

The background is a dark blue gradient. On the right side, there is a complex, swirling pattern of thin, light blue lines that create a sense of depth and movement, resembling a stylized tunnel or a digital landscape. The lines are arranged in a grid-like fashion that curves and twists as it moves towards the right edge of the frame.

CSS

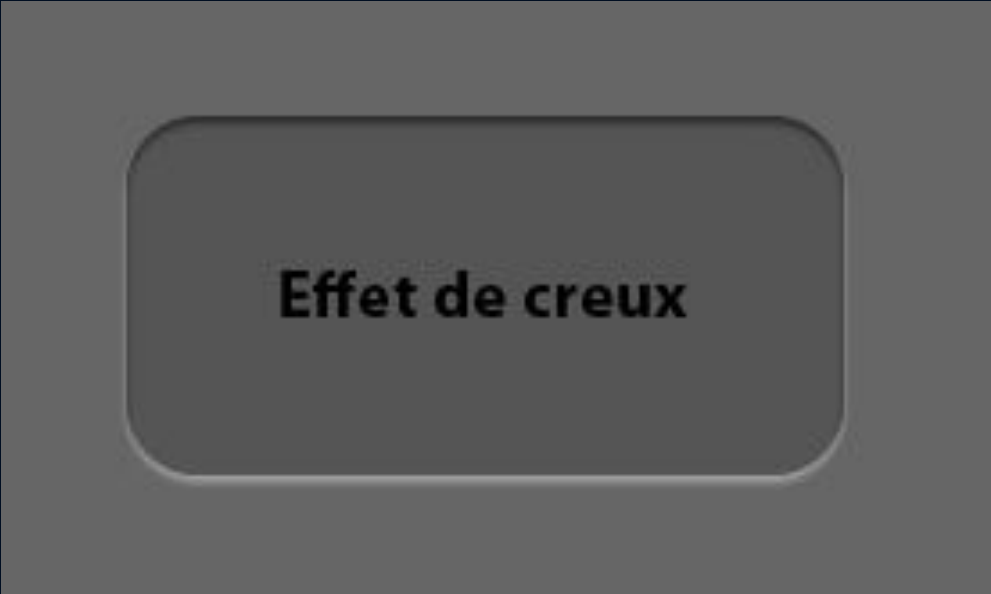
# CSS3 – Compagnon indispensable à HTML5

- Une nouvelle page de l'histoire du web est en train de se tourner. CSS3 est désormais plus prisé que jamais car c'est le compagnon indispensable à la superstar HTML5.
- C'est CSS3 qui fera que votre site ou votre application web sera attrayante ou non.



# Attentes des utilisateurs

- Les attentes des utilisateurs vis-à-vis des interfaces ont nettement évoluées suite aux succès des smartphones et tablettes.
- Ils veulent des interfaces novatrices, attirantes et efficaces.
- CSS3 propose pour cela toute une palette d'effets graphiques à base d'ombre, de dégradés, de bords non rectangulaires, de transformations 2D voire même en 3D !

A dark gray rounded rectangle with a subtle inset effect, where the center is slightly darker than the edges, creating a 3D effect of a button being pressed.

**Effet de creux**

A light gray rounded rectangle with a distinct drop shadow, appearing to float above a darker gray background, demonstrating the CSS3 box-shadow property.

**CSS3 box-shadow**



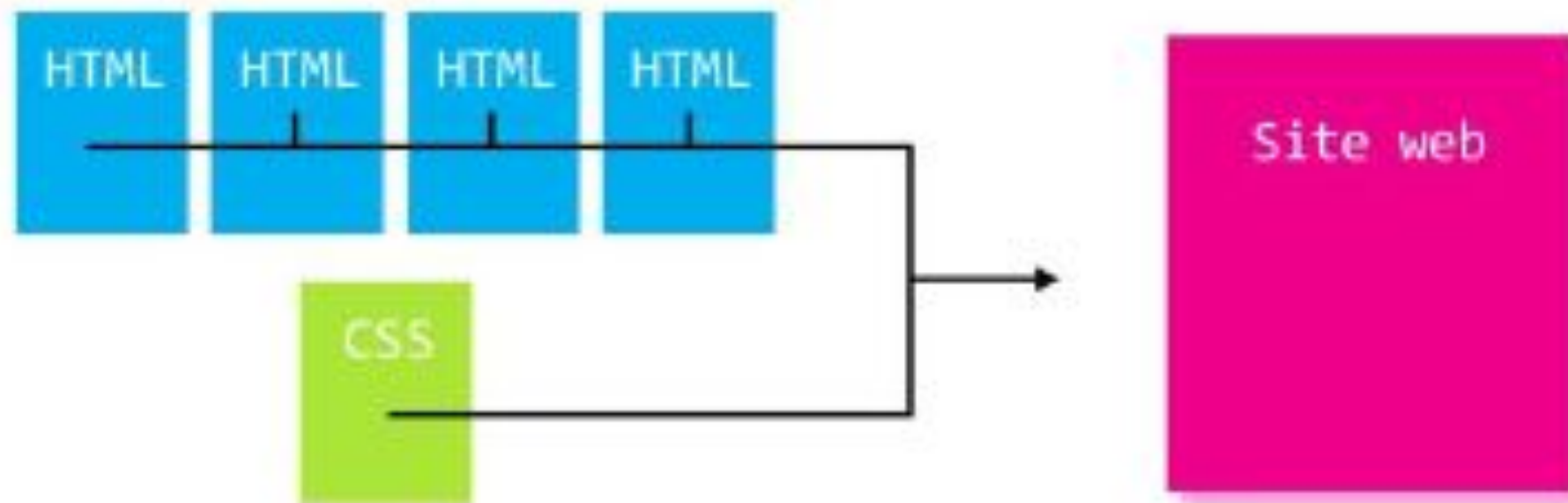
Le HTML permet de décrire les données, c'est le fond. Il n'a pas été créé pour définir le rendu visuel des documents web, mais pour définir la structure du document et organiser le contenu des informations qu'il contient.

