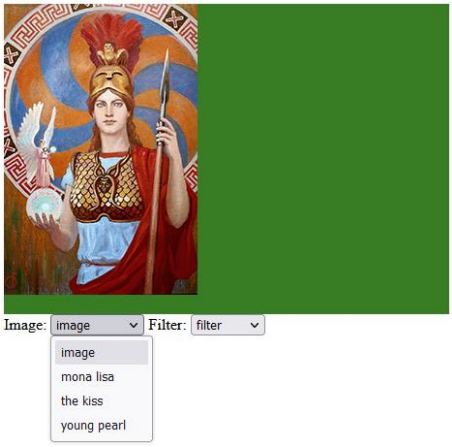
Part B: Medium

B1: Simple Image Filter (Medium)

For this project you must create a function that allows you to add a filter in a certain image using canvas. This speed Project is divided into two parts.

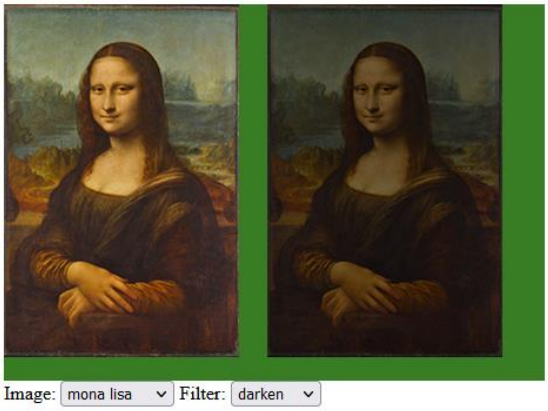
Part 1 - Develop the following design:

- The size of the canvas should be 460 pixels wide and 320 pixels high.
- Create a drop-down list, with the image options: athena.jpg, the-kiss.jpg, young-pearl.jpg and monalisa.jpg. There must be a first option called image.
- Load the images on the canvas depending on the option chosen in the drop-down list.
- The image of athena.jpg is loaded by default.



Part 2

- Create a drop-down list, called filter with the options: Darken and Lighten. There must be a first option called filter.
- When choosing a specific image in the dropdown list and applying a filter, the result should appear on the right side of the canvas according to the chosen option.

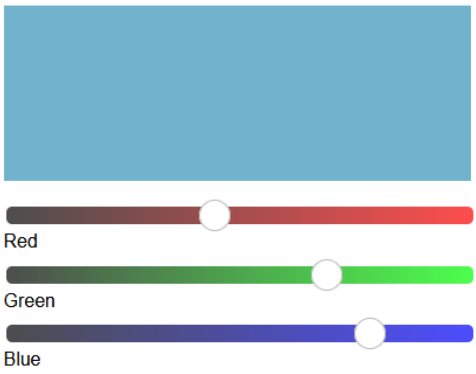


B2: Transition (Medium)

You are given a picture that you need to cut into N cards and each card should disappear animated. X is 5 and Y is 3. Refer to the sample provided (b2_media.mp4). You can use either CSS or JS.

B3 RGB Slider (Medium)

Please implement an RGB slider as following. There are 3 sliders for adjusting values of red, green and blue. Refer to the sample provided (b3_media.mp4)

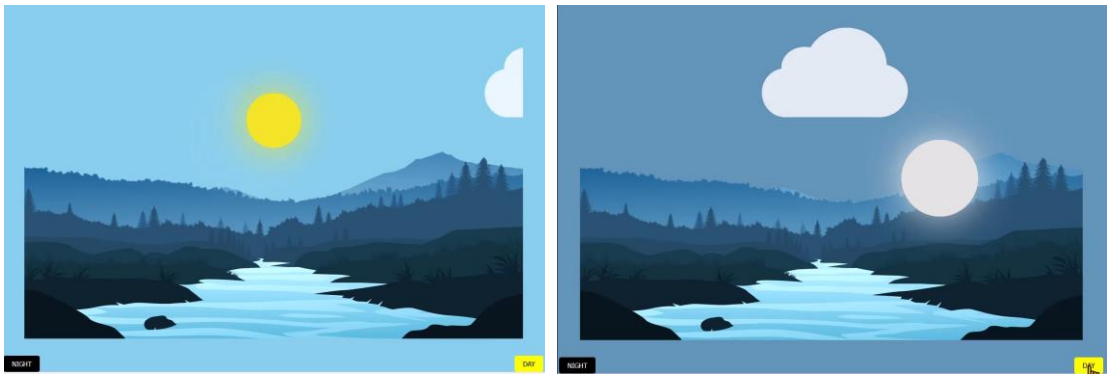


B4 Image blur linear slider (Medium)

Using JavaScript create a linear slider that will blur an imagine progressively left and right. The required effect is presented in the b4_media.mp4. The slider should start as default from 50% of the imagine.

B5 Sunrise and sunset (Medium)

Create a sunrise and sunset, follow the video b5_sample.mp4 as an example of how it should work and look. The images are located in the images folder.



