

HELLO WORLD

HTML les fondamentaux

ET SI ISAAC ASIMOV AVAIT RAISON ?

Né le 2 janvier 1920, Isaac Asimov est un scientifique américano-russe connu pour ses œuvres de fiction où il explore un futur à la technologie omniprésente. Pionnier en la matière, il a imaginé une immense quantité de scénarios articulés autour du risque de rendre exclusives des connaissances utiles au bien de tous à une forme d'intelligence supposée supérieure. Si l'Internet d'aujourd'hui est accessible à tous, c'est bien parce que ses inventeurs ont voulu apporter ces technologies au plus grand nombre en rendant publiques leurs recherches pour qu'Internet devienne un outil bénéfique au plus grand nombre.



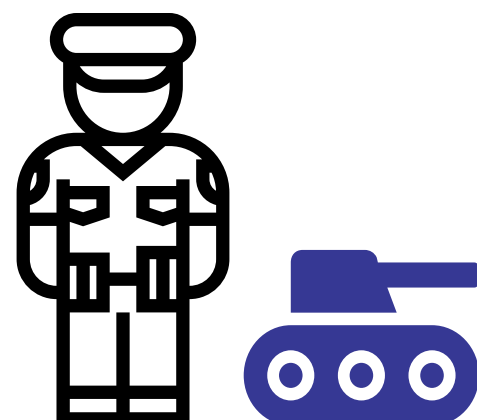
**“ IF KNOWLEDGE CAN
CREATE PROBLEMS, IT IS
NOT THROUGH IGNORANCE
THAT WE CAN SOLVE IT ”**

DU SECRET DÉFENSE À L'OPENSOURCE

1969 : LA DARPA CRÉE

ARPANET

Premier réseau à transfert de paquets développé par la **Defense Advanced Research Projects Agency**, programme de défense nationale Américain.



ANNÉES 1980

PROTOCOLES TCP-IP

Inventés par **Vinton Gray Cerf et Bob Kahn**, cette suite de protocoles permet un gestion de transfert de données en réseaux plus rapide.

Le réseau **Arpanet adopte le 1er janvier 1983** la suite de protocoles TCP/IP qui sera la base d'Internet



INTERNET

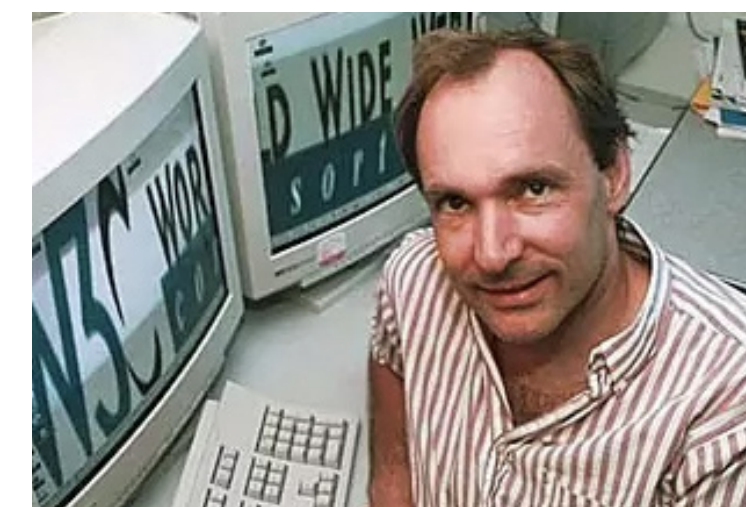
Réseau composé aujourd'hui de millions de réseaux publics ou privés

6 AOÛT 1991

TIMOTHY BERNERS-LEE

Il développe un système hypertexte organisé en toile pour améliorer la diffusion d'informations pour le CERN de Genève **qu'il rendra publique le 6 août 1991**.

En **octobre 1994** il fonde le **World Wild Web Consortium**, organisme à but non lucratif chargé de normaliser les technologies du Web.



LE SQUELETTE DE BASE D'UNE PAGE HTML

Le squelette d'une page HTML est la structure qui permet de créer et d'organiser du contenu, il est constitué de balises obligatoire pour définir le fichier HTML puis une suite de balises sémantiques qui identifie des zones spécifiques.

