

Quelques dates clés à retenir

- **1965** : Invention du langage *hypertexte* (Ted Nelson) qui permet de connecter entre eux divers documents.

Quelques dates clés à retenir

- **1965** : Invention du langage *hypertexte* (Ted Nelson) qui permet de connecter entre eux divers documents.
- **1989** : Naissance officiel du web crée par Tim Berners-Lee au CERN de Genève.

Quelques dates clés à retenir

- **1965** : Invention du langage *hypertexte* (Ted Nelson) qui permet de connecter entre eux divers documents.
- **1989** : Naissance officiel du web crée par Tim Berners-Lee au CERN de Genève.
- **1993** : Ouverture du code source des technologies du web ce qui permet sans doute son développement rapide.

Quelques dates clés à retenir

- **1965** : Invention du langage *hypertexte* (Ted Nelson) qui permet de connecter entre eux divers documents.
- **1989** : Naissance officiel du web crée par Tim Berners-Lee au CERN de Genève.
- **1993** : Ouverture du code source des technologies du web ce qui permet sans doute son développement rapide.
- **1994** : Création du W3C, organisme en charge de la standardisation des langages et technologies du web.

Quelques dates clés à retenir

- **1965** : Invention du langage *hypertexte* (Ted Nelson) qui permet de connecter entre eux divers documents.
- **1989** : Naissance officiel du web crée par Tim Berners-Lee au CERN de Genève.
- **1993** : Ouverture du code source des technologies du web ce qui permet sans doute son développement rapide.
- **1994** : Création du W3C, organisme en charge de la standardisation des langages et technologies du web.
- **1995** : Web interactif avec des technologies publiques. Javascript côté client et PHP côté serveur.

Quelques dates clés à retenir

- **1965** : Invention du langage *hypertexte* (Ted Nelson) qui permet de connecter entre eux divers documents.
- **1989** : Naissance officiel du web crée par Tim Berners-Lee au CERN de Genève.
- **1993** : Ouverture du code source des technologies du web ce qui permet sans doute son développement rapide.
- **1994** : Création du W3C, organisme en charge de la standardisation des langages et technologies du web.
- **1995** : Web interactif avec des technologies publiques. Javascript côté client et PHP côté serveur.

Quelques dates clés à retenir

- **1965** : Invention du langage *hypertexte* (Ted Nelson) qui permet de connecter entre eux divers documents.
- **1989** : Naissance officiel du web créée par Tim Berners-Lee au CERN de Genève.
- **1993** : Ouverture du code source des technologies du web ce qui permet sans doute son développement rapide.
- **1994** : Création du W3C, organisme en charge de la standardisation des langages et technologies du web.
- **1995** : Web interactif avec des technologies publiques. Javascript côté client et PHP côté serveur.
- **2001** : Standardisation des pages avec un modèle en arbre pour les documents : le DOM.

Quelques dates clés à retenir

- **1965** : Invention du langage *hypertexte* (Ted Nelson) qui permet de connecter entre eux divers documents.
- **1989** : Naissance officiel du web crée par Tim Berners-Lee au CERN de Genève.
- **1993** : Ouverture du code source des technologies du web ce qui permet sans doute son développement rapide.
- **1994** : Création du W3C, organisme en charge de la standardisation des langages et technologies du web.
- **1995** : Web interactif avec des technologies publiques. Javascript côté client et PHP côté serveur.
- **2001** : Standardisation des pages avec un modèle en arbre pour les documents : le DOM.
- **2010** : Développement du *Web* sur mobiles.

Internet et Web

Attention à ne pas confondre le Web et Internet :

Internet et Web

Attention à ne pas confondre le Web et Internet :

- **Internet** est né bien avant le Web et désigne un réseau informatique reliant entre eux les ordinateurs du monde entier.

Internet et Web

Attention à ne pas confondre le Web et Internet :

- **Internet** est né bien avant le Web et désigne un réseau informatique reliant entre eux les ordinateurs du monde entier.
- Divers protocoles de communication et divers services existent sur internet, par exemple : le *mail*, la messagerie instantanée, les reseaux pair à pair et ... le **web**.