CSS modifie l'aspect visuel de ces données, c'est la forme. HTML et CSS sont donc indissociables



# Où écrire le CSS ?

- Vous avez le choix car on peut écrire du code en langage CSS à trois endroits différents :
  - Dans un fichier .css (méthode recommandée)
  - Dans l'en-tête <head> du fichier HTML
  - Directement dans les balises HTML (méthode déconseillée)



**Figure 1.2** — Le principe de la séparation fond/forme

# Intérêts de la séparation multiple

- pouvoir disposer d'une feuille de styles unique, applicable à des documents multiples ;
- pouvoir disposer de feuilles de styles alternatives, en fonction du choix des concepteurs ou des utilisateurs ou en fonction des médias ;
- avoir une meilleure maintenabilité d'un site web (les mises à jour sont plus simples et plus rapides) ;
- • avoir une plus grande flexibilité (une seule modification du CSS modifie l'ensemble des pages d'un site web).

# Appeler la feuille de style

- Pour commencer, il va vous falloir appeler votre feuille de style :
  - Placez-vous sur votre page html, dans la partie <head>
  - Et notez : 

```
<link rel=« stylesheet » href=« style.css » />
```



# Les commentaires

Comme en HTML, il est possible de mettre des commentaires.

```
/* Mon commentaire CSS
```

```
-----
```

```
----- */
```

# Les classes et id

The background of the slide is a deep blue with a complex, abstract pattern. It features a grid of fine, light blue lines that curve and warp, creating a sense of depth and movement, similar to a tunnel or a stylized architectural structure. The lighting is brighter on the right side, where the grid lines are more pronounced and the color transitions to a lighter blue.



# Appliquer un style

- Les attributs « class » et « id » se ressemblent beaucoup
- Ils servent à identifier vos balises
- Vous pouvez les nommer comme vous voulez du moment que le nom commence par une lettre

Fichier .html :

```
<p> Bonjour </p>  
<span class=« rep1 »> Je s'appelle </span>  
<span id=« rep2 »> Groot </span>
```

Fichier .css :

```
p{  
    color: blue; /* les paragraphes en bleu */  
}  
.rep1{  
    color: red; /* les classes « toto » en rouge */  
}  
#rep2 {  
    color: yellow; /* les id « toto » en jaune */  
}
```

# Différences entre class et id

	Class	Id
Appel dans le fichier CSS	.	#
Nombre d'appel dans une page	Plusieurs	Un

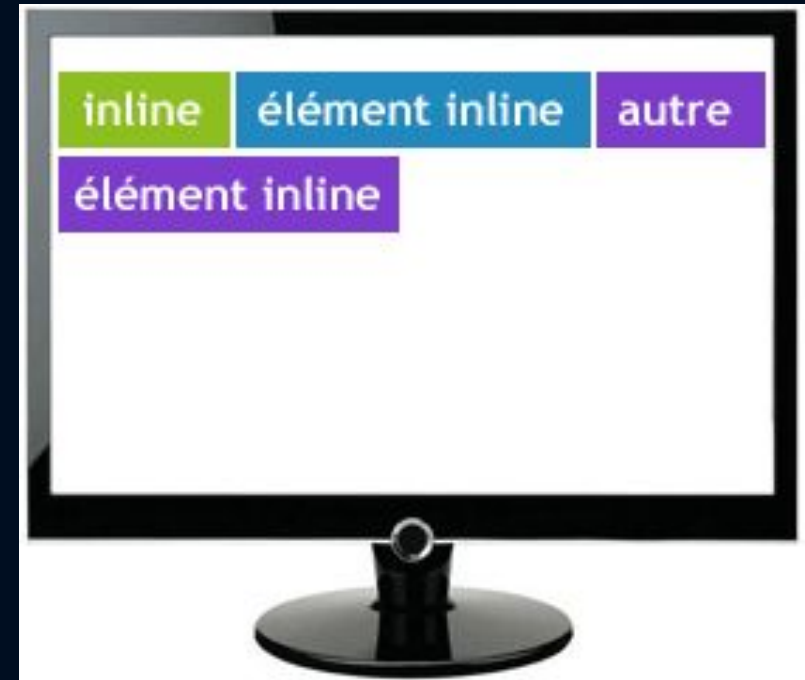


# Les balises universelles

- Ces balises n'ont aucune signification particulière et peuvent être utilisées pour ajouter une class ou un id à un objet sans balise :
  - `<span></span>` : type inline
  - `<div></div>` : type block

