



# Programmation d'applications mobiles 1

Séance de laboratoire 1

HTML, CSS,  
JAVASCRIPT

## Table of Contents

<b>OBJECTIF :</b> .....	<b>3</b>
<b>HTML &amp; CSS :</b> .....	<b>3</b>
<b>A- HEADER.....</b>	<b>3</b>
<b>B- SECTION .....</b>	<b>4</b>
<b>C- JOURS DE LA SEMAINE .....</b>	<b>5</b>
<b>D- JOURS DU MOIS ET BOUTONS D’ACTIONS.....</b>	<b>6</b>
<b>JAVASCRIPT :</b> .....	<b>7</b>
<b>A- MOMENT.JS (OPTIONNEL) .....</b>	<b>7</b>
<b>B- GENERER LES JOURS DU MOIS COURANT .....</b>	<b>8</b>
<b>C- LES BOUTONS « PREVIOUS » ET « NEXT ».....</b>	<b>8</b>
<b>D- SELECTIONNER LE JOUR COURANT .....</b>	<b>8</b>
<b>E- AJOUTER LA SECTION DES TACHES .....</b>	<b>9</b>
<b>F- LISTE DES TACHES.....</b>	<b>11</b>
<b>G- SUPPRESSION DES TACHES AU DOUBLE CLIQUE .....</b>	<b>12</b>

## Objectif :

L'objectif de cette séance de laboratoire est:

- De se familiariser avec la mise en page d'un document HTML
- De mettre en pratique nos connaissances en CSS afin d'ajouter du style à la page
- D'utiliser le langage JavaScript pour concrétiser le comportement souhaité

## HTML & CSS :

Nous voulons construire un calendrier qui affiche le mois courant et les jours associés.

Voici un aperçu du rendu final de notre calendrier :

Calendar App						
JULY 2021						
Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
27 June	28 June	29 June	30 June	01 July	02 July	03 July
04 July	05 July	06 July	07 July	08 July	09 July	10 July
11 July	12 July	13 July	14 July	15 July	16 July	17 July
18 July	19 July	20 July	21 July	22 July	23 July	24 July
25 July	26 July	27 July	28 July	29 July	30 July	31 July
Previous				Next		

### a- Header

Ajouter l'entête de la page avec le style CSS correspondant. La couleur utilisée est « teal » :

`background-color: teal;`

Calendar App

Code Html :

@ Gabriel Y. Gamy

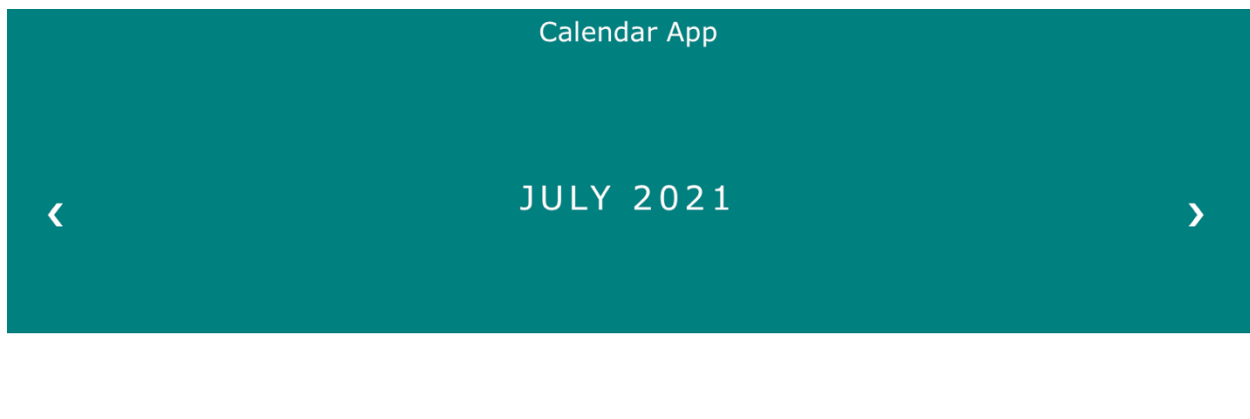
```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Calendar</title>
    <link rel="stylesheet" href="styles.css" />
  </head>
  <body>
    <header>Calendar App</header>
  </body>
</html>

```

## b- Section

Ajouter la section du mois et les deux icônes « previous » et « next » pour avancer ou reculer les mois du calendrier.



