

HTML / CSS / PHP

Todolist

Le projet

Description du projet

Nous devons réaliser une application de Todo list (tâches à faire).

En partant de la maquette graphique proposé par notre équipe design, nous allons créer l'application de façon à ce qu'elle soit parfaitement fonctionnelle sur un navigateur.

Pour cela nous allons utiliser **HTML**, un langage descriptif qui définit une page web, pour réaliser la structure de notre application.

Afin de respecter les bonnes pratiques en vigueur nous allons utiliser **CSS**, langage déclaratif utilisé pour décrire la présentation de pages web, pour les aspects graphiques (couleurs, taille des polices de caractères, espacements, ...)

Enfin, pour gérer les interactions avec notre application, nous utiliserons **PHP**, langage de programmation côté serveur

Todo List

Gestionnaire de tâches

Ajouter une tâche

ou



Passer le balai



Saluer le boss



Couper l'ordi



Chapitre 1

Etape 1 : les bases HTML & CSS

Intégration

De la maquette à une
représentation en blocs.

Todo List

Gestionnaire de tâches

Ajouter une tâche

 ou

☐

 Passer le balai

☒

~~Saluer le boss~~

☐

 Couper l'ordi

Todo List

Gestionnaire de tâches

Ajouter une tâche

ou



Passer le balai



Saluer le boss



Couper l'ordi



Todo List

Gestionnaire de tâches

Ajouter une tâche

ou



Passer le balai



Saluer le boss



Couper l'ordi



Intégration

Des blocs aux éléments HTML

[...] HTML est utilisé pour dire si votre contenu web doit être reconnu en tant que paragraphe, liste, en-tête, lien, image, lecteur multimédia, formulaire ou l'un des nombreux autres éléments disponibles ou même un nouvel élément à définir par vous même.

<https://developer.mozilla.org/fr/docs/Learn/HTML>

Todo List

Gestionnaire de tâches

Ajouter une tâche

ou



Passer le balai



Saluer le boss



Couper l'ordi



<h1>

<h2>

<form>

<label>

<div>

<input>

<select>

<button>

<checkbox>

<button>

<checkbox>

<button>

<checkbox>

<button>

HTML

Documentations

En HTML une balise est utilisée pour créer un element.

Le nom d'un élément HTML est le nom utilisé dans des chevrons comme par exemple `<p>` pour un paragraphe.

Notez que le nom de la balise fermante est précédé par un caractère barre oblique, "`</p>`", et que pour les éléments vides la balise de fin n'est pas nécessaire ni permise (on n'oubliera pas le `/` juste avant le chevron, ces balises sont dites « auto-fermantes »).

Liste de toutes les balises HTML

<https://developer.mozilla.org/en-US/docs/Web/HTML/Element>

Pense bête (liste non exhaustive !)

Liste de toutes les balises HTML
<https://developer.mozilla.org/en-US/docs/Web/HTML/Element>

HTML

Documentations

Les attributs

On peut préciser les balises HTML avec des attributs pour fournir des informations complémentaires qui vont modifier la façon dont le navigateur va interpréter l'élément :

Anatomie d'un élément HTML



Un *attribut* vient compléter un tag (ou balise). Sa présence peut être requise ou facultative. Il peut fournir des méta-informations ou changer le comportement du tag. La syntaxe est **name=value** où *name* est l'identificateur de l'attribut et *value* sa valeur attribuée.