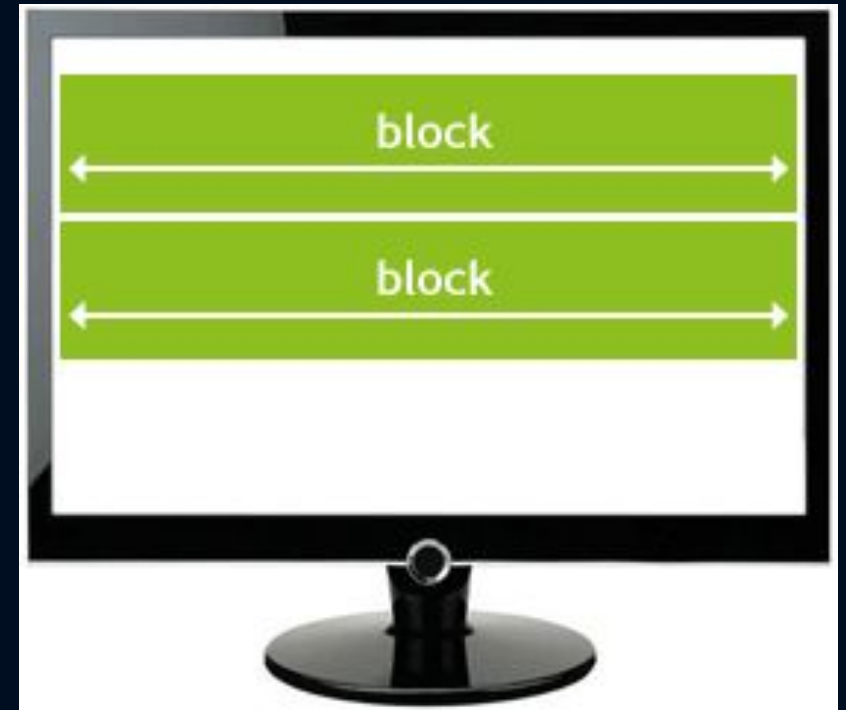
# Balises de type inline

- Les éléments de rendu **inline** se placent toujours l'un à côté de l'autre afin de rester dans le texte
- Par défaut, il n'est pas prévu qu'ils puissent se positionner sur la page (même si cela est possible), ni de leur donner des dimensions (hauteur, largeur, profondeur) : leur taille va être déterminée par le texte ou l'élément qu'ils contiennent. Certaines propriétés de marges peuvent être appliquées aux éléments, comme les marges latérales.



# Balises de type block

- Les éléments de rendu CSS **block** se placent toujours l'un en dessous de l'autre par défaut (comme un retour chariot). Par exemple: une suite de paragraphes `<p>`.
- Par ailleurs, un élément "bloc" occupe automatiquement, par défaut, toute la largeur disponible dans son conteneur et peut être dimensionné à l'aide des propriétés telles que « width », « height », « min-width », etc...





# Les sélecteurs avancés



# Le sélecteur universel

- Il sélectionne toutes les balises sans exception. On l'appelle le sélecteur universel :

```
*  
{  
}
```

# Une balise contenue dans une autre

- Sélectionne toutes les balises `<strong>` à l'intérieur des balises `<p>` :

```
p strong  
{  
  
}
```

```
<p>Bonjour <strong>Toto</strong></p>
```

# Balise avec attribut

- Sélectionne tous les liens <a> ayant un attribut « title » :

```
a[title]
{
}
}
```

```
<a href=« www.monsite.fr » title=« titi »  
>Toto</a>
```



Les couleurs

Indiquer le nom de  
la couleur

Aqua #00FFFF	Black #000000	Blue #0000FF	Fuschia #FF00FF
Gray #808080	Green #008800	Lime #00FF00	Maroon #800000
Navy #000080	Olive #808000	Purple #800080	Red #FF0000
Silver #C0C0C0	Teal #008080	White #FFFFFF	Yellow #FFFF00



# Couleurs du texte

- Hexadécimal : c'est une combinaison de lettres et de chiffres qui indiquent une couleur. On commence toujours par #. Ces lettres fonctionnent 2 par 2. Les 2 premiers indiquent le rouge / puis le vert / et le bleu. #000000 -> noir / #FFFFFF -> blanc. Raccourci possible avec 3 caractères : #FA3 = #FFAA33
- <http://www.colorpicker.com> permet de trouver la valeur hexadécimale d'une couleur très facilement :
- RGB : pour Red / Green / Blue. -> voir sélection couleurs dans paint -  
Exemple : p{ rgb(240,96,204);}

- Vous pouvez aussi trouver des extensions de navigateur qui permettent de "récupérer" n'importe quelle couleur qui vous plaît sur un site web :
- [ColorPicker](#) pour Firefox
- [ColorZilla](#) pour Chrome

# Couleur de fond

- Background-color -> utilisable avec hexadécimal ou RGB

```
body
{
    background-color: black; /* Couleur de fond noir */
    color: white; /* Couleur du texte blanc */
}
```

# La transparence

- Le CSS nous permet de jouer très facilement avec les niveaux de transparence des éléments ! Pour cela, nous allons utiliser des fonctionnalités de CSS3 : la propriété « opacity » et la notation RGBa.
- Avec une valeur de 1, l'élément sera totalement opaque : c'est le comportement par défaut.
- Avec une valeur de 0, l'élément sera totalement transparent.

```
p
{
    opacity: 0.3;
}
```