Code Html :

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Calendar</title>
    <link rel="stylesheet" href="styles.css" />
  </head>
  <body>
    <header>Calendar App</header>

    <section class="calendar-container">

```

```

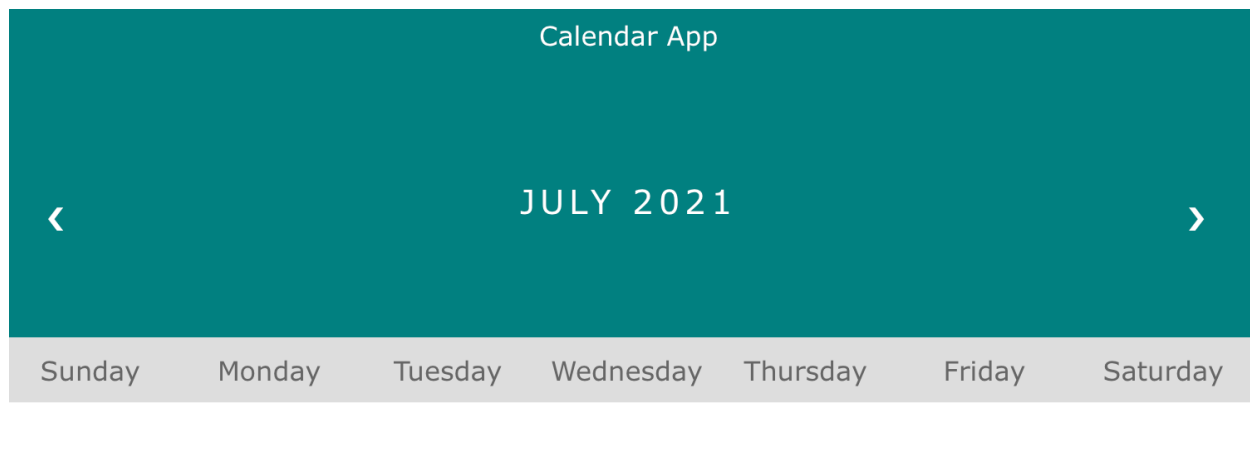
<div class="month">
  <ul>
    <li class="prev">#10094;</li>
    <li class="next">#10095;</li>
    <li id="month-text">August 2021</li>
  </ul>
</div>
</section>

</body>
</html>

```

### c- Jours de la semaine

Ajouter les jours de la semaine en commençant par Dimanche.



Code Html :

```

<!DOCTYPE html>
<html lang="en">
  <body>
    ...

    <section class="calendar-container">
      <div class="month">
        <ul>
          <li class="prev">#10094;</li>
          <li class="next">#10095;</li>
          <li id="month-text">August 2021</li>
        </ul>
      </div>