LE DOCTYPE

Le fichier est configuré en HTML5

```
<!DOCTYPE html>
<html lang="fr">
```

LA BALISE HTML

Cette balise contient tout ce qui constitue la page

```
<head>
```

LA BALISE HEAD

Permet d'ajouter de l'intelligence dans la page

```
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width,...">
<title>Titre du document HTML</title>
</head>
```

LA BALISE BODY

Le corps de la page contient la structure du DOM

```
<body>
```

LES BALISES SÉMANTIQUES

Pour donner de la structuration au contenu d'une page HTML, il faut utiliser les balises qui sont prévues à cet effet que l'on peut séparer en deux groupes.

USAGE UNIQUE

Le **header** contient la **nav** pour la navigation principale, le **main** pour le contenu et le **footer** pour finir la page

USAGE MULTIPLE

Un **section** sert à découper une balise à usage unique, les **articles** découpent les **sections** et les balises **aside** servent à afficher un contenu connexe.

```
<header>
  <!-- Navigation principale -->
  <nav></nav>
</header>
<!-- Contenu principal de la page -->
<main>
  <section>
    <article></article>
    <aside></aside>
  </section>
</main>
<!-- Fermeture de la page -->
<footer></footer>
</body>
</html>
```

DEFINITION DE LA BALISE <HEAD>

Quand nous le regardons de haut en bas, un squelette commence par la tête, l'analogie avec le HTML ne s'arrête pas là car c'est dans la balise <head> que se trouvent les informations nécessaires au fonctionnement du corps de la page. La balise <head> contient des éléments *obligatoires et d'autres optionnelles.

<HEAD>

Ouverture de la balise

`<head>`

*CHARSET UTF-8

Cette balise permet d'afficher correctement les accents.

`<meta charset="utf-8" />`

KEYWORDS

Mots clés de la page (tags)

`<meta name="keywords" content="exercice, html, css">`

VIEWPORT

Surface de la fenêtre pour l'affichage mobile

`<meta name="viewport" content="initial-scale=1">`

*<TITLE>...</TITLE>

Titre de la page

`<title>Bonjour le monde</title>`

<LINK ... >

Permet d'ajouter une feuille de style CSS.

`<link rel="stylesheet" type="text/css" href="css/style.css">`

</HEAD>

Fermeture de la balise

`</head>`

STRUCTURE DES BALISES HTML

Les balises HTML peuvent être de type block ou in-line. Leurs comportements sont différents mais leurs structures restent la même : il faut toujours ouvrir et fermer une balise HTML, lui ajouter un contenu et en option un attribut.

OUVERTURE DE LA BALISE

Les accolades encadrent l'ouverture :
<p> il s'agit d'un paragraphe

FERMETURE DE LA BALISE

Le slash permet de fermer la balise :
</p> pour fermer le paragraphe

<p class="textBjr">Hello World!</p>

ATTRIBUT DE LA BALISE

Peut être de type class , id ou les deux :
"textBjr" est une référence de classe CSS

CONTENU

LES TYPES DE BALISES HTML

Les types des balises HTML ont chacune une fonction particulière : le type block permet de créer des sections et le type inline permet de créer des sous-sections. Les balises de types blocks englobent des balises de type in-line.

BALISES DE TYPE BLOC

Elles occupent toute la largeur disponible. Lorsque deux éléments blocs se suivent dans une page, ils sont positionnés (par défaut) l'un sous l'autre.

<section> Balise sectionnante HTML5
<h1> Balise de titre (1 à 6)
<p> Balise de paragraphe

```
<section>

  <h1>Duis aute irure dolor in reprehenderit</h1>

  <p>Lorem ipsum sit amet</p>

</section>
```

BALISES DE TYPE INLINE

Elles n'occupent que la largeur indispensable à l'affichage du contenu et ne provoquent pas de retour à la ligne.

**** Rend un texte gras
<i> Rend le texte italique
**** Balise pour charger une image

```
<section>

  <p><strong>2010-2015 :</strong>Lorem ipsum sit amet,
  dolor <i>Consectetur</i></p>

</section>
```

FONCTIONNEMENT DES LISTES HTML

Une liste HTML est un paragraphe structuré contenant une suite de lignes, ces lignes peuvent être ordonnées ou non-ordonnées. Pour fonctionner, une liste a besoin d'un conteneur `` ou `` dans lequel sont associées des balises ``.