En savoir plus

<https://developer.mozilla.org/fr/docs/Glossary/HTML>

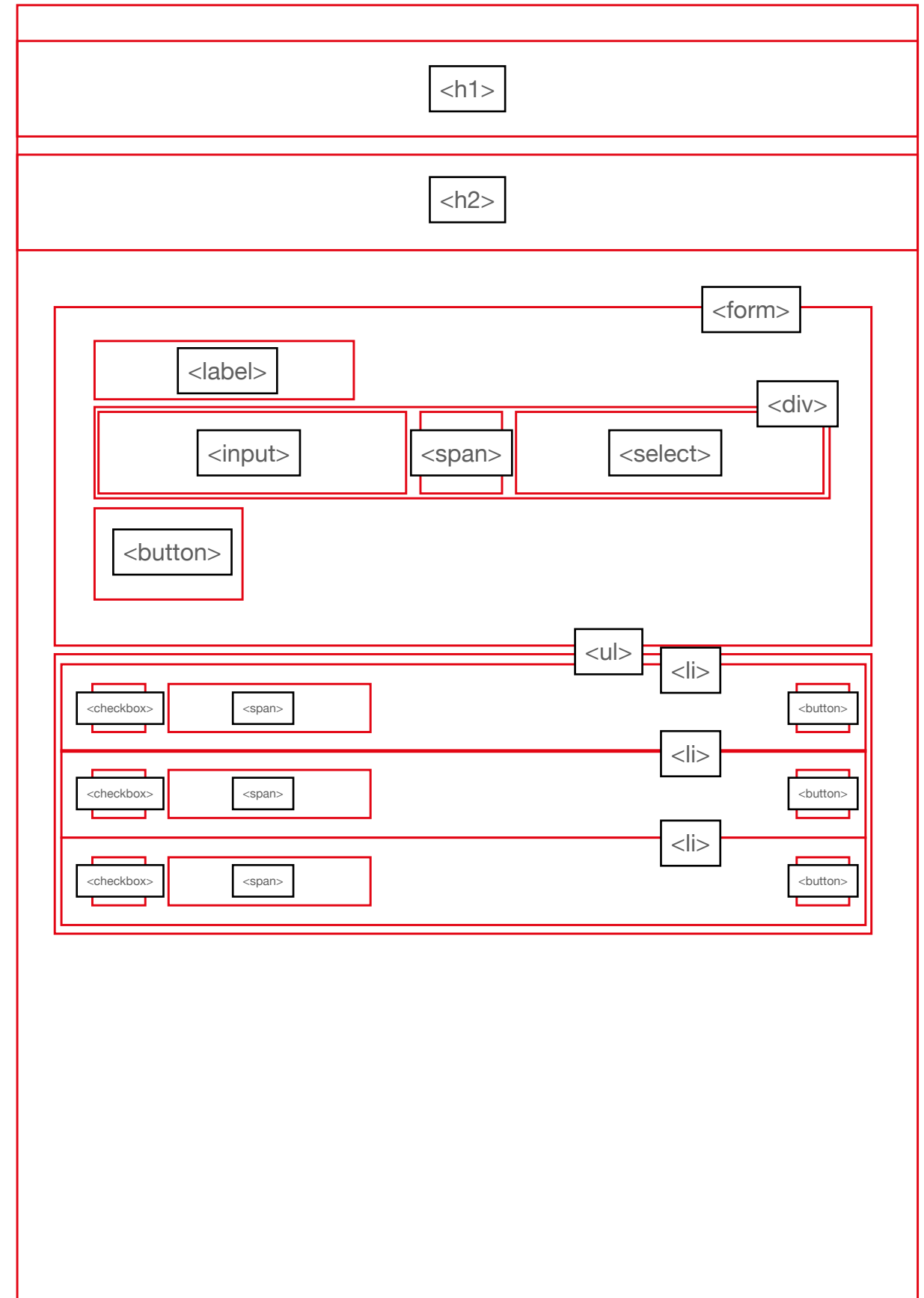
<https://developer.mozilla.org/fr/docs/Glossary/Attribute>

Exercice

Créer un fichier index.php et le code HTML correspondant en vous basant sur le schéma ci-contre.

A ce stade, il n'y a pas de code PHP à écrire.

Temps : 1H



Résultat

Todo List

Gestionnaire de tâches

Ajouter une tâche

- ☐ Passer le balai ☐
- ☒ Saluer le boss ☐
- ☐ Couper l'ordi ☐

CSS

Documentation

Une règle CSS est un ensemble de propriétés associé à un sélecteur.

Un sélecteur CSS est la partie de la règle CSS qui désigne les éléments d'un document ciblés par cette règle. Les éléments correspondants se verront appliquer la mise en forme indiquée par la règle.

Une propriété CSS est une caractéristique (telle que **color**) dont la valeur associée définit un aspect de la manière dont le navigateur doit afficher l'élément.

Exemple de règles CSS

```
h1 {  
  margin: 30px 0;  
  text-align: center;  
  font-size: 30px;  
}
```

```
h2 {  
  margin-bottom: 50px;  
  text-align: center;  
  font-size: 20px;  
}
```

```
.list-todo li.ok span {  
  text-decoration: line-through;  
}
```

En savoir plus

https://developer.mozilla.org/fr/docs/Learn/CSS/First_steps/Getting_started

CSS

Pense bête (liste non exhaustive !)

Sizes and Colours

0 0 requires no unit

Relative Sizes

em 1em equal to font size of parent (same as 100%)

ex Height of lower case "x"

% Percentage

Absolute Sizes

px Pixels

cm Centimeters

mm Millimeters

in Inches

pt 1pt = 1/72in

pc 1pc = 12pt

Colours

#789abc RGB Hex Notation

#acf Equates to "#aaccff"

rgb(0,25,50) Value of each of red, green, and blue. 0 to 255, may be swapped for percentages.