      <ul class="weekdays">
        <li>Sunday</li>
        <li>Monday</li>

```

```

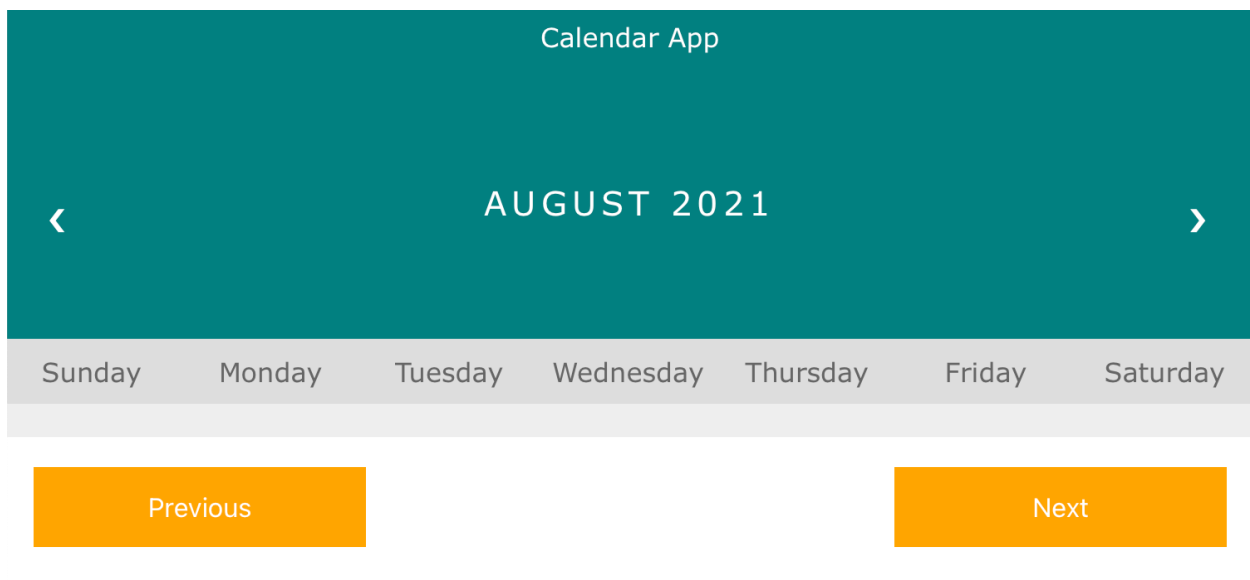
        <li>Tuesday</li>
        <li>Wednesday</li>
        <li>Thursday</li>
        <li>Friday</li>
        <li>Saturday</li>
    </ul>
</section>
</body>
</html>

```

#### d- Jours du mois et boutons d'actions

Il s'agit maintenant d'insérer un bloc dans lequel nous allons construire une liste d'éléments pour les jours du mois.

Ajouter aussi les boutons d'actions « Previous » et « Next » qui auront le même comportement que les icônes d'en haut.



#### Code Html

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta http-equiv="X-UA-Compatible" content="IE=edge" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0" />
  <title>Calendar</title>
  <link rel="stylesheet" href="styles.css" />
</head>
<body>

```

```

<header>Calendar App</header>

<section class="calendar-container">
  <div class="month">
    <ul>
      <li class="prev">&#10094;</li>
      <li class="next">&#10095;</li>
      <li id="month-text">August 2021</li>
    </ul>
  </div>

  <ul class="weekdays">
    <li>Sunday</li>
    <li>Monday</li>
    <li>Tuesday</li>
    <li>Wednesday</li>
    <li>Thursday</li>
    <li>Friday</li>
    <li>Saturday</li>
  </ul>

  <ul id="calendar-days" class="days"></ul>

  <div class="calendar-actions">
    <button class="calendar-actions__btn js-prev" type="button">
      Previous
    </button>
    <button class="calendar-actions__btn js-next" type="button">
      Next
    </button>
  </div>
</section>
</body>
</html>

```

## JavaScript :

Nous voulons maintenant avoir un comportement à notre calendrier. D'abord permettre de générer les jours du mois et les insérer dans le bloc `<ul id="calendar-days" class="days"></ul>`, et ensuite implémenter les actions des boutons « Previous » et « Next ».

### a- Moment.js (optionnel)

***Vous pouvez le faire à votre manière sans utiliser l'API moment.js***

L'API JavaScript [moment.js](https://momentjs.com/) offre une large collection de fonctionnalités de manipulation des dates et heures. Nous pouvons l'utiliser pour générer la liste des jours du mois.

Insérer la balise script de l'api moment.js au bas de la page Html.

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/moment.js/2.29.1/moment.min.js">
</script>
```

Insérer la balise script de notre code JavaScript au bas de la page Html.