**ATTENTION:** la propriété opacity s'hérite aux enfants de tous les éléments contenus  
A utiliser avec précaution.

# La notation RGBa

- CSS3 nous propose une autre façon de jouer avec la transparence : la notation RGBa. Il s'agit en fait de la notation RGB que nous avons vue précédemment, mais avec un quatrième paramètre : le niveau de transparence (appelé « canal alpha »). De la même façon que précédemment, avec une valeur de 1, le fond est complètement opaque. Avec une valeur inférieure à 1, il est transparent.

```
p
{
    background-color: rgba(255, 0, 0, 0.5); /* Fond rouge transparence à 50% */
}
```



- Cette notation est connue de tous les navigateurs récents, y compris Internet Explorer (à partir de IE9). Pour les navigateurs plus anciens, il est recommandé d'indiquer la notation RGB classique en plus de RGBa.
- Pour ces navigateurs, le fond ne sera alors pas transparent mais, au moins, il y aura bien une couleur d'arrière-plan.

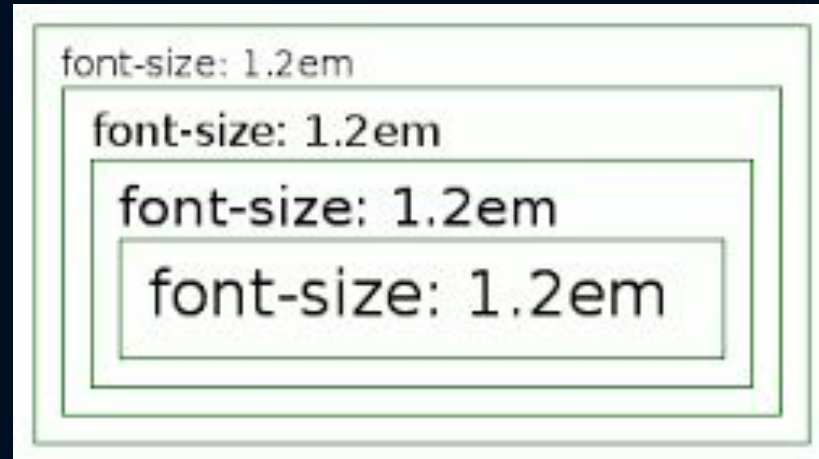
```
p
{
    background-color: rgb(255,0,0); /* Anciens navigateurs */
    background-color: rgba(255,0,0,0.5); /* Navigateurs plus récents */
}
```

The background is a dark blue gradient. On the right side, there is a complex, swirling pattern of thin, light blue lines that form a grid-like structure, resembling a stylized tunnel or a digital wave. The text "Le texte" is positioned on the left side of the image.

Le texte

# Modifier la taille d'un texte

- On utilise la propriété « font-size »
- On peut indiquer la taille en :
  - Pixels, centimètres ou millimètres (taille absolue)
  - En pourcentage : em ou ex (taille relative)
  - Utiliser les mots anglais spécifiques



**xx-small** : minuscule  
**x-small** : très petit  
**small** : petit  
**medium** : moyen  
**large** : grand  
**x-large** : très grand  
**xx-large** : gigantesque

# La police

- La propriété CSS qui permet d'indiquer la police à utiliser est « font-family »
- Il vous faudra préciser plusieurs noms de police, séparés par des virgules

```
balise  
{  
    font-family : police1, police2, police3, etc...;  
}
```





# Google Font

- Vous trouvez le choix de polices limités ?
- Allez faire un tour sur Google Font ;-)



# Personnaliser le texte

- On utilise l'attribut « **font-style** » avec :

**italic** : italique  
**normal** : par défaut

- On utilise l'attribut « **font-weight** » avec :

**bold** : gras  
**normal** : par défaut

- On utilise l'attribut « **text-decoration** » avec :

**underline** : souligné  
**line-through** : barré  
**overline** : ligne au-dessus  
**blink** : clignotant  
**none** : par défaut

# Aligner le texte

- On utilise la propriété « `text-align` » avec :

`left` : à gauche  
`center` : centré  
`right` : à droite  
`justify` : justifié

**ATTENTION** : L'alignement ne fonctionne que sur les balises de type block



# Éléments flottants

The background is a deep blue gradient. On the right side, there is a complex, three-dimensional grid pattern that curves and spirals inward, creating a tunnel-like effect. The grid lines are lighter blue and become more pronounced as they recede into the distance.



# float

- Nous pouvons faire « flotter » un élément autour du texte en utilisant la propriété « float », qui pourra prendre deux valeurs :
  - Left : gauche
  - Right : droite
- *Clear: both* -> permet de stopper le flottement (à écrire dans le css de l'élément qui suit l'entourage du flottant)





Arrière-plan

# Images de fond

- N'oubliez pas que l'image de fond ne s'applique pas forcément à la page entière. On peut aussi mettre une image de fond derrière les titres, paragraphes, etc...

```
body
{
    background-image: url(«plage.jpg»);
}
```

# Options pour les images

- Background-attachment : permet de fixer le fond. Le texte semble alors glisser par-dessus.
- Deux valeurs sont disponibles :
  - **fixed** -> l'image de fond reste fixe
  - **scroll** -> l'image de fond défile avec le texte (par défaut)

```
body
{
    background-image: url(«plage.jpg»);
    background-attachment: fixed; /* Arrière-plan fixe */
}
```