Internet et Web

Attention à ne pas confondre le Web et Internet :

- **Internet** est né bien avant le Web et désigne un réseau informatique reliant entre eux les ordinateurs du monde entier.
- Divers protocoles de communication et divers services existent sur internet, par exemple : le *mail*, la messagerie instantanée, les reseaux pair à pair et ... le **web**.
- Le **web** n'est donc qu'un des services d'internet même si c'est le plus largement utilisé.

Fonctionnement du web

Le *Web* fonctionne suivant le **modèle client/serveur** :

Fonctionnement du web

Le *Web* fonctionne suivant le **modèle client/serveur** :

- Le **client** est (par exemple) le **navigateur** présent sur notre ordinateur (*firefox, explorer, ...*)

Fonctionnement du web

Le *Web* fonctionne suivant le **modèle client/serveur** :

- Le **client** est (par exemple) le **navigateur** présent sur notre ordinateur (*firefox, explorer, ...*)
- Le **serveur** est un logiciel installé sur un ordinateur distant (par exemple : *apache, IIS, nginx, ...*)

Fonctionnement du web

Le *Web* fonctionne suivant le **modèle client/serveur** :

- Le **client** est (par exemple) le **navigateur** présent sur notre ordinateur (*firefox, explorer, ...*)
- Le **serveur** est un logiciel installé sur un ordinateur distant (par exemple : *apache, IIS, nginx, ...*)
- Le client et le serveur pour se comprendre doivent « parler la même langue », cette langue est le **protocole HTTP** (pour HyperText Transfer Protocol).

Fonctionnement du web

Le *Web* fonctionne suivant le **modèle client/serveur** :

- Le **client** est (par exemple) le **navigateur** présent sur notre ordinateur (*firefox, explorer, ...*)
- Le **serveur** est un logiciel installé sur un ordinateur distant (par exemple : *apache, IIS, nginx, ...*)
- Le client et le serveur pour se comprendre doivent « parler la même langue », cette langue est le **protocole HTTP** (pour **H**yper**T**ext **T**ransfer **P**rotocol).
- Le client envoie des demandes (on dit des **requêtes**) au serveur, lorsque celles ci sont correctement formulées et que la ressource demandée est disponible le serveur y répond favorablement.

Sécurité des transactions client/serveur

Sécurité des transactions client/serveur

- Le protocole http n'est pas sécurisé.

Sécurité des transactions client/serveur

- Le protocole `http` n'est **pas** sécurisé. Les informations sensibles (mot de passe, numéro de carte bleue, ...) qui circulent entre le client et le serveur peuvent donc être récupérées et lues.

Sécurité des transactions client/serveur

- Le protocole `http` n'est **pas** sécurisé. Les informations sensibles (mot de passe, numéro de carte bleue, ...) qui circulent entre le client et le serveur peuvent donc être récupérées et lues.
- Le protocole `https` est sécurisé, de plus en plus de sites web l'utilisent.

Sécurité des transactions client/serveur

- Le protocole `http` n'est **pas** sécurisé. Les informations sensibles (mot de passe, numéro de carte bleue, ...) qui circulent entre le client et le serveur peuvent donc être récupérées et lues.
- Le protocole `https` est sécurisé, de plus en plus de sites web l'utilisent. Les informations qui circulent sont alors cryptées

Sécurité des transactions client/serveur

- Le protocole `http` n'est **pas** sécurisé. Les informations sensibles (mot de passe, numéro de carte bleue, ...) qui circulent entre le client et le serveur peuvent donc être récupérées et lues.
- Le protocole `https` est sécurisé, de plus en plus de sites web l'utilisent. Les informations qui circulent sont alors cryptées
- Le navigateur affiche normalement une icône en forme de cadenas pour signaler que le site visité est en `https`.