Part C: Difficult

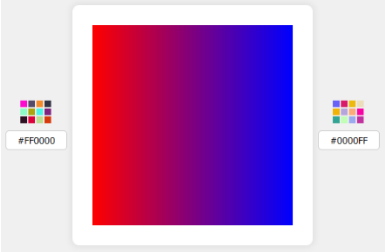
C1: Linear gradient (Difficult)

Create the JavaScript code for the following task:

On the centre of the screen there is a 300 px sized square. By default, it has a red to blue linear gradient background. On the left and right of the square there are 1-1 input fields where you can change the starting, and ending colour of the linear gradient, by writing in the HEX value of the colour.

Over the inputs there are 12 randomized colour buttons (10px*10px) on each side. If you click one of the randomized colours, its colour is inserted in the input. Whenever any input is changed, and is a valid hex, that side's linear gradient colour changes on the square.

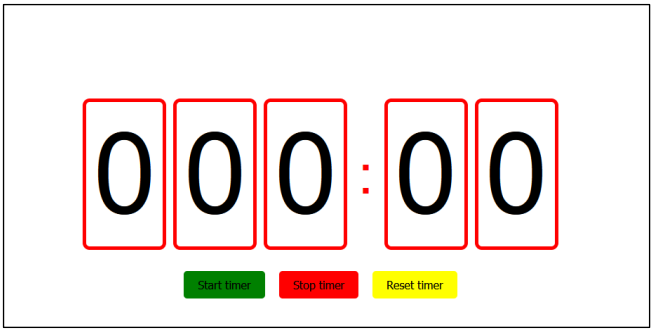
If you manually input any valid HEX value the colour should change to the given value.



See c1_media.mp4

C2: Digital Timer (Difficult)

Time format is: 3 digits of seconds passed, and cent seconds (*a unit of time equal to 0.01 seconds*) following it. Its max value is: "**999:99**" seconds. **Start timer** button starts, **Stop timer** button stops, **Reset timer** button stops and resets to 0 the timer.



C3: Roman Converter (Difficult)

Create a function that will take either a string containing a roman numeral, or an integer.

I ➞ 1, V ➞ 5, X ➞ 10, L ➞ 50, C ➞ 100, D ➞ 500, M ➞ 1000

Given a string, return the integer value of that roman numeral.

Given an integer, return the equivalent roman numeral.

C4: Analog Clock (Difficult)

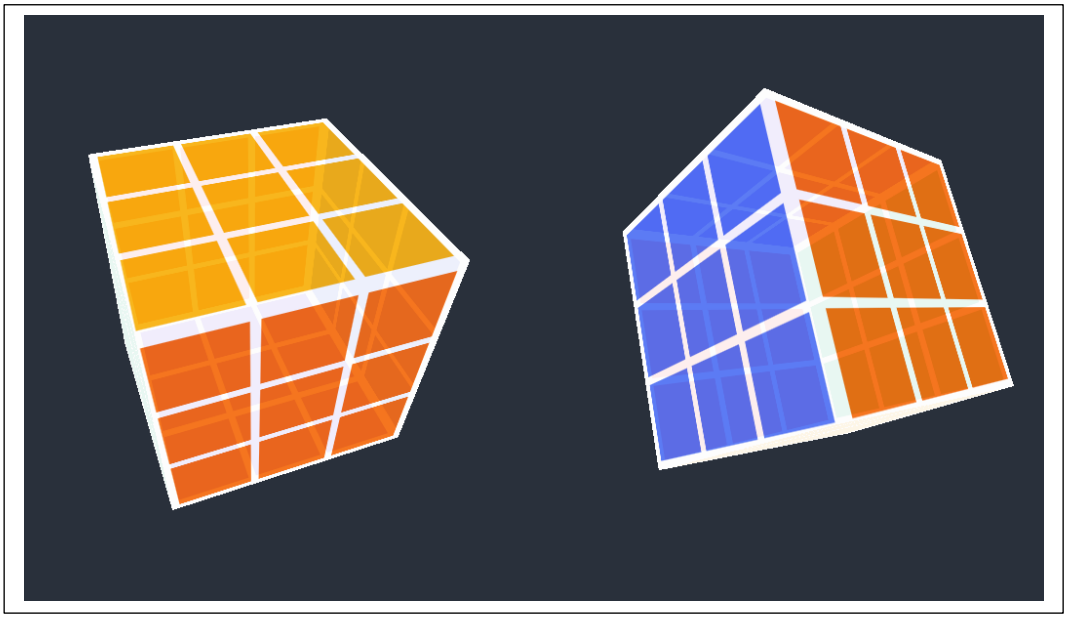
You have to implement an analog clock showing the current local system time.

The background should look as depicted in the media file "full-clock-fancy.png" containing clock border and ticks for hour and minute. Also, the clock hands should look as depicted in the image. Seconds clock hand is updated every second, minutes clock hand is updated every minute, hour clock hand is updated every minute.

The image “full-clock-fancy.png “can be used as you wish.

C5: Cube Rotation (Difficult)

We want to create the following cube rotation effect.



a) You should create two cubes using the index.html and the cubes should be rotated same as c5_sample.mp4.
b) Left cube should be rotated left to right.
c) Right cube should be rotated up to down.
d) Colour codes
    - front - #fa5252
    - back - #f76707
    - right - #12b886
    - left - #4c6ef5
    - top - #fab005
    - bottom - #7950f2
e) You can only use ***style.css*** for the Cube Animation (*CSS only – no JS is allowed*).