**Auia 1 – Psicologia das Cores – Capítulo 13 do material**

A cor do produto e/ou da embalagem podem influenciar as emoções e impactar na decisão de compra. Isso atua de maneira inconsciente.

Empresas de tecnologia gostam de tons de **azul** 🡪 Competência, sabedoria, calma, confiança, profissionalismo e segurança; além de ser a cor com a menor taxa de rejeição.

O material **PDF** da uma explicação sobre cada cor.

Fundo preto com letra branca pode causar cansaço durante a leitura.

**Auia 2 – Representando as Cores com CSS3 – Capítulo 13 do material**

Você consegue alterar a cor, o tom e a transparência quando configurar uma cor

Representação por nomes – green, White, etc...

Representação por códigos Hexadecimais - #XYZ

Representação por RGB (Red, Green and Blue)

Representação por HSL (Hue, Saturation and Luminosity)

Além disso, ainda pode colocar a transparência depois.

**Auia 3 – Harmonia de Cores com CSS3 – Capítulo 13 do material**

**Círculo cromático** – 12 cores principais, usado para harmonizar cores

Cores **primárias** do círculo: **Amarelo, vermelho e azul** (essas cores formam um triângulo, dentro do círculo)

Cores **secundárias** do círculo: **Laranja, violeta e verde** (essas cores formam um triângulo invertido, dentro do círculo)

Cores **terciárias (também chamadas de tons pastéis)** são mistura das cores primárias e secundárias; primeiro o nome da cor primária e depois o da secundária.

EX: **Amarelo**-**esverdeado; amarelo alaranjado, vermelho alaranjado, vermelho arroxeado, azul arroxeado, azul esverdeado**

Cores Frias: Ao dividir o círculo no meio, seria a parte da esquerda.

Cores Quentes: Ao dividir o círculo no meio, seria a parte da direita.

A paleta de cores possui de **3 a 5 cores**, exceto branco e preto, e parte de uma **cor principal,** você pode basear a cor principal do site semelhante a principal cor da logo do cliente.

**Cores complementares (para contraste):** Escolha a cor e a cor complementar irá se situar no outro extremo do círculo, basta traçar uma reta. (EX: Amarelo com roxo, Vermelho com verde, azul com laranja).

**Cores Análogas (cores “irmãs”)**: Basta pegar as cores que ficam ao lado (EX: Amarelo, verde e laranja)

**Cores Análogas e uma complementar:** Escolha uma primária principal, selecione suas cores análogas e finalize com uma complementar.(EX:Amarelo, verde, laranja e roxo)

**Cores Análogas Relacionadas**: Escolha uma primária principal, escolha a do seu lado, pule uma casa e escolha a do lado da que foi pulada. (EX: Amarelo, laranja e quase vermelho).

**Cores Intercaladas:** Escolhe a cor, pula e escolhe, pula a do lado e escolhe. (EX: Verde, roxo e verde claro)

**Cores Triádicas:** Escolhe a cor, pula 3 do lado e escolhe, pula 3 do lado e escolhe. (EX: Amarelo, vermelho e azul)

**Cores em Quadrado:** Escolhe a cor, pula 2 do lado e escolhe, pula 2 do lado e escolhe, pula 2 do lado e escolhe. (EX: Amarelo, laranja e roxo e verde)

**Cores Tetrádica:** Escolhe 2 cores e escolhe os seus opostos. (EX: Verde, vermelho, laranja e azul).

**Monocromia (gera o efeito degradê):** Escolhe uma cor e modifica sua saturação e luminosidade, gerando um total de 1 cor + 3 variações dela (com a alteração de saturação e luminosidade)

**Auia 4 – Paleta de Cores – Capítulo 13 do material**

[**https://color.adobe.com/pt/create/color-wheel#**](https://color.adobe.com/pt/create/color-wheel) **-** para criação da paleta

Selecione a cor de base e depois o método (baseado acima) para as cores restantes.

Em **extrair tema** – Você consegue fazer o upload de uma imagem e ver a paleta de cores, ai é só voltar em disco de cores para ter acesso ao código hexadecimal de cada cor.

Em **extrair gradiente 🡪** Você consegue criar o degradê baseado em uma imagem

Em exibir – tema de cores – você pode ver as paletas mais usadas

<https://paletton.com/#uid=1000u0kllllaFw0g0qFqFg0w0aF> 🡪 Em exemplo ele te mostra modelo de sites com a paleta escolhida, depois vá me page layout e troque as opções para ter mais ideias.

<https://coolors.co/3b429f-aa7dce-f5d7e3-f4a5ae-a8577e> --> Aqui você pode selecionar e introduzir as cores desejadas, coloque o cadeado e veja quais as outras sugestões

**Aula 7 – Como criar degradê com CSS3 – Capítulo 13 do material**

Abra a tag style na head, faça o body { e usa o parâmetro **background-image: linear gradiente (to right OU to left, cor 1, cor 2, cor 3);**

De cor 1 para cor 2 para cor 3 (pode ter várias cores); após a cor você também pode colocar uma **porcentagem %** OU colocar em graus (EX: 45deg, 90deg, 15deg), assim o gradiente fica na diagonal.

