

## Documents

### 1.1 élément et balise

Un élément HTML peut être soit constitué d'une paire de balises (ouvrante et fermante) et d'un contenu, soit d'une balise unique qu'on dit alors « orpheline ». L'élément p (qui sert à définir un paragraphe) est par exemple constitué d'une balise ouvrante, d'une balise fermante et d'un contenu textuel entre les balises :

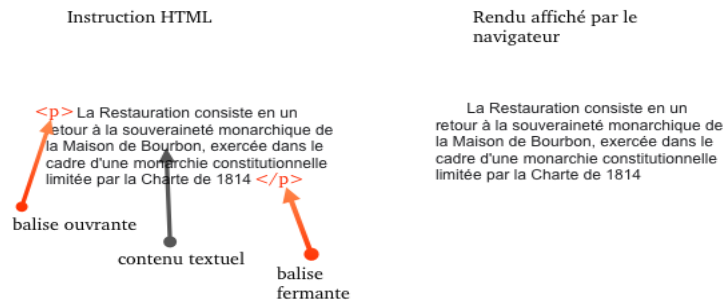


FIGURE 1 – élément p

### 1.2 inline et bloc

1. A partir de l'exemple ci-dessous : Quel élément est de type *inline*, et quel élément est de type *bloc*? Comment ceux-ci se disposent-ils les uns par rapports aux autres?
2. Quelle est la largeur par défaut d'un élément de type *bloc*? Et de type *inline*?

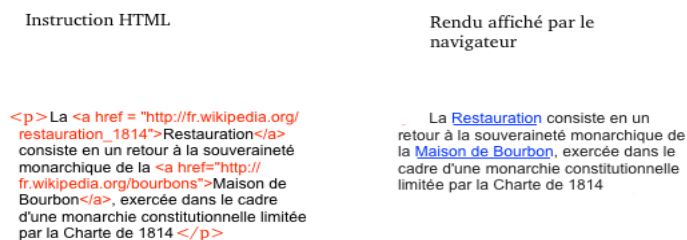


FIGURE 2 – éléments p et a

## Exercice : CSS pour le texte

Parmi les paramètres suivants, lesquels peuvent être employés pour des règles de style relatives à un élément textuel ?

```
1 color
2 background-color
3 font-family
4 font
5 text-align
6 font-weight
```

Parmi les paramètres suivants, lesquels peuvent être appliqués à un élément de type bloc ?

```
1 border
2 margin
3 padding
4 vertical-align
5 width
6 height
```

Parmi les paramètres suivants, lesquels peuvent être appliqués à un élément de type inline ?

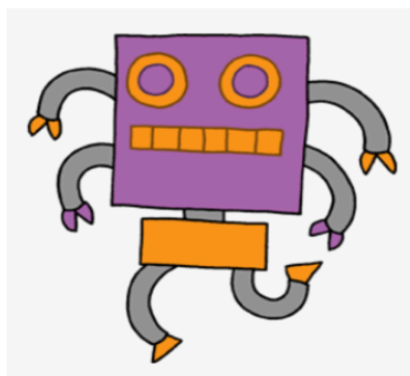
```
1 border
2 margin
3 padding
4 vertical-align
5 width
6 height
```

## Exercice : l'arbre du DOM

On étudie le lien entre les balises pour la page suivante :

### Wanted!

Have you seen this robot?



Description: Height: 30cm, Colour: purple and orange, Arms: 4

If you have any information, please contact 6207 332 2310

FIGURE 3 – Exemple de page HTML

Voici le script HTML pour cette page :

```

1 <html>
2   <head>
3     <link rel="stylesheet" href="style.css">
4   </head>
5   <body>
6     <div>
7       <h1>Wanted!</h1>
8       <h3>Have you seen this robot?</h3>
9       
10      <p>Description: Height: 30cm, Colour: purple and orange, Arms: 4</p>
11      <p>If you have any information, please contact 6207 332 2310</p>
12    </div>
13  </body>
14 </html>

```

1. Représenter l'arbre du DOM pour cette page.
2. Y-a-t-il des éléments qui sont frères (c'est-à-dire qui ont le même parent)? Lesquels?

Partie 4

## Formulaires

Les formulaires permettent à l'utilisateur de saisir un **ensemble de données** et d'envoyer une **requête** à un serveur en cliquant sur un bouton. Le formulaire est délimité par l'élément HTML `<form>` qui précise :

- L'URL (attribut `action`)
- La méthode (attribut `method`)

Entrez votre nom

Selectionner une temperature

Entrer un nombre

Aimez vous la soupe ?

Oui

Non

Quelle est votre langue marternelle ?

FIGURE 4 – Exemple de formulaire

Il contient un ensemble d'éléments interactifs permettant la saisie de données, spécifiés par l'élément `<input>`. Son attribut `type` permet de préciser le type de saisie :

- Text
- Number
- Range (valeur numerique)
- Checkbox
- Radio
- Button

- ...

Les type *reset* et *submit* sont des boutons spéciaux pour réinitialiser ou envoyer la requête vers le serveur. L'attribut *value* spécifie la valeur initiale du champ de saisie.

Voici un exemple de script HTML pour un formulaire.

