

Pt12 – Botons plans

En aquesta pràctica durem a terme, la implementació de botons plans incorporant icones i fonts externes que ens donaran molta més usabilitat i millora el disseny visual de les nostres webs.

Partirem de la premissa que ja hem preparat o importat prèviament la nostra web amb les icones importades que ens interessin, i només restarà treballar amb el html i el css.

.

L'objectiu de OOCSS, és optimitzar el nostre codi per reduir-lo i poder afegir més codi de forma ràpida.

Oblidem-nos dels id:

Es començarà per deixar d'escriure id's per utilitzar-los com a selectors CSS, els id els reservarem per l'ús de formularis; ara es faran servir classes, ja que aporten més avantatges enfront dels id.

Com per exemple que les classes es poden repetir en diferents elements, els id's no; a més són més fàcils d'accedir, mentre que els id són molt específics i no solen permetre reutilitzar codi, a més no passa res de greu pel fet d'escriure una classe i usar-la una sola vegada com si fos un id.

Lluny de pensar que es tracta de un nou llenguatge de programació, més aviat es tracta d'una metodologia de treball, i una major estructuració del codi, però com sempre, ha de ser un procés adaptatiu, i en base a pràctica.

Eliminar propietats en comú:

La idea principal és reduir codi, tant com puguem, i al mateix temps mantenir-lo per facilitar l'escriptura de més codi, per la qual cosa cal eliminar les propietats que hi hai en comú en certs elements similars.

Hem de intentar que cada classe compleixi una funció que volem, això permet reutilitzar codi. És important teneir present que si indiquem dues o més classes a un element i aquestes tenen propietats iguals, el navegador llegirà la propietat duplicada de la ultima classe indicada, degut als pesos i ponderàncies (ordre d'aplicació de les classes).

Tot es basa en separar l'estructura de l'aparença; separar mides, marges i posicions del que són les fonts, colors i fons. Un petit truc per saber si estàs aplicant bé la tècnica, és revisar el codi CSS i si es troba una propietat igual en diferents elements similars, vol dir que no s'està separant correctament tot.

Amb la tècnica OOCSS, els fitxers CSS haurien de contenir menys línies de codi i per tant pesar menys, no passa res si el codi html s'omple de classes, encara que sigui així com es vegi, serà més fàcil de mantenir el codi i saber quins estils corresponen a determinats elements.

OOCSS està pensat per grans projectes, en els que reutilitzar línies de codi, encara que siguin poques suposa una millora.

Procés

L'objectiu, és crear els diferents botons a través de html, en general l'estructura dels "botons" que crearem són etiquetes de tipus <a> (links).

Per fer-ho utilitzarem la tècnica OOCSS (Object Oriented CSS), on la idea principal és seguir els fonaments bàsics de la programació orientada a objectes, i reutilitzar el codi el màxim possible.

També incorporarem icones i fonts externes, procés que ja haurem fet prèviament. Generarem un html de la manera:

Tal com hem dit simularem els botons a través de l'element <a>, i a partir d'aquí l'anirem vestint amb diferents classes de manera que hi anirem afegint estil a través de les classes i així composant diferents classes que incorporarem als elements, farà que aquests vagin adquirint unes característiques que els diferencià els uns dels altres, sempre en funció de les classes que tingui cada element.

1.- Començarem per estructurar un html senzill per definir un boto:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Document</title>
  <link rel="stylesheet" type="text/css" href="css/estil.css">
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <a href="#" class="boto"><span class="icon-home"></span> Inici </a>
</body>
</html>
```

2.- En l'element <a>, hi definirem una classe boto, que contindrà les propietats genèriques de tots els botons, aquestes estaran definides en el CSS en la classe .boto:

