

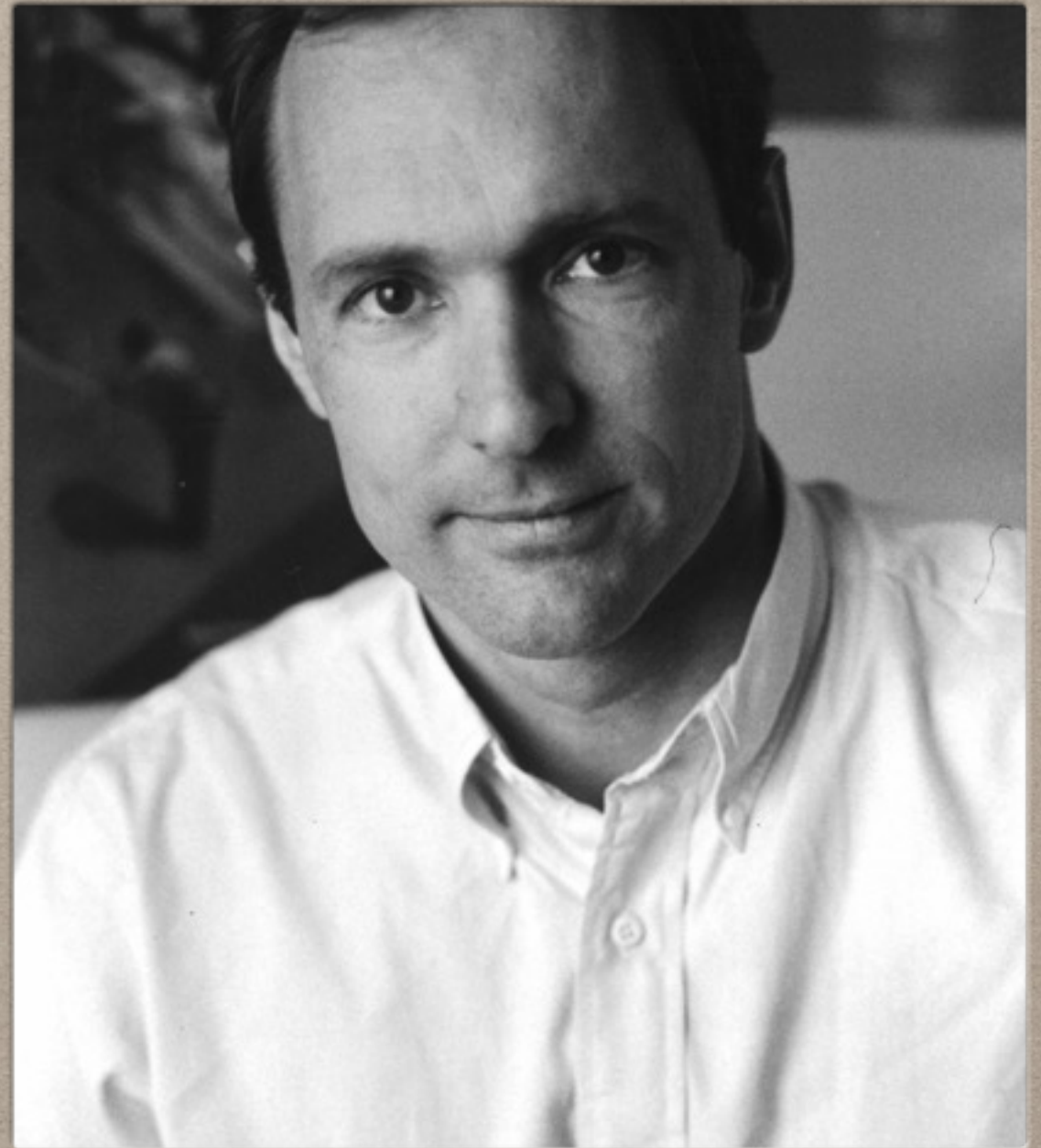


HTML / CSS

Alexis Ducerf - alexis.ducerf@gmail.com

HISTOIRE

- Idée de Tim Berners-Lee en 1989
- Création en 1990
- Basé sur le protocole TCP/IP



