<style>

p { Por elemento

color: green;

}

#para1 { Por clase

color: red;

}

.colortext { Por tipo

color: red;

}

h1, p, { Varios a la vez

color: red;

background-color: lightblue;

Hay varias unidades en CSS, y todas se pueden usar en todos los casos, la primera son los pixeles o px; la segunda son los porcentajes, siendo que el 100% es el tamaño completo de la pantalla (estos son los mejores porque son escalables a toda pantalla); la tercera son los EM, los cuales toman el tamaño de un elemento fuente (si mi elemento fuente mide 16 px, entonces 1Em = 16px), además pueden tomar distintas fuentes (si mi elemento fuente mide 10 px y yo hago un elemento que mida 1.2 Em entonces este medira 12 pixeles, después puedo crear otro elemento que tenga como fuente esos 12 pixeles; el elemento fuente es el padre); en cuarto lugar estan los Rem, que funcionan de la misma manera, pero solo tienen un elemento fuente que es general para todos, a este se le llama elemento núcleo

border-style: dotted;

border-width: 4px;

border-top-color: red; Esto solo se aplica al borde de arriba

border: 10px solid green; Se pueden encadenar varias propiedad separándolas con espacios, no importa el orden

outline se puede reemplazar con border y funciona igual, los comandos de abajo son la excepción

border-radius: 25px; Lo hace circular

border-bottom-left-radius: 25px;

border-image: url(border.png) 30 round; Pone una imagen como borde, la recorta a los 30 pixeles y la repite; si fuera 30 stretch, la recorta a 30 pixeles y la extiende para que cubra los 30 pixeles

Padding es el espacio que se deja entre el borde y el texto, se ajusta para poder leer los elementos más facilmente; margen es el espacio entre el borde y otros elementos, se ajusta para diferenciar facilmente distintos elementos; ancho y alto es el tamaño per se del elemento; siempre hay un espacio entre el margen y el borde, a esto se le llama el outline

margin-left: 20px; Específico

margin: 50px; General

margin: 50px 25px; El primer parámetro es de arriba y abajo, el segundo de izquierda y derecha

margin: auto; Lo centra horizontalmente

padding se puede reemplazar con margin y funciona igual

width: 50%;

width: 100px;

text-align: center;

text-decoration: none; Quita cosas como negritas, cursiva, subrayado, etc.

text-transform: capitalize; Cambia el texto, puede poner mayúsculas al inicio, todo en mayúsculas, etc.

text-indent: 20px;

font-family: Verdana;

font-style: italic; Esta es para estilos normal|italic|oblique

font-weight: bold; Esta es para que tan grueso normal|bold|bolder|lighter|number

font-size: 3em;

font: italic 20px Verdana; Encadenable

visibility: hidden; Hace que no se vea el objeto, pero sigue ocupando el espacio

display: none; Hace que no se vea el objeto, pero no ocupa espacio

opacity: 0.4;

text-shadow: 2px 2px; Sombra horizontal y vertical

text-shadow: 2px 2px 5px green; Le da un blur de 5 pixeles y le pone color

text-shadow: 2px 2px 5px green, 0 0 10px red; Agrega una sombra extra

box-shadow: 10px 10px;

text-overflow: ellipsis; Si se sale del espacio se le ponen ...

word-wrap: break-word; Hace que si no cabe en toooooodo el espacio la palabra, la corte hasta donde quedo y escriba lo demás abajo

word-break: break-all; Hace que todo se escriba hasta donde llega

}

div {

position: static; Es el default

position: relative; A partir del lugar donde se supone que debería de estar, se aplica lo que se pide

position: absolute; Toma las coordenadas según en lo que este contenido (si esta en la página principal y no dentro de una tabla, es lo mismo que fixed)

position: fixed; Toma las coordenadas de la pantalla

position: sticky; Toma las coordenadas de la pantalla y te sigue si haces scroll

left: 0px; Position siempre va seguido de estas especificaciones

top: 0px;

z-index: -1; Lo manda al fondo

overflow: scroll; Si se sale del espacio, se pone una barra de scroll

overflow-x: scroll; Si se sale del espacio horizontalmente, se pone una barra de scroll horizontal

overflow: hidden; Si se sale del espacio, se desaparece

transform: translate(100px,200px);

transform: rotate(45deg);

transform: scale(0.5,2); Ancho y alto

transform: skew(20deg, 30deg); Eje de las x y de las y

transform: rotateX(150deg); Igual para Y, Z

transition: width 2s; Hace que aquellas cosas que modifiquen width tarden 2 segundos en hacerlo completamente, se pueden especificar varios en el mismo lugar separándolos con coma

transition-timing-function: ease-in-out; Hace que la transición sea más lenta al inicio y al final

transition-delay: 0.5s;

transition: width 2s ease-in-out 0.5s; Encadenable

animation-name: example; Invoca a la animación creada en keyframes

animation-duration: 2s;

animation-delay: 1s;

animation-iteration-count: infinite; Hace que nunca se acabe, si se pone un número lo hace ese numero de veces

animation-direction: alternate; Hace que la animación se invierta a cada iteración

animation-timing-function: ease-in-out;