1. Quels sont les attributs obligatoires pour les balises des éléments *form*, *label* et *input*? Quel est leur rôle?
2. Représenter l'élément *input* de type *text* tel qu'il sera affiché dans la page.

```

1 <form method="post" action="action.php">
2   <p>
3     <label for="prenom">Entrez votre prénom :</label>
4     <input type="text" name="prenom" id="prenom" placeholder="Ex: Jojo"
      size="30" maxlength="10" />
5   </p>
6 <p>
7   <input type="submit" value="Envoyer" />
8 </p>
9 </form>

```

Partie 5

## Position

### 5.1 Positionner des carrés l'un par rapport à l'autre

Deux carrés sont positionnés de la manière suivante



FIGURE 5 – position relative de 2 carrés

Le script correspondant est le suivant :

```

1 <div class="parent-div">
2   <div class="child-div"></div>
3 </div>
4
5 <style>
6 .parent-div{
7   position: relative;
8   background-color: #CCC;
9   width: 200px;
10  height: 200px;
11 }
12
13 .child-div{
14   display: inline-block;
15   width: 50px;
16   height: 50px;

```

```

17 position: absolute;
18 right: 0px;
19 bottom: 0px;
20 background-color: blue;
21 }
22 </style>

```

Pour positionner un élément fils par rapport à son parent, les 2 éléments doivent avoir la propriété CSS suivante :

- pour le parent : `position: relative`
- pour l'enfant : `position: absolute`

Il y a alors 2 options pour le positionnement :

- Option 1 : On définit alors la position de l'enfant par rapport aux rebords du parent, avec les paramètres `right` (bord droit), `left`, `top` (bord supérieur), `bottom`.
- Option 2 : utiliser des marges pour les éléments : `margin-left`, `margin-right`, ...

1. Quel est la classe de l'élément parent ?
2. Quel est la classe de l'élément fils ?
3. Peut-on interpréter cette relation parent-fils à partir du script HTML ?
4. Quelles sont les règles CSS qui permettent de positionner l'élément fils ?
5. Comment faut-il modifier ces règles pour centrer le petit carré dans le grand ?

## 5.2 Design d'une application de SMS

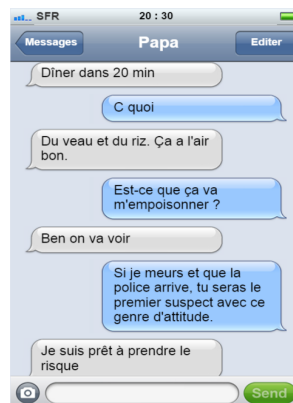


FIGURE 6 – blague

En vous inspirant de l'exercice précédent, créer un mini dialogue entre 2 personnes (une question, une réponse, une autre question). Ajouter les règles de style pour que la présentation de la conversation ressemble à celle d'une application de messagerie instantanée.

## Règles CSS orientée objet

Souvent, les objets semblables partagent les mêmes règles, avec certaines différences. Comme par exemple, ici, pour 2 boutons rouge et vert :

```
1 .boutonRouge {
2     display: block;
3     font-size: 12px;
4     color: red;
5     border: 1px solid red;
6 }
7
8 .boutonVert {
9     display: block;
10    font-size: 12px;
11    color: red;
12    border: 1px solid red;
13 }
```

On souhaite rassembler dans une même classe les règles communes à tous les boutons. Et créer d'autres classes selon les différences entre ces boutons.

On utilisera alors la balise HTML suivante pour appeler le bouton rouge ; l'élément div a alors 2 classes :

```
1 <div class="bouton bouton-rouge">Bouton rouge</div>
```

Modifier les règles CSS en partageant celles-ci entre les classes :

- bouton
- bouton-rouge
- bouton-vert

## Correction

### 7.1 Exercice positionnement des carrés

```
1 <div class="parent-div">
2   <div class="child-div"></div>
3 </div>
```

```
1 .parent-div{
2   position: relative;
3   background-color: #CCC;
4   height: 200px;
5 }
6
7 .child-div{
8   display: inline-block;
9   width: 50px;
10  height: 50px;
11  position: absolute;
12  right: 0px;
13  bottom: 0px;
14  background-color: blue;
15 }
```

### 7.2 Exercice SMS

```
1 <div class="container">
2   <div class = "gauche">
3     Hello
4   </div>
5 </div>
6
7 <div class="container">
8   <div class = "droite">
9     Hey! ça va?
10  </div>
11 </div>
12
13 <div class="container">
14   <div class = "gauche message2">
15     Yep!
16   </div>
17 </div>
```

```
1 .container {
2   width: 200px;
3   position: relative;
4   background-color: red;
5 }
6
7 .gauche, .droite{
8   position: absolute;
9   height: 50px;
10  display: inline-block;
11  width: 100px;
```

```
12     border-radius: 10px;
13 }
14
15 .milieu {
16     height: 10px;
17     background-color: red;
18 }
19
20 .gauche {
21     background-color: yellow;
22
23 }
24
25 .droite {
26     top:60px;
27     //right:0;
28     margin-left:50px;
29     background-color: blue;
30     text-align: right;
31     color: white;
32 }
33
34 .message2 {
35     top: 120px;
36 }
```