HTML ?

```
TYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
</html>
<head>
  <meta name="TITLE" content="Title of the page" />
  <meta name="KEYWORDS" content="Keywords" />
  <meta name="DESCRIPTION" content="Description" />
  <link rel="stylesheet" href="style.css" />
  <script src="script.js" />
</head>
<body bgcolor="white" />
```


HTML

- Hypertext Markup Language
- Standards W3C (HTML5 & HTML4.01)
- Basé sur SGML
- Arrivée de HTML5 ([wikipedia](#))

DOCTYPE

HTML 4.01 Strict

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
```

HTML 4.01 Transitional

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
```

HTML 5

```
<!DOCTYPE html>
```


SQUELETTE

The diagram shows an HTML skeleton with several tags. Arrows point from labels on the right to specific tags in the code on the left:

- Doctype* points to `<!DOCTYPE html>`
- Balise HTML* points to `<html lang="en">`
- Balise Head* points to `<head>`
- Jeu de caractères* points to `<meta charset="UTF-8">`
- Balise title* points to `<title>Document</title>`
- Balise Body* points to `<body>`

```
<!DOCTYPE html>  
<html lang="en">  
  <head>  
    <meta charset="UTF-8">  
    <title>Document</title>  
  </head>  
  <body>  
  
  </body>  
</html>
```

NB : Le <title> est un des éléments les plus importants en SEO !

BALISES ET INDENTATION

Une balise doit toujours se fermer :

```
<html></html>
```

Pour une meilleure lisibilité, on utilisera l'indentation :

```
<head>  
  <meta charset="UTF-8">  
  <title>Document</title>  
</head>
```


LES TITRES

6 niveaux de titres

```
<h1>Titre 1</h1>  
<h2>Titre 2</h2>  
<h3>Titre 3</h3>  
<h4>Titre 4</h4>  
<h5>Titre 5</h5>  
<h6>Titre 6</h6>
```

Titre 1

Titre 2

Titre 3

Titre 4

Titre 5

Titre 6

NB : Il faut essayer de n'avoir que un H1 par page pour le SEO

LES PARAGRAPHES

<p>

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Placeat ex nam, id maxime porro sed quasi amet, esse dolorum cupiditate consequuntur.

**
**

Nemo commodi aspernatur, id eius iure et. Expedita, omnis.

**
**

**
**

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Deserunt ipsum recusandae unde, nam, nemo voluptate distinctio tempore vero quos hic maxime accusamus maiores voluptates placeat eius vitae quasi et animi?

</p>

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Placeat ex nam, id maxime porro sed quasi amet, esse dolorum cupiditate consequuntur.

Nemo commodi aspernatur, id eius iure et. Expedita, omnis.

Lorem ipsum dolor sit amet, consectetur adipisicing elit. Deserunt ipsum recusandae unde, nam, nemo voluptate distinctio tempore vero quos hic maxime accusamus maiores voluptates placeat eius vitae quasi et animi?

Le <p> signifie qu'il y a un paragraphe

Le
 signifie qu'il y a un retour à la ligne

LES LISTES

Les listes à puces

```
<ul>  
  <li>Puce 1</li>  
  <li>Puce 2</li>  
  <li>Puce 3</li>  
</ul>
```

- Puce 1
- Puce 2
- Puce 3

Les listes numérotées

```
<ol>  
  <li>Puce 1</li>  
  <li>Puce 2</li>  
  <li>Puce 3</li>  
</ol>
```

1. Puce 1
2. Puce 2
3. Puce 3

LES LIENS

```
<a href="http://www.monsite.fr" title="Texte au survol" >Mon lien</a>
```

- Le *href* nous donne l'url pour accéder au site
- Le *title* nous donne le titre au survol
- Le texte dans les deux balises sera le texte affiché

```
<a href="#ancre" title="Texte au survol">Mon ancre</a>
```