`...`

Dans une liste ordonnée, chaque `` est numérotés

`...`

Chaque `` correspond à une donnée de la liste

``

`Lorem ipsum dolor`

`Duis aute irure dolor in`

`Excepteur sint occaecat`

``

`...`

Dans une liste non-ordonnée, les `` ne sont pas numérotés, sert à faire un menu

`...`

Quelles soit ordonnées ou non, les `` sont nécessaires

``

`Sed ut perspiciatis unde`

`Neque porro quisquam est`

`Ut enim ad minima veniam`

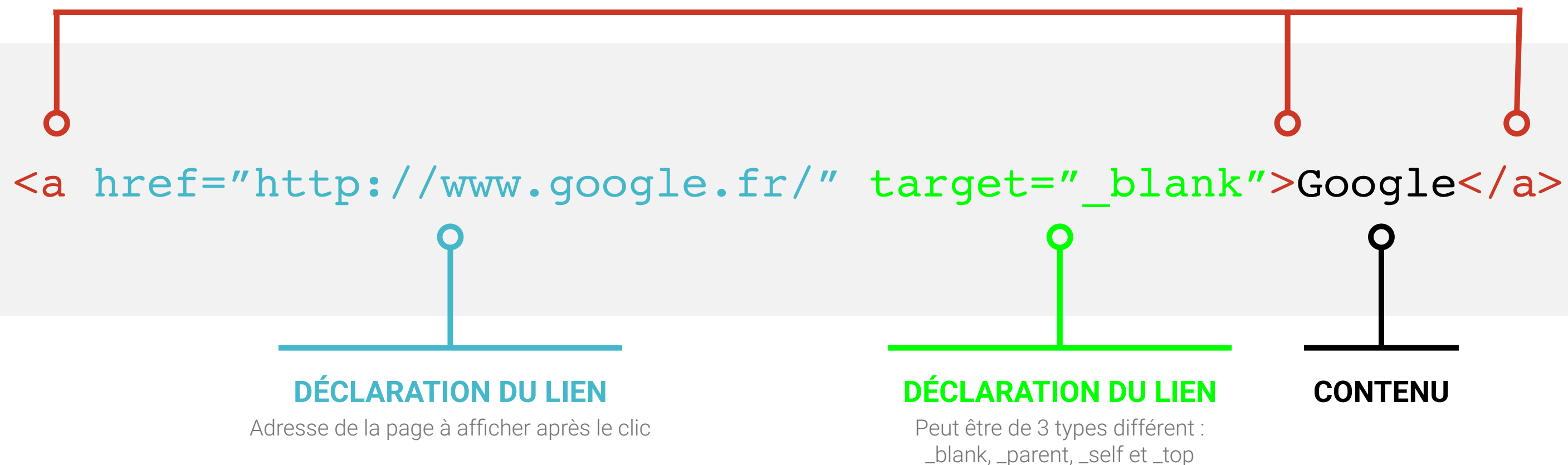
``

LE FONCTIONNEMENT DES LIENS HTML

Les liens HTML peuvent être de deux types : interne ou externe. Les liens internes servent à gérer la navigation sur plusieurs pages d'un même site alors que les externes affichent des pages qui ne font pas partie du site, les liens externes doivent être spécifiés à l'utilisateur.