# Background-repeat : répétition du fond

- no-repeat : le fond ne sera pas répété
- repeat-x : le fond est répété horizontalement
- repeat-y : le fond est répété verticalement
- repeat : le fond est répété en mosaïque (par défaut)

```
body
{
    background-image: url(«plage.jpg»);
    background-attachment: no-repeat; /* Arrière-plan sans répétition */
}
```

# Background-position : position du fond

- On peut indiquer où doit se trouver l'image de fond avec background-position. Cette propriété n'est intéressante que si elle est combinée avec background-repeat: no-repeat;
- Vous devez donner deux valeurs en pixels pour indiquer la position du fond par rapport au coin supérieur gauche de la page (ou du paragraphe, si vous appliquez le fond à un paragraphe). Ainsi, si vous tapez :

```
background-position : 50px 150px;
```
- 
- Votre fond sera placé à 50 pixels de la gauche et à 150 pixels du haut.

- Il est aussi possible d'utiliser ces valeurs en anglais :

- `top` : en haut ;
- `bottom` : en bas ;
- `left` : à gauche ;
- `center` : centré ;
- `right` : à droite.

- Il est possible de combiner ces mots. Par exemple, pour aligner une image en haut à droite, vous taperez :

**`background-position : top right;`**

# DEFI

- Trouvez une petite image, une icone ou un logo
- Placez cette image en bas à droite de votre page internet
- Cette image doit toujours être visible



- Si vous utilisez beaucoup de propriétés en rapport avec le fond (comme c'est le cas sur ce dernier exemple), vous pouvez utiliser une sorte de « super-propriété » appelée « background » dont la valeur peut combiner plusieurs des propriétés vues précédemment :
  - background-image
  - background-repeat
  - background-attachment
  - background-position
- On peut donc tout simplement écrire :

```
background: url(«plage.jpg») fixed no-repeat bottom right;
```

- C'est la première « super-propriété » que nous voyons, il y en aura d'autres.
- Il faut savoir que :
  - L'ordre des valeurs n'a pas d'importance. Vous pouvez combiner les valeurs dans n'importe quel ordre.
  - Vous n'êtes pas obligés de mettre toutes les valeurs.

# Plusieurs images de fond

- Depuis CSS3, il est possible de donner plusieurs images de fond à un élément. Pour cela, il suffit de séparer les déclarations par une virgule, comme ceci :

```
body
{
    background: url(«plage.jpg») fixed no-repeat bottom right, url(«palmier.jpg») fixed;
}
```

- La première image de cette liste sera placée par-dessus les autres (figure suivante). Attention donc, l'ordre de déclaration des images a son importance!

The background is a deep blue gradient. On the right side, a series of concentric, curved lines form a tunnel-like perspective, leading towards a bright light source. These lines are composed of a fine grid of lighter blue and white dots or lines, creating a sense of depth and movement. The overall effect is futuristic and dynamic.