- Le fait de mettre un # nous envoie directement à l'id donné

```
<a href="http://www.monsite.fr" title="Texte au survol" target="_blank">Mon lien dans une nouvelle page</a>
```

- Le target *_blank* ouvre la page dans un nouvel onglet

LES IMAGES

```

```

- L'attribut *src* indique où trouver l'image
- L'attribut *alt* indique une information alternative si l'image ne s'affiche pas. Elle est utile en référencement car les moteurs de recherche se servent de cette information pour indexer une image
- Les attributs *width* et *height* ne sont pas obligatoires, ils permettent de fixer une taille à l'image. Si on ne spécifie que l'attribut *width*, l'attribut *height* se dimensionnera proportionnellement et inversement
- L'attribut *srcset* permet de fournir une image de taille différente suivant la résolution de l'écran

LES FORMULAIRES

```
<form action="#" method="post">
```

```
</form>
```

Le formulaire est en post, les informations ne seront donc pas visibles dans l'url

```
<form action="valider.html" method="get">
```

```
</form>
```

Le formulaire est en get, les informations seront donc visibles dans l'url. Si aucune méthode n'est précisée, la méthode par défaut est le get

```
<form action="#" method="post" enctype="multipart/form-data">
```

```
</form>
```

Le formulaire contient un type cryptage qui permet l'upload de fichiers

BUTTON

```
<form action="#" method="post">
  <input type="submit" value="Valider">

  <input type="button" value="Ne fait rien">

  <button type="submit">Valider</button>
</form>
```

Il existe 2 façons de faire un bouton, via :

- <button> qui est la méthode HTML5
 - <input>
-
- Le type *submit* permet d'envoyer le formulaire
 - Le type *button* sert essentiellement en javascript

INPUT TEXTE

```
<form action="#" method="post">
  <label for="name">Nom :</label>
  <input type="text" name="name" id="name">

  <input type="text" placeholder="First name" value="Alexis">

  <button type="submit">Valider</button>
</form>
```

Le label permet de lier le texte au input, ainsi le champ input a automatiquement le focus

L'attribut *placeholder* permet de prévenir l'utilisateur de ce qu'il doit saisir dans le champ

L'attribut *value* donne une valeur au champ

INPUT EMAIL

```
<form action="#" method="post">  
  <input type="email" name="email">  
  
  <button type="submit">Valider</button>  
</form>
```

Le champ de type email permet d'avoir un clavier spécial sur smartphone et fait une vérification sur l'email (sur certains navigateurs)

! Veuillez inclure "@" dans l'adresse e-mail. Il manque un symbole "@" dans "fezfezfez".

INPUT PASSWORD

```
<form action="#" method="post">  
  <input type="password" name="password" placeholder="Mot de passe">  
  
  <button type="submit">Valider</button>  
</form>
```