```
<script src="calendar.js" defer></script>
```

#### b- Générer les jours du mois courant

Maintenant nous allons construire les jours du calendrier en Javascript à l'aide de notre API moment.js.

L'objectif ici est de remplir les éléments suivants :

```
/** Element HTML <ul id="calendar-days"></ul> */
var calendarDaysElement = document.getElementById("calendar-days");

/** Element HTML <li id="month-text"> */
var monthTextElement = document.getElementById("month-text");
```

#### c- Les boutons « Previous » et « Next »

À l'aide des événements JavaScript « click », permettre la navigation dans le calendrier. Le bouton « Previous » permet d'afficher le mois précédent, tandis que le bouton « Next » permet d'afficher le mois suivant.

Vous pouvez à l'aide de JavaScript obtenir les éléments HTML des boutons « Previous » et « Next » afin d'y ajouter le comportement souhaité.

```
const prevButtons = document.getElementsByClassName("js-prev");

const nextButtons = document.getElementsByClassName("js-next");
```

#### d- Sélectionner le jour courant

Ajouter une logique pour sélectionner la journée d'aujourd'hui. Il faudrait utiliser une classe CSS sur le jour actif.

Il faut aussi un événement « click » sur tous les jours du calendrier de sorte qu'en cliquant sur un jour du calendrier, ce jour devient automatiquement le jour actif.

Par exemple, pour la journée du 27 Juillet 2021.

@ Gabriel Y. Gamy



JULY 2021						
Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
27 June	28 June	29 June	30 June	01 July	02 July	03 July
04 July	05 July	06 July	07 July	08 July	09 July	10 July
11 July	12 July	13 July	14 July	15 July	16 July	17 July
18 July	19 July	20 July	21 July	22 July	23 July	24 July
25 July	26 July	27 July	28 July	29 July	30 July	31 July
Previous			Next			

#### e- Ajouter la section des tâches

Ajouter en dessous des boutons, une nouvelle section pour permettre l'ajout des tâches pour la journée active.

- Un changement de la journée active entraine un affichage de la liste des tâches de cette journée
- Un double clic sur une tâche entraine la suppression de cette tâche.
- Un maximum de cinq tâches supportées. Après un dépassement, afficher un message d'erreur sous forme d'alerte.
- Pour une journée sans tâches, afficher un message « No tasks found »

La section contient :

- Une zone pour afficher les taches dans une liste ordonnée
- Un champ de saisie pour saisir la tâche que l'on souhaite ajouter
- Un bouton d'ajout

Code HTML :

```
<section class="calendar-tasks">
  <div class="calendar-tasks__title">Tasks</div>
  <div class="calendar-tasks__info">
    * Double click on the task to perform a delete action
  </div>
```

@ Gabriel Y. Gamy

```
<ol class="calendar-tasks__items">
  No tasks found
</ol>
<div class="calendar-tasks__save">
  <div class="calendar-tasks__save-title">
    Add a new task:
  </div>
  <textarea class="calendar-tasks__save-input"
    rows="4"
    cols="50"
  ></textarea>
  <button class="calendar-tasks__save-btn" type="button">
    Add task
  </button>
</div>
</section>
```

Tasks

*\* Double click on the task to perform a delete action*

No tasks found

Add a new task:

Add task

<

JULY 2021

>

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
27 June	28 June	29 June	30 June	01 July	02 July	03 July
04 July	05 July	06 July	07 July	08 July	09 July	10 July
11 July	12 July	13 July	14 July	15 July	16 July	17 July
18 July	19 July	20 July	21 July	22 July	23 July	24 July
25 July	26 July	27 July	28 July	29 July	30 July	31 July

Previous

Next

Tasks

*\* Double click on the task to perform a delete action*

No tasks found

Add a new task:

Add task

## f- Liste des tâches

Implémenter le bouton « Add Task ». La liste des tâches devra être affichée comme ci-dessous :

Tasks

*\* Double click on the task to perform a delete action*

- Learn HTML, CSS and Javascript
- Learn React
- Learn React native

Add a new task:

Add task

#### g- Suppression des tâches au double clique

Pour supprimer une tâche, il faut double cliquer sur cette tâche.  
Implémenter cette fonctionnalité.