**CSS**

**Seletores:**

<style>

p {

//exemplos de atributos

color: red;

border: 1px solid blue //colocar bordas,tamanho, tipo e cor (não importa a ordem)

font-size: 80px; //tamanho da fonte

background-color: gray //cor de fundo

}

</style>

.classe //para classes

#id //para ids

\*//universal

[atributo] //para selecionar os elementos que tenham esse atributo, não importa o valor

[atributo]{

}

[atributo=valor] //para selecionar os elementos que tenham esse atributo, com um valor especificado

[atributo|=val] //seleciona todos os elementos que contenha um valor val (além de ter mais coisas além de val)

[atributo^=val] //seleciona os valores que começam com val

[atributo$=val] //seleciona os elementos que terminem com val

a:hover, a:active, a:focus {

color: darkred;

text-decoration: none;

} // Não selecionam os elementos mas sim certas partes de elementos ou elementos apenas em determinados contextos

**BORDAS**

border-style: dotted; //propriedade especifica o tipo de borda a ser exibido, propriedade pode ter de um a quatro valores (para a borda superior, borda direita, borda inferior e borda esquerda).

border-color: red; //especifica uma cor especifica para cada lateral da borda. Segue a mesma ordem

border-width: 10px; //especifica o tamanho das bordas. Segue as mesmas ordens.

dotted - Define uma borda pontilhada

dashed - Define uma borda tracejada

solid - Define uma borda sólida

double - Define uma fronteira dupla

groove- Define uma borda com sulcos 3D. O efeito depende do valor da cor da borda

ridge- Define uma borda 3D sulcada. O efeito depende do valor da cor da borda

inset- Define uma borda inserida em 3D. O efeito depende do valor da cor da borda

outset- Define uma borda de partida 3D. O efeito depende do valor da cor da borda

none - não define fronteiras

hidden - Define uma borda oculta

**FONTES E CORES**

color: #86864C; //código de uma determinada cor html gerada.

Fontes: família genéricas (um grupo de famílias de fontes com aparência semelhantes) e família de fontes (uma família de fontes específicas).

font-family: "Times New Roman", Times, serif; //tag para especificar a fonte do texto, para formatar. Se não encontrar a primeira fonte, tenta encontrar a segunda, depois a terceira, e assim por adiante.

font-size: 80px; //tamanho da fonte, em pixels, porcentagem ou em.

font-style: italic; /\* normal, oblique \*/ //escolher o estilo da fonte

fonte-weigth: bold; //para deixar o texto em negrito, pode ser por número também para controlar o peso do negrito.

font: ; //Tudo dentro dessa propriedade.

text-decoration: overline; /\* line-through, underline, underline overline \*/ //define decorações de texto;

**BOX MODEL**

Conteúdo (Content) – Preenchimento (padding) – Borda (border) – Margem (margin)

Content - atributos: heigth, width.

Paddin, border (pode ter também style, width, color) e margin – atributos: top, right, bottom e left

display: inline; //block, inline-block. São para a formatação de textos. O primeiro não a quebra, o segundo a quebra, e o terceiro é um hibrido dos dois.

**ELEMENTOS FLUTUANTES**

Float é o posicionamento;

float: rigth; //left,top,bottom

.clear{

clear:both; //limpar um fluxo que já está flutuando, para um elemento não ficar em volta.  
}

**POSICIONAMENTO**

margin: 30px auto; //margem lateral automática, fica centralizado o conteúdo

left: 30px; //top,rigth,bottom -> para funcionar, precisa do position

position: relative; //alinhar sua tag relativa a ela mesma.

position: absolute;//a sua tag se alinha de acordo com a posição absoluta do seu parente mais próximo.

position: fixed// a sua tag se alinha de acordo com a área que você consegue visualizar da sua página.

**FORMATAÇÃO DE LINKS**

a:link{ //para links não visitados

color: red;

}

a:visited{ //para links visitados

color: blue;

}

a:hover{ //para quando passar o cursor do mouse

color: black;

}

a:active{ //quando você clicar e segurar o botão do cursor

color: orange;

}

height: cal(100% - 40px);

**CSS GRID**

display:grid; //criar o grid

grid-template-columns: 50%; //repeat(4, 8.33%) //definir as colunas

grid-template-columns: [inicio]20% [meio1]20% [meio2]20% [fim]; //colunas com nomes

minmax(100px,300px); //função para tamanho máximo e mínimo

grid-template-rows:50%; //define as linhas

grid-column-start:2; //coluna deve começar

grid-column-end:4; //coluna que deve terminar

grid-row: 1 / 5; //inicio fim

grid-area: linhainicio / colunainicio /linhafim /colunafim

grid-column-gap:20px; //distância entre dois blocos

grid-gap: 50px 30px; //linha colunas

grid-auto-rows; //linhas implícitas

grid-auto-columns; //linhas implícitas

GRID POR ÁREA – NOME

.header {

grid-area: hd;

background-color: deepskyblue;

}

.footer {

grid-area: ft;

background-color: coral;

}

.content {

grid-area: main;

background-color: yellow;

}

.sidebar {

grid-area: sd;

background-color: firebrick;

}

.wrapper {

display: grid;

grid-template-columns: repeat(9, 1fr);

grid-auto-rows: minmax(100px, auto);

grid-template-areas:

"hd hd hd hd hd hd hd hd hd"

"sd sd sd sd main main main main main"

"ft ft ft ft ft ft ft ft ft";

}

//Nesse esquema, você cria uma matriz para manipular o seu grid. Para deixar sem nada na matriz, use .(ponto).

grid-template-columns: repeat(auto-fill, minmax(200px, 1fr)); //grid responsivo, onde o auto-fill é o responsável.

// Diferença de auto-fill e auto-fit?