Le champ de type *password* permet de cacher le mot de passe tapé

INPUT CHECKBOX

```
<form action="#" method="post">
  <label>
    <input type="checkbox" name="sport[]" value="Football">Football
  </label>

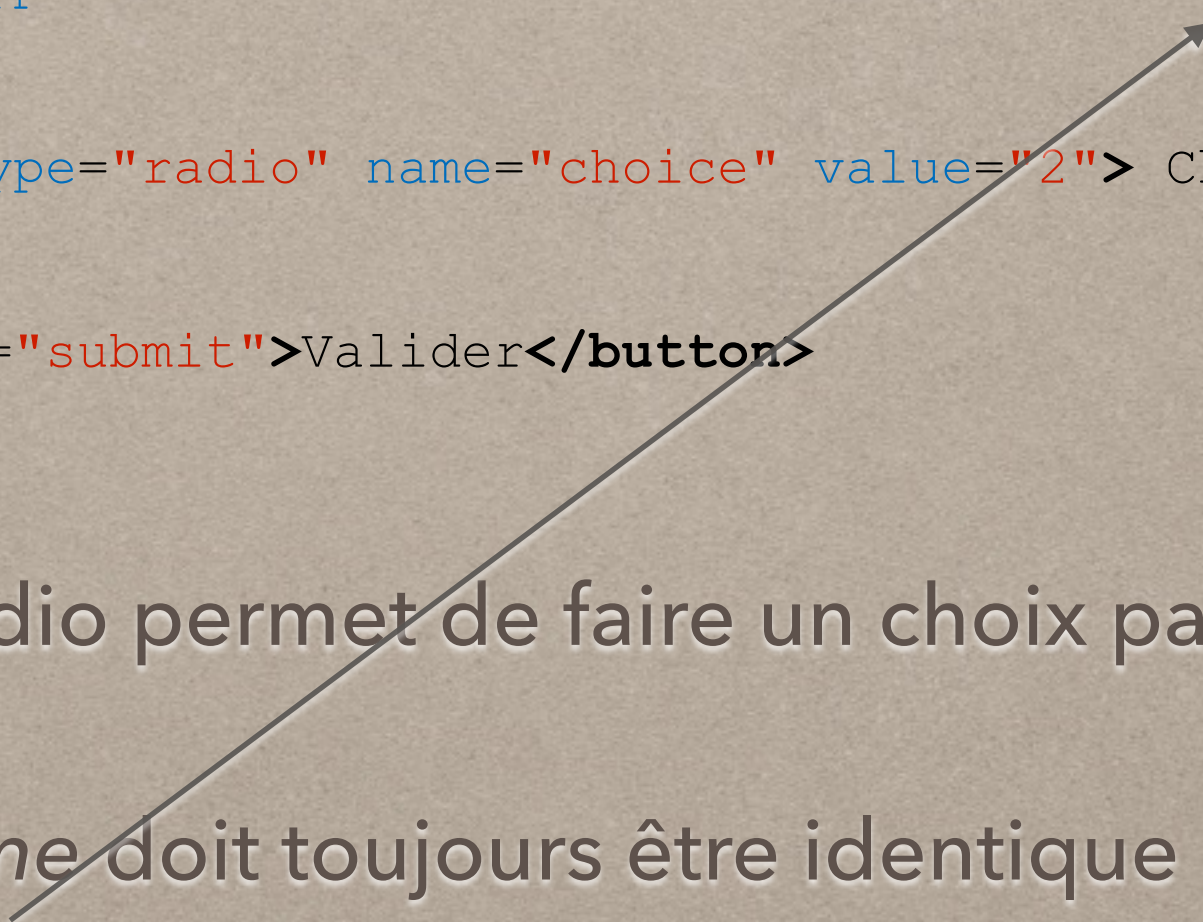
  <label>
    <input type="checkbox" name="sport[]" value="Basketball">Basketball
  </label>
  <button type="submit">Valider</button>
</form>
```

- Le champ de type *checkbox* permet d'avoir des cases à cocher
- Dans l'attribut *name*, il y a *[]* car on veut récupérer un tableau de valeurs