Création de sites web

La mise au point de sites web, utilise trois technologies distinctes :

Création de sites web

La mise au point de sites web, utilise trois technologies distinctes :

- HTML : Hyper Text Markup Language

Création de sites web

La mise au point de sites web, utilise trois technologies distinctes :

- HTML : Hyper Text Markup Language
C'est le langage qui **structure** le document présenté (titres, sous-titres, division en paragraphe, tableaux, images ...)

Création de sites web

La mise au point de sites web, utilise trois technologies distinctes :

- HTML : Hyper Text Markup Language
C'est le langage qui **structure** le document présenté (titres, sous-titres, division en paragraphe, tableaux, images ...)
- CSS : Cascading Style Sheet

Création de sites web

La mise au point de sites web, utilise trois technologies distinctes :

- HTML : Hyper Text Markup Language
C'est le langage qui **structure** le document présenté (titres, sous-titres, division en paragraphe, tableaux, images ...)
- CSS : Cascading Style Sheet
Chargé de l'aspect du document : couleurs, bordures, styles et polices de caractères ...
- Javascript

Création de sites web

La mise au point de sites web, utilise trois technologies distinctes :

- **HTML : Hyper Text Markup Language**
C'est le langage qui **structure** le document présenté (titres, sous-titres, division en paragraphe, tableaux, images ...)
- **CSS : Cascading Style Sheet**
Chargé de l'aspect du document : couleurs, bordures, styles et polices de caractères ...
- **Javascript**
C'est un langage de programmation fonctionnant dans un navigateur et chargé de gérer les interactions avec l'utilisateur de la page

Structure d'une page HTML

```
<!DOCTYPE html>
<html>
<head>
  <meta charset='utf-8'>
  <title>Titre ici</title>
</head>
<body>
  <!-- Ici corps du document -->
</body>
</html>
```

Quelques balises à connaître

- Les balises `<h1> ...</h1>`, `<h2> ...</h2>`, ... `<h6> ... </h6>` permettent de définir jusqu'à 6 niveaux de titres.

Quelques balises à connaître

- Les balises `<h1> ...</h1>`, `<h2> ...</h2>`, ... `<h6> ... </h6>` permettent de définir jusqu'à 6 niveaux de titres.
- Les balises `<div> ...</div>` délimitent une division du document, `<p> ...</p>` un paragraphe.

Quelques balises à connaître

- Les balises `<h1> ...</h1>`, `<h2> ...</h2>`, ... `<h6> ... </h6>` permettent de définir jusqu'à 6 niveaux de titres.
- Les balises `<div> ...</div>` délimitent une division du document, `<p> ...</p>` un paragraphe.
- La balise `<hr>` crée une ligne de séparation horizontale, et `
` permet de passer à la ligne

Quelques balises à connaître

- Les balises `<h1> ...</h1>`, `<h2> ...</h2>`, ... `<h6> ... </h6>` permettent de définir jusqu'à 6 niveaux de titres.
- Les balises `<div> ...</div>` délimitent une division du document, `<p> ...</p>` un paragraphe.
- La balise `<hr>` crée une ligne de séparation horizontale, et `
` permet de passer à la ligne
- La balise ` ...` pour mettre un texte en **gras**, ` ...` pour mettre un texte en *italique*.

Quelques balises à connaître

- Les balises `<h1> ...</h1>`, `<h2> ...</h2>`, ... `<h6> ... </h6>` permettent de définir jusqu'à 6 niveaux de titres.
- Les balises `<div> ...</div>` délimitent une division du document, `<p> ...</p>` un paragraphe.
- La balise `<hr>` crée une ligne de séparation horizontale, et `
` permet de passer à la ligne
- La balise ` ...` pour mettre un texte en **gras**, ` ...` pour mettre un texte en *italique*.
- La balise ` ...` permet d'insérer un lien hypertexte.