}

@keyframes example { Agrega una nueva animación para usarse desde el style, hace que vaya de un color a otro

from {background-color: red;}

to {background-color: blue;}

}

@keyframes example { Pone 5 pasos y especifica que tiene que tener cada paso

0% {background-color: red; left:0px; top:0px;}

25% {background-color: blue; left:0px; top:200px;}

50% {background-color: green; left:200px; top:200px;}

75% {background-color: yellow; left:200px; top:0px;}

100% {background-color: red; left:0px; top:0px;}

}

@font-face { Agrega una nueva fuente para usarse desde internet

font-family: sansation;

src: url(sansation\_light.woff);

}

@font-face { Si esto se agrega además del que ya esta arriba, hace que podamos usar tambien la versión bold de la letra

font-family: sansation;

src: url(sansation\_bold.woff);

font-weight: bold;

}

a[target] { Afecta a aquellos elementos de tipo a, con el atributo target

background-color: lightblue;

}

[target="\_blank"] { Afecta a aquellos elementos con el atributo target según el nombre del target

background-color: lightblue;

}

[title~="red"] { Si el atributo title contiene la palabra red en alguna parte, se aplican las propiedades

border: 5px solid red;

}

[title^="red"] { Si el atributo title empieza con la palabra red, se aplican las propiedades

border: 5px solid red;

}

[title$="flower"] { Si el atributo title acaba con la palabra red, se aplican las propiedades

border: 5px solid red;

}

[title\*="flow"] { Si el atributo title contiene las letras red en alguna parte, se aplican las propiedades

border: 5px solid red;

}

p::first-line { Solo la primera línea de p tiene estas características

color: red;

}

p::first-letter { Solo la primera letra de p tiene estas características

color: red;

}

p::before{ Inserta cosas antes de cada p

content: url(smiley.gif);

}

p::after{ Inserta cosas después de cada p

content: url(smiley.gif);

}

div p { Hace que todos los p que sean descendientes de div tengan estas características

color: red;

}

div > p { Hace que todos los p que sean descendientes inmediatos de div tengan estas características

color: red;

}

div ~ p { Hace que todos los p que sean hermanos de div tengan estas características

color: red;

}

div + p { Hace que el p que sea el hermano más próximo de div tenga estas características (solo el más próximo, el otro ya no; el más próximo hacia adelante)

color: red;

}

strong { Hace que los elementos con la característica de strong tengan su propia línea

display:block;

}

a:link { /\* unvisited link \*/

color: red;

}

a:visited { /\* visited link \*/

color: blue;

}

a:hover { /\* mouse over link \*/

color: black;

}

a:active { /\* selected link \*/

color: green;

}

ol {

list-style-type: upper-roman;

list-style-image: url('sqpurple.gif');

list-style: circle inside url('img\_marker.png'); Pone la imagen y además pone como backup el estilo de círculo por si no llega a cargar

}

li { Hace que los bullets desaparezcan y los pone todos en una sola línea separados por espacios

display: inline;

}

table, th, td { Table es para el borde externo general, th es para el header y td para los componentes

border: 2px solid green;

Se pueden usar todas las propiedades vistas previamente, y se considera lo que hay dentro de la celda como el tamaño completo de la pag; por ejemplo, si yo le agrego margen a td y th, lo hará con respecto a los bordes de table

}

table { Une todos los bordes, si configuro el borde general, pero no el del header, en la parte del header no va a aparecer ninguno (es como una intersección)

border-collapse: collapse;

}

body { El fondo general

background-color: linen; Tambien se puede usar hexadecimal, rgb, rgba, hsl y hsla

background-image: linear-gradient(white, green); Da un gradiente de blanco a verde, también puedo agregar más colores, los colores los puedo especificar como dice arriba también

background-image: linear-gradient(to bottom right, white, green); en lugar de bottom right puedo poner 70deg para que vaya con 70 grados y así, o circle

background-image: radial-gradient(white, green); Lo mismo pero en versión circular

background-image: url("paper.gif");

background-size: 100px 80px; Si pongo cover en lugar de los pixeles, se autoajusta para cubrir toda la página

background-repeat: repeat-y; Repite la background image sobre el eje de las y

background-repeat: no-repeat top right; Se pueden encadenar

background-image: url(img\_flwr.gif), url(paper.gif); Se queda en el fondo la última

background-origin: content-box; Al ponerlo en algún lugar distinto a body (como esta ahorita) hace eque el fondo solo aplique a ese elemento

background-clip: padding-box;

}

div {

background-origin: content-box; Hace que el fondo solo aplique a ese elemento

background-clip: padding-box; Hace que el fondo no salga del padding

}

<link rel="stylesheet" type="text/css" href="mystyle.css"> Plantilla importada

</style>

<body style="background-color: linen"> Se puede no usar el style