# Bordures

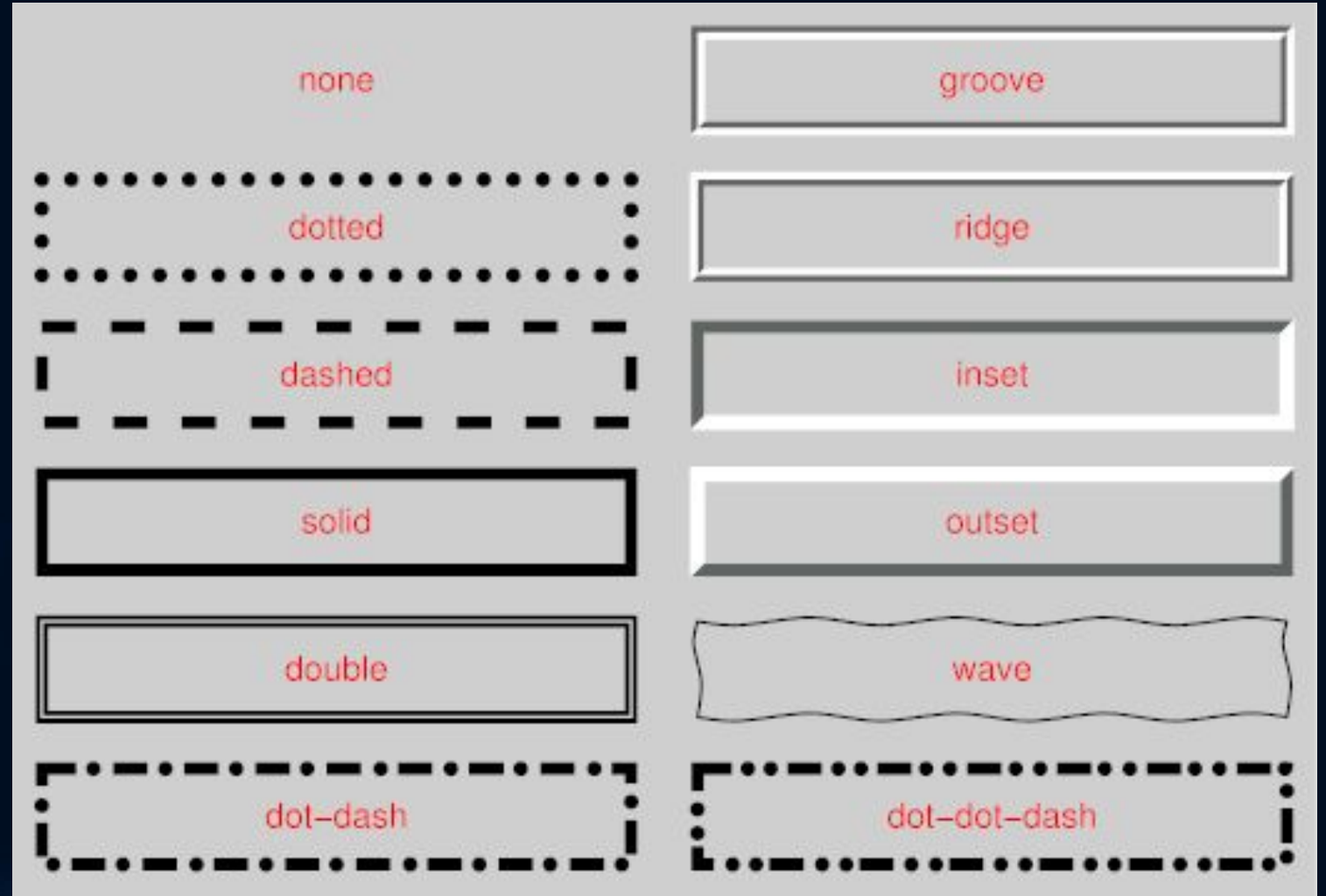


# Bordures standard

La super-propriété border va vous permettre de gérer :

- La largeur : valeur en px
- La couleur : nom de la couleur, valeur hexadécimale ou valeur RGB
- Le type de bordure : type selon les valeurs proposées

**border : 5px red ridge;**



- Si vous voulez mettre des bordures différentes en fonction du côté (haut, bas, gauche ou droite), vous pouvez le faire sans problème. Dans ce cas, vous devrez utiliser ces quatre propriétés :

- `border-top` : bordure du haut ;
- `border-bottom` : bordure du bas ;
- `border-left` : bordure de gauche ;
- `border-right` : bordure de droite.

**`border-top : 3px blue ridge;`**

- Vous pouvez modifier les bordure de n'importe quel type d'élément. Essayez de le faire sur des paragraphes, des images, des titres, des caractères, etc...

# Bordures arrondies

- La propriété « border-radius » va nous permettre d'arrondir facilement les angles de n'importe quel élément. Il suffit d'indiquer l'arrondi en pixels :

```
border-radius : 5px;
```

- Pour préciser la forme de l'arrondi pour chaque coin, utilisez :

```
border-radius : 5px 3px 5px  
3px;
```

Les valeurs correspondent, dans cet ordre :

- En haut à gauche
- En haut à droite
- En bas à droite
- En bas à gauche

# Bordures elliptiques

- Il est possible d'affiner l'arrondi de nos angles en créant des courbes elliptiques.
- Il faut indiquer deux valeurs séparées par *slash* :

```
border-radius : 5px / 10px;
```



The background is a dark blue gradient. On the right side, there is a complex, swirling pattern of thin, light blue lines that form a grid-like structure, resembling a stylized tunnel or a digital wave. The lines curve and twist, creating a sense of depth and movement.

Les ombres

# Les ombres

- Box-shadow : les ombres des boites
- S'applique à tout le bloc et prend quatre valeurs dans l'ordre suivant :
  - le décalage horizontal de l'ombre ;
  - le décalage vertical de l'ombre ;
  - l'adoucissement du dégradé ;
  - La couleur de l'ombre.
- Par exemple, pour une ombre noire de 10 pixels, sans adoucissement, on écrira :

```
box-shadow: 10px 10px 0px black;
```

**N'hésitez pas à faire des tests ! 😊**

# Attribut : inset

- On peut aussi rajouter une cinquième valeur facultative : « inset »
- Dans ce cas, l'ombre sera placée à l'intérieur du bloc, pour donner un effet enfoncé :

```
box-shadow: 10px 10px 0px black inset;
```

# Text-shadow : l'ombre du texte

- Ajoutez une ombre à vos textes avec text-shadow
- Les valeurs sont les mêmes qu'avec box-shadow

```
text-shadow: 2px 3px 2px #000;
```

X Offset    Y Offset    Blur    Color



**Text Shadow**

```
text-shadow: 10px 10px 0px black;
```



The background is a deep blue with a subtle grid pattern. On the right side, the grid lines curve and converge, creating a sense of depth and movement, resembling a tunnel or a dynamic flow. The text is centered horizontally and positioned in the lower half of the image.