Para usar o gradiente **to top** ou **to bottom**, você deve:

Crie uma seleção global das CSS ( **\* { }** ) dentro da tag style do head

Para deixar essa cor de BG **degradê fixo**, apesar do scrool você deve inserir o parâmetro:

**Na tag que o bg está** 🡪 **background-attachment: fixed**;

Após o **background-image:** , você pode usar o parâmetro **circle**

**Aula 8 – Criando um exemplo real – Capítulo 13 do material**

O código principal da body ele deixa dentro de uma **tag <main>** (ela é semelhante a div, mas ela será a principal).

**No style.css**

\* {

Height: 100%;

Font-family: ;

}

**Margin: auto** 🡪 irá centralizar o conteúdo no meio da tela, independentemente do tamanho dela

**Widht: Zpx** 🡪 Largura

**Padding: Zpx** 🡪 Acolchoamento (engordadinha kkk)

**Border-radius:** **Zpx** 🡪 Da o arredondamento nas pontas

-+

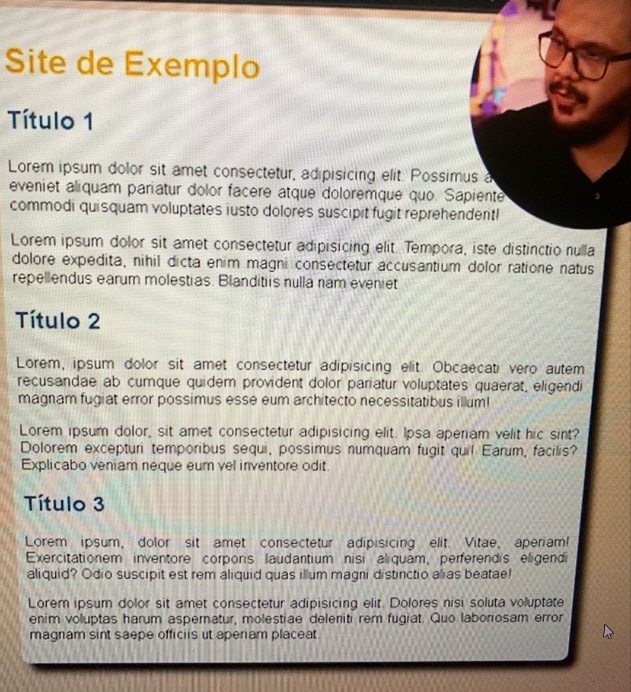
**Box-shadow** 🡪 Adiciona sombra e possui 4 parâmetros: (também existe o **text-shadow**)

1º quanto vai espalhar para o lado

2º quanto vai espalhar para baixo

3º quanto a sombra vai espalhar

4º qual a transparência da sombra



**Aula 1 – Primeiros passos em tipografia – Capítulo 14 do material**

Surgiu quando as pessoas queriam melhorar a visibilidade das letras, na época da prensa.

Também influenciam as mesmas coisas que as cores.

A tipografia tem que casar com o que a palavra vai transmitir.

**Aula 2 – Anatomia do tipo – Capítulo 14 do material**

O inicio da criação de uma nova fonte começa com o **x** minúsculo, a **altura** desse **x** dita o tamanho das outras letras minúsculas.

O objetivo é facilitar a leitura.

**Branco no fundo e preto na letra.**

**Altura do corpo:** Considera todo tamanho das letras maiúsculas e todo o vazamento descendente (para baixo) das letras (g, f, q).

**Glifo ou caracteres:** É uma letra

**Fonte:** Conjunto de glifos

**Categorias de fontes:**

**Serifada**

**Sem serifa (sans=serif)** 🡪 Dê preferência quando for textos muito longos

**Monoespaçada (c/ ou s/ serifa)** 🡪 As letras possuem a mesma largura

**Handwriting** 🡪 Simula a escrita à mão

**Display** 🡪 Fonte sem nenhuma das características acima

**Aula 3 – Família de fontes com CSS – Capítulo 14 do material**

**Font-family:** **‘**fonte 1**’**, fonte 2, fonte 3 (genéricas: sans-serif, serif, monospace)

Nem todos dispositivos aceitam todas as fontes.

Veja as indicações de combinações no site abaixo:

<https://www.w3schools.com/cssref/css_websafe_fonts.php>

**Aula 4 – Tamanho de fonte e suas medidas – Capítulo 14 do material**

Formas de colocar o **font-size:**

**Medidas absolutas:** cm, mm(milímetro), in(polegada), px, pt (ponto), pc (paica). As 2 últimas não são recomendadas

Sua apresentação pode ser diferente, a depender do tamanho da tela do usuário.

**Medidas Relativas:**

**em:** Medida relativa ao tamanho ATUAL da fonte; relativo ao M maiúsculo de uma determinada fonte

**ex:** Tamanho da altura X de uma fonte

**rem:** Tipo **em** só que relativo à fonte configurada no **<body>**

**vw (viewport widht):** Referente à largura da tela do usuário (viewport)

**vh (viewport hight):** Referente à altura da tela do usuário (viewport)

**%**: Porcentagem

O mais usual é o **em ou px**

**NÃO PODE TER ESPAÇO ENTRE O NÚMERO E A MEDIDA**

Geralmente, o tamanho normal da fonte é **16px**, ou