INPUT RADIO

```
<form action="#" method="post">
  <label>
    <input type="radio" name="choice" value="1" checked> Choix 1
  </label>
  <label>
    <input type="radio" name="choice" value="2"> Choix 2
  </label>

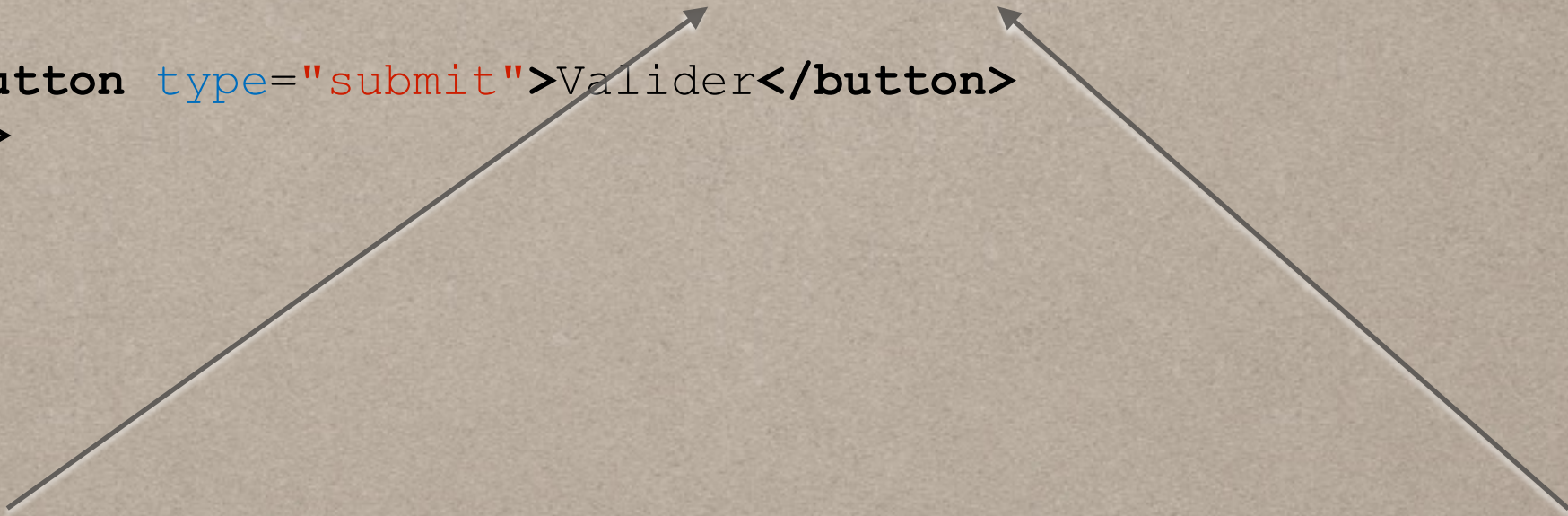
  <button type="submit">Valider</button>
</form>
```



- Le champ radio permet de faire un choix parmi plusieurs choix
- L'attribut *name* doit toujours être identique
- L'attribut *checked* permet de le mettre coché par défaut

TEXTAREA

```
<form action="#" method="post">  
  <textarea name="message" cols="30" rows="10"></textarea>  
  <button type="submit">Valider</button>  
</form>
```


A diagram consisting of two arrows. One arrow originates from the text 'L'attribut cols' and points to the 'cols="30"' attribute in the code. The other arrow originates from the text 'l'attribut rows' and points to the 'rows="10"' attribute in the code.

L'attribut *cols* donne la largeur en colonnes et l'attribut *rows* donne la hauteur en rangées