Apparence dynamique

# Au survol :hover

- Quand la souris se trouve sur l'élément (lien, image, paragraphe, etc...)

```
a:hover
{
    text-decoration: none;
    color: yellow;
}
```

# Lors du clic :active

- Quand la souris se trouve sur le lien

```
a:active
{
    text-decoration: underline;
}
```

Lorsque le lien a déjà été visité :visited

```
a:visited  
{  
    color:blue;  
}
```





# Dégradation gracieuse et Amélioration progressive

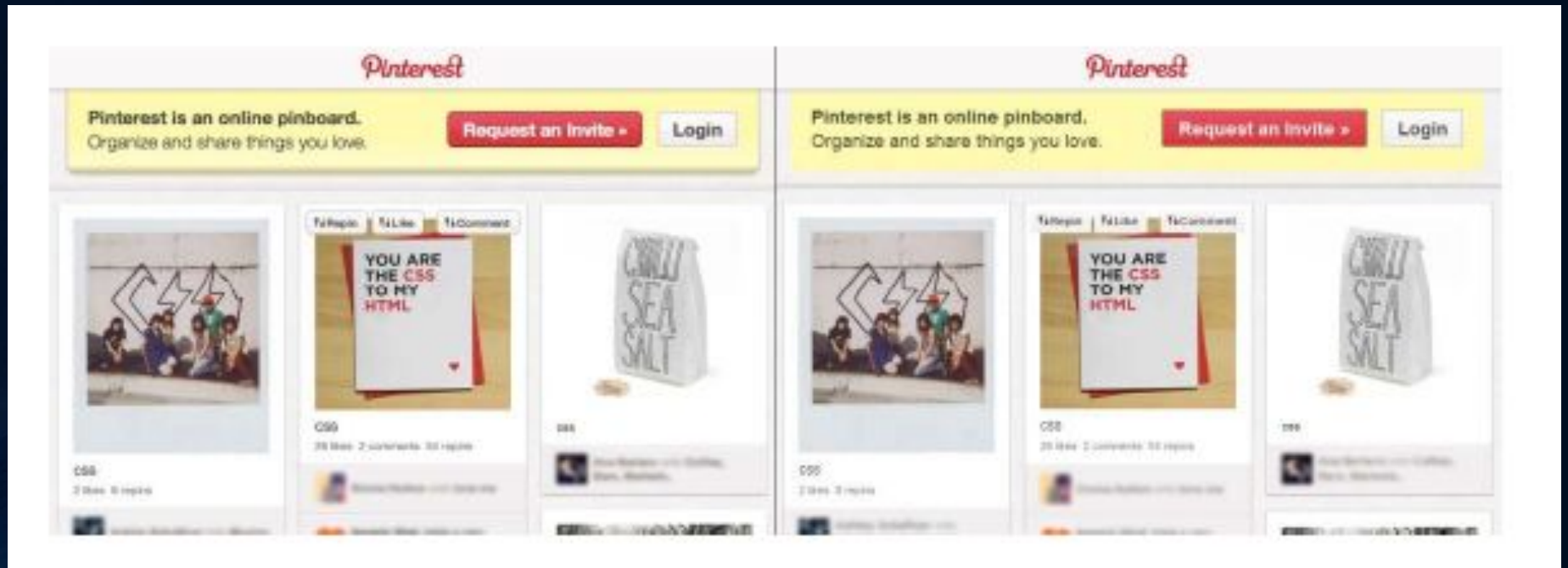
# Dégradation gracieuse

- Un site doit pouvoir être visualisé et utilisé quel que soit le support du navigateur
- Le site doit continuer à fonctionner même si les techniques employées ne sont pas reconnues dans un navigateur ayant été développé bien avant l'apparition de celles-ci
- Il n'est pas indispensable que votre site ait rigoureusement le même aspect sur l'ensemble des navigateurs. C'est tout simplement impossible



# Amélioration progressive

- Le principe est de :
  - créer un site qui soit compatible avec l'ensemble des navigateurs
  - ensuite apporter une plus-value pour les navigateurs compatibles





# Les préfixes vendors



# Utilisation des préfixes

Afin de permettre le développement progressif et l'implémentation des modules CSS, les éditeurs de navigateurs ont la possibilité d'introduire le support de propriétés spécifiques.

Chaque moteur de navigateur utilise alors propre préfixe afin de limiter les interférences entre eux. Voici les différents préfixes connus :

- **-ms-** pour Microsoft (Internet Explorer, IE Mobile) ;
- **-moz-** pour les produits Mozilla (Firefox, Firefox Mobile) ;
- **-webkit-** pour Chrome, Safari, leurs versions mobiles et en règle générale tous les navigateurs basés sur le moteur Webkit ;
- **-o-** pour les navigateurs Opéra (Opéra, Opéra Mobile, Opéra Mini) ;

# Le principe

Ce principe permet aux navigateurs d'ajouter le support de fonctionnalités particulières sans attendre la publication définitive dans la norme, notamment CSS3, et de tester en situation réelle le rendu, souvent réclamé avec insistance par les développeurs web ou suggéré par l'équipe même de l'éditeur. Le but est aussi de permettre la migration vers la syntaxe finale en retirant uniquement le préfixe lors de la validation officielle de la propriété et des valeurs qu'elle peut accepter.