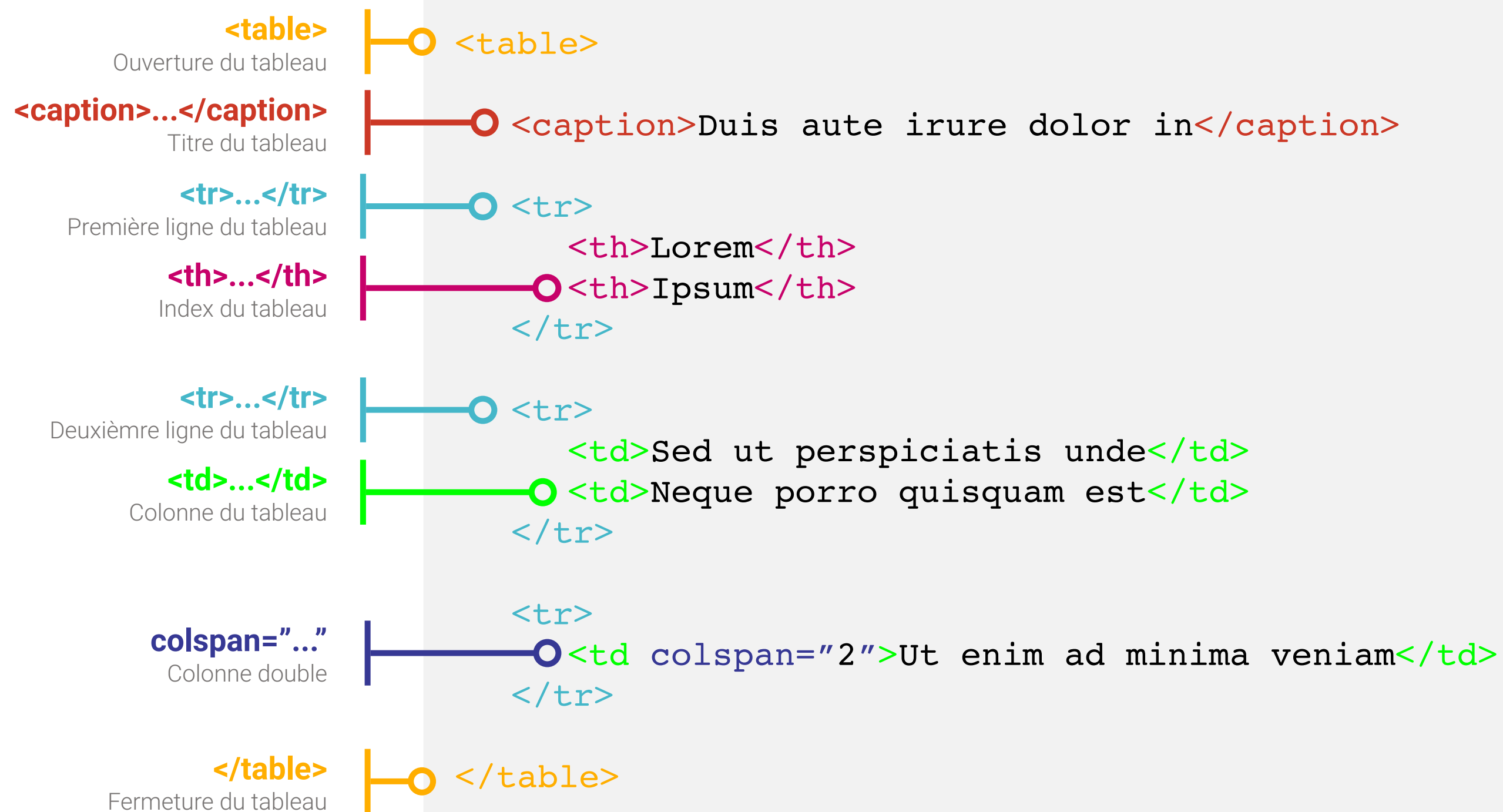
DÉCLARATION DE LA BALISE

Les accolades encadrent la déclaration : `<a>...` il s'agit d'un lien



FONCTIONNEMENT DES TABLEAUX HTML

Les tableaux ont un fonctionnement proche des listes mais sont composés de plus de balises. Il est possible d'afficher un tableau avec différentes options comme un titre ou un index mais également un header et un footer.



FONCTIONNEMENT DES FORMULAIRES HTML

Pour réaliser un formulaire, il faut utiliser une suite de balises HTML pour permettre à l'utilisateur de renseigner des informations. Comme pour une liste ou un tableau, un formulaire est englobé dans un conteneur dans lequel sont associées des balises `<label>` et `<input>`.

`<form ... >`

Ouverture du formulaire, la méthode et l'action font référence à la programmation PHP du formulaire

`<form method="post" action="connexion.php">`

`<label ... >`

Les labels sont associés aux inputs, la valeur de `for="..."` doit être égale au `id="..."` du input.

`<label for="lePseudo">Votre pseudo</label> :`
`<input type="text" name="pseudo" id="lePseudo" />`

`<input ... >`

Les inputs permettent à l'utilisateur de renseigner ces informations et peuvent être de différents types.

`<label for="lePass">Votre mot de passe :</label>`
`<input type="password" name="pass" id="lePass" />`

`<input type="submit" ... >`

Ce input est nécessaire pour valider un formulaire.

`<input type="submit" value="Envoyer" />`

`</form>`

Fermeture du formulaire, tous les inputs doivent être placés entre les balises `<form>...</form>`.

`</form>`