```
.boto{
  background: #7f8c8d;
  color:white;
  display: inline-block;
  font-size: 1.25em;
  margin:20px;
  padding: 10px 0px;
  text-align: center;
  width: 200px;
  text-decoration: none;
  box-shadow: 0px 3px 0px #373c3c;
}
```

Amb això haurem definit les propietats genèriques pels nostres botons, facilitant que al fer copy pastes, en el html, i variant la icona de la etiqueta `` i el text del enllaç, definirem diferents botons genèrics, sense haver d'escriure més que una sola línia de codi en el html, donat que els estils ja els agafarà per la classe.

3.- Si ens fixem, veiem que la icona de la font importada, queda massa propera al text, i ens interessa separar-la una mica; com que tenim definida la classe boto, podem introduir una propietat al span, que també s'apliqui de forma genèrica a tota la tipologia de botons:

```
.boto span{
    margin-right: 10px;
    font-size: 1.25em;
}
```

Ara a través de la classe boto, hi tenim definit l'estructura bàsica dels botons genèrics, de manera que ens faci la reutilització de codi, permetent-nos generar tants botons com vulguem, només introduint línies en el html amb la classe corresponent, permetent crear tants botons genèrics com vulguem.

4.- Ara el que pot interessar-nos, és que els botons tinguin diferents colors els uns dels altres, per tant, en aquells elements `<a>` que vulguem variar el color base per altres, introduïrem una nova classe amb un color identificat, anem al html i crearem un nou botó, i li afegirem la nova classe relacionada amb color:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Document</title>
    <link rel="stylesheet" type="text/css" href="css/estil.css">
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <a href="#" class="boto blue"><span class="icon-home"></span> Inici
</a>
    <a href="#" class="boto"><span class="icon-cloud"></span> Inici </a>
</body>
</html>
```

5.- En el CSS, hi definim les propietats associades a aquesta nova classe, i que ens permetrà identificar a través d'una classe propietats úniques per la classe:

- Per un color blau:

```
.boto.blue{
  background-color: #3498db;
  box-shadow: 0px 3px 0px #2980b9;
}
```

- Per un color verd:

```
.boto.verd{
  background-color: #2ecc71;
  box-shadow: 0px 3px 0px #27ae60;
}
```

- Per altres colors, seguirem el mateix procés, definint un color més fosc en la vora inferior, respecte al color de fons.

6.- De igual forma que s'ha fet pels colors, es pot fer per les formes, creant diferents classes segons les mides que es vulguin ex: mitja, gran.

Les classes mitja i gran:

Html: Afegirem classes segons forma:

```
<body>
  <a href="#" class="boto blue"><span class="icon-home"> </span>
  Inici </a>
  <a href="#" class="boto verd mitja"><span class="icon-
  cloud"></span> Inici </a>
  <a href="#" class="boto gran "><span class="icon-aircraft"> </span>
  Inici </a>
</body>
```

CSS: Serà on diferenciarem unes de les altres:

```
.boto.mitja{
  width:350px;
}
```

```
.boto.gran{
  width:450px;
}
```

7.- De la mateixa manera pel que podríem dir acabats:

Html:

```
<body>
  <a href="#" class="boto blue"><span class="icon-home"> </span> Inici
  </a>
  <a href="#" class="boto verd mitja"><span class="icon-cloud"> </span>
  Inici </a>
```

```
<a href="#" class="boto gran rodo"><span class="icon-aircraft"></span> Inici </a></body>
```

I en el CSS:

```
.boto.rodo{  
  border-radius: 10px;  
}
```

8.- Finalment, només faltaria afegir el comportament amb la pseudoclasa hover, de manera que les vores d'ombra desapareguin.

```
.boto:hover{  
  box-shadow: 0px 0px 0px;  
  padding-top: 11px;  
}
```

Tasca:

- Intentareu pensar amb algun altre tipus de propietat que podem afegir i implementar-la i afegir-la.
- Tingueu present de afegir aquesta funcionalitat en el vostre projecte de pàgina principal. (definida en una tasca anterior).