SELECT

```
<form action="#" method="post">
  <select name="age" >
    <option value="1" selected>1</option>
    <option value="2">2</option>
    <option value="3">3</option>
  </select>

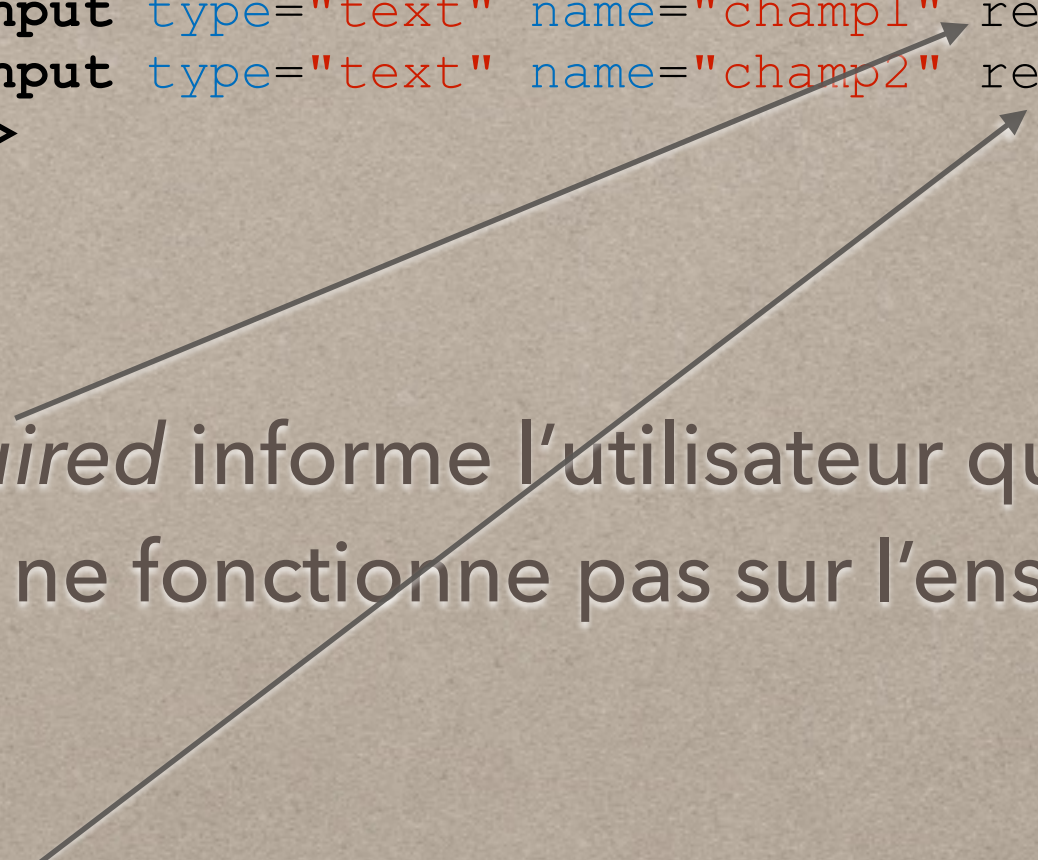
  <button type="submit">Valider</button>
</form>
```



- Le *select* permet de créer une liste déroulante
- Chaque choix est inséré dans une balise *option*
- Si un champ contient l'attribut *selected*, il sera sélectionné par défaut

READONLY / REQUIRED

```
<form action="#" method="post">  
  <input type="text" name="champ1" required>  
  <input type="text" name="champ2" readonly>  
</form>
```



- L'attribut *required* informe l'utilisateur que le champ est obligatoire, il ne fonctionne pas sur l'ensemble des navigateurs
- L'attribut *readonly* rend le champ impossible à modifier

COMPATIBILITÉ

- Normalize.css (à placer dans la balise <head>)

```
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/normalize/3.0.3/normalize.min.css">
```

- HTML5Shiv (à placer en fin de balise <body>)

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/html5shiv/3.7.3/html5shiv.min.js "></script>
```

- Vérifier sur [CanIUse.com](https://caniuse.com)

LIENS UTILES

- <https://openclassrooms.com/courses/apprenez-a-creer-votre-site-web-avec-html5-et-css3>
- <https://www.codecademy.com/fr/tracks/web>
- <http://www.grafikart.fr/tutoriels/html-css>

TP 1

Faire l'architecture de son portfolio en HTML



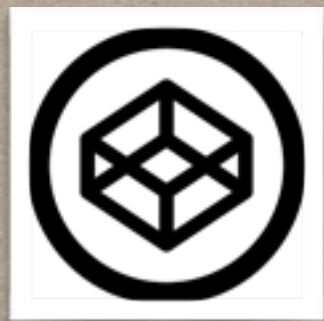
Sublime Text



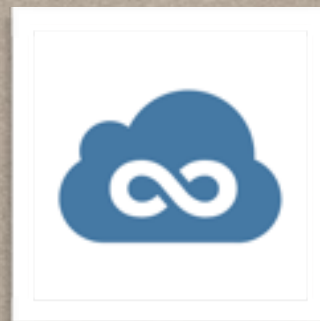
Brackets



Notepad++



Codepen



JSFiddle



JS Bin