Quelques balises à connaître

- Les balises `<h1> ...</h1>`, `<h2> ...</h2>`, ... `<h6> ... </h6>` permettent de définir jusqu'à 6 niveaux de titres.
- Les balises `<div> ...</div>` délimitent une division du document, `<p> ...</p>` un paragraphe.
- La balise `<hr>` crée une ligne de séparation horizontale, et `
` permet de passer à la ligne
- La balise ` ...` pour mettre un texte en **gras**, ` ...` pour mettre un texte en *italique*.
- La balise ` ...` permet d'insérer un lien hypertexte.
- La balise `` permet d'insérer une image.

Définir un style en css

- On définit un style en donnant des paires `<attribut>:<valeur>`

Définir un style en css

- On définit un style en donnant des paires `<attribut>:<valeur>`
Par exemple : `color:red; font-weight:bold` indique que l'attribut `color` prend la valeur `red` et que l'attribut `font-weight` prend la valeur `bold`.

Définir un style en css

- On définit un style en donnant des paires `<attribut>:<valeur>`
Par exemple : `color:red; font-weight:bold` indique que l'attribut `color` prend la valeur `red` et que l'attribut `font-weight` prend la valeur `bold`.
- Attention à bien respecter la syntaxe, un caractère `:` sépare l'attribut de sa valeur. Les paires `<attribut> :<valeur>` se terminent par un caractère `;`.

Définir un style en css

- On définit un style en donnant des paires `<attribut>:<valeur>`
Par exemple : `color:red; font-weight:bold` indique que l'attribut `color` prend la valeur `red` et que l'attribut `font-weight` prend la valeur `bold`.
- Attention à bien respecter la syntaxe, un caractère `:` sépare l'attribut de sa valeur. Les paires `<attribut> :<valeur>` se terminent par un caractère `;`.
- Voir le tableau donné en activité pour les attributs les plus courants ainsi que des valeurs possibles

Appliquer un style en css

- On peut appliquer un style directement à un élément html grâce à l'attribut style de cet élément.

```
1 <h1 style="color : yellow;">Ce titre est jaune</h1>
```

Appliquer un style en css

- On peut appliquer un style directement à un élément html grâce à l'attribut style de cet élément.

```
1 <h1 style="color : yellow;">Ce titre est jaune</h1>
```

Appliquer un style en css

- On peut appliquer un style directement à un élément html grâce à l'attribut style de cet élément.

```
1 <h1 style="color : yellow;">Ce titre est jaune</h1>
```

- On peut définir le style dans l'en-tête du document (entre les balises <style> et </style> ou dans un fichier séparé. Il suffit alors de modifier ce fichier pour changer l'apparence de toutes les pages qui l'utilisent.

Définir un style en css

Pour indiquer les éléments auxquels le style s'applique on utilise les **sélecteurs**. Sans entrer dans les détails, un sélecteur peut notamment être :

Définir un style en css

Pour indiquer les éléments auxquels le style s'applique on utilise les **sélecteurs**. Sans entrer dans les détails, un sélecteur peut notamment être :

- une balise html, par exemple pour avoir les titres de niveau 1 de son document en rouge, on peut taper dans le fichier de style :

```
1 h1 { color : red ; }
```

Le sélecteur est ici **h1**, et toutes les balises `<h1>` seront rouges.

Définir un style en css

Pour indiquer les éléments auxquels le style s'applique on utilise les **sélecteurs**. Sans entrer dans les détails, un sélecteur peut notamment être :

- une balise html, par exemple pour avoir les titres de niveau 1 de son document en rouge, on peut taper dans le fichier de style :

```
1 h1 { color : red ; }
```

Le sélecteur est ici **h1**, et toutes les balises `<h1>` seront rouges.

- un nom quelconque précédé du caractère **.**, par exemple on peut avoir dans le fichier de style :

```
1 .encadre { border : 1pt solid ; }
```

Le sélecteur est ici **.encadre**, et toutes les balises html ayant encadre dans leur attribut `class` apparaitront avec une bordure.

```
1 <p class="encadre"> ce paragraphe sera encadré</p>
```