## Préfixe des propriétés CSS des différents navigateurs

- Les nouvelles propriétés CSS3 mettent du temps à être normalisées.
- Grâce aux « **préfixes** », les navigateurs peuvent supporter des fonctionnalités CSS particulières sans attendre la publication définitive d'une nouvelle version du standard.
- Elles se distinguent par un préfixe propre à chaque navigateur :

- **-webkit-** Chrome, Safari, Android...
- **-khtml-** Konqueror
- **-moz-** Mozilla
- **-ms-** Internet Explorer
- **-o-** Opera



# Exemple d'utilisation des préfixes

```
div{  
  -webkit-transform: rotateX(45deg);  
  -moz-transform: rotateX(45deg);  
  -ms-transform: rotateX(45deg);  
  -o-transform: rotateX(45deg);  
  transform: rotateX(45deg);  
}
```

L'idéal est de toujours commencer une déclaration par les propriétés précitées, puis de la finir par celle qui devra les remplacer un jour, pour garantir que l'implémentation finale sera bien assurée par cette dernière (le navigateur devant alors faire le choix d'ignorer les déclarations propriétaires initiales

L'une des étapes pour profiter des fabuleuses propriétés CSS3 sur un site web est d'utiliser les différentes versions préfixées de ces propriétés, car la majorité des modules n'ont pas encore atteint le statut de recommandation. Cela nous permet d'offrir un support au sein de tous les navigateurs qui implémentent ces propriétés « brouillons ».

Pour ce faire, nous nous devons d'écrire la même propriété CSS3 plusieurs fois. Par exemple, dans le cas d'une transformation d'un élément dans l'espace, la propriété transform doit être écrite cinq fois.



# Styles alternatifs



# Règle CSS de la dernière ligne

- Pour les navigateurs qui ne supportent pas les fonctionnalités définies dans les spécifications CSS3, il nous faut quelques fois trouver des astuces pour maintenir une dégradation gracieuse. Il existe pour cela plusieurs méthodes spécifiques à chaque fonctionnalité que nous souhaitons reproduire. Cela passe principalement par des règles CSS différentes ou conditionnelles.
- Lors de l'utilisation, nous devons alors prévoir une deuxième déclaration identique avec une valeur de couleur reconnue par l'ensemble des navigateurs. Cette déclaration doit être ajoutée avant la notation CSS3, comme ceci :

```
p
{
    background-color: rgb(255,0,0);
    background-color: rgba(255,0,0,0.5); /* Les anciens navigateurs ignoreront simplement cette ligne */
    /* Les navigateurs récents conserveront la ligne lue en dernier */
}
```

# Structure conditionnelle

- Concernant les anciennes versions d'Internet Explorer qui posent problème en CSS, la méthodologie peut quelques fois être différente. En effet, certains styles doivent être appliqués exclusivement à certaines versions du navigateur. Nous utilisons alors les feuilles de styles ou les classes conditionnelles, *via* les commentaires conditionnels. Ce mécanisme, uniquement disponible sur le navigateur Internet Explorer jusqu'en version 9, permet de limiter l'interprétation d'une partie du code HTML à ces navigateurs.

```
<!--[if IE]>
```

```
<p>Code HTML exclusif à IE</p>
```

```
<![endif]-->
```

- Afin d'être encore plus précis dans le ciblage, les mots-clés suivants sont reconnus:

- **lt** (*less than* : moins que),
- **gt** (*greater than* : plus que),
- **lte** (*less than equal* : moins ou égal à),
- **gte** (*greater than equal* : plus ou égal à)
- **&** (et logique),
- **|** (ou logique)
- **!** (négation logique).

```
<!--[if IE 8]> ... Code HTML exclusif à IE 8 <![endif]-->
```

```
<!--[if lt IE 8]> ... Code HTML exclusif à IE 7 et inférieurs <![endif]-->
```

```
<!--[if lte IE 8]> ... Code HTML exclusif à IE 8 et inférieurs <![endif]-->
```

```
<!--[if gte IE 8]> ... Code HTML exclusif à IE 8 et supérieur <![endif]-->
```

```
<!--[if !(IE 8)]> ... Code HTML exclusif à IE sauf IE 8 <![endif]-->
```

Grâce à ces commentaires conditionnels, nous allons pouvoir charger une feuille de style conditionnelle, uniquement pour les navigateurs qui demandent des correctifs, comme ceci :

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

<!--[if lte IE 7]>

<link rel="stylesheet" type="text/css" href="styleIE.css"
media="screen" />

<![endif]-->
```

Dans ce cas, la feuille de styles est chargée pour les versions d'IE égale ou inférieure à 7. Cette feuille de styles est généralement appelée après la feuille de styles standard, afin de pouvoir surclasser les règles.



# Notions d'héritage

The background is a deep blue gradient. On the right side, there is a complex, three-dimensional grid pattern that curves and spirals inward, creating a sense of depth and movement, resembling a tunnel or a futuristic architectural element. The grid lines are lighter blue and become more pronounced as they recede into the distance.

# Le CSS et l'héritage

- En CSS, si vous appliquez un style à une balise, toutes les balises qui se trouvent à l'intérieur prendront le même style.
- Si j'applique une couleur de fond noire et une couleur de texte blanche à la balise `<body>`, tous mes titres et paragraphes auront eux aussi un arrière-plan de couleur noire et un texte de couleur blanche...
- C'est ce phénomène qu'on appelle l'héritage : on dit que les balises qui se trouvent à l'intérieur d'une autre balise « héritent » de ses propriétés.

```
body
{
    background-color: black; /* Couleur de fond noir */
    color: white; /* Couleur du texte blanc */
}
p
{
    color: red; /* la couleur prendra le dessus sur celle du body */
}
```

- C'est la propriété CSS de l'élément le plus précis qui a la priorité
- Le même principe vaut pour toutes les balises HTML et toutes les propriétés CSS !
- Le CSS décide que c'est la déclaration la plus précise qui l'emporte