Color / Background

color + background-repeat

background x background-image

background-color background-position

background-attachment

Text

text-indent + word-spacing +

text-align + text-transform +

text-decoration white-space +

text-shadow line-height +

letter-spacing +

Fonts

font + x font-weight +

font-family + font-stretch +

font-style + font-size +

font-variant + font-size-adjust +

Interface

cursor + outline-style

outline x outline-color

outline-width

Visual

volume + elevation

speak + speech-rate

pause x voice-family

pause-before pitch

pause-after pitch-range

cue x stress

cue-before richness

cue-after speak-punctuation

play-during speak-numeral

azimuth +

Iscellaneous

content list-style-type +

quotes + list-style-image +

counter-reset list-style-position +

counter-increment marker-offset

list-style + x

Note

Shorthand properties are marked x
Properties that inherit are marked +

Available free from
www.AddedBytes.com

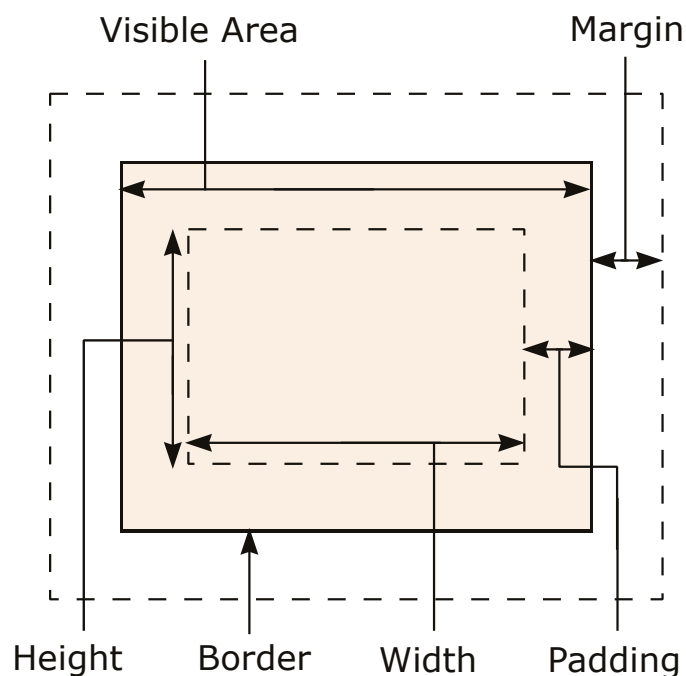
Selectors

*	All elements
div	<div>
div *	All elements within <div>
div span	 within <div>
div, span	<div> and
div > span	 with parent <div>
div + span	 preceded by <div>
.class	Elements of class "class"
div.class	<div> of class "class"
#itemid	Element with id "itemid"
div#itemid	<div> with id "itemid"
a[attr]	<a> with attribute "attr"
a[attr='x']	<a> when "attr" is "x"
a[class~='x']	<a> when class is a list containing 'x'
a[lang ='en']	<a> when lang begins "en"

Pseudo-Selectors and Pseudo-Classes

:first-child	First child element
:first-line	First line of element
:first-letter	First letter of element
:hover	Element with mouse over
:active	Active element
:focus	Element with focus
:link	Unvisited links
:visited	Visited links
:lang(var)	Element with language "var"
:before	Before element
:after	After element

Box Model



Positioning

display	clear
position	z-index
top	direction +
right	unicode-bidi
bottom	overflow
left	clip
float	visibility

Dimensions

width	min-height
min-width	max-height
max-width	vertical-align
height	

Boxes

margin x	border-color x
margin-top	border-top-color
margin-right	border-right-color
margin-bottom	border-bottom-color
margin-left	border-left-color
padding x	border-style x
padding-top	border-top-style
padding-right	border-right-style
padding-bottom	border-bottom-style
padding-left	border-left-style
border x	border-width x
border-top x	border-top-width
border-bottom x	border-right-width
border-right x	border-bottom-width
border-left x	border-left-width

Tables

caption-side +	border-spacing +
table-layout	empty-cells +
border-collapse +	caption-header +

Paging

size	page-break-inside +
marks	page +
page-break-before	orphans +
page-break-after	widows +

Exercice

Créer un fichier style.css l'intégrer à la page index.php

Looker l'application via CSS afin de respecter le design attendu

Todo List

Gestionnaire de tâches

Ajouter une tâche

ou

☐

Passer le balai

☒

Saluer le boss

☐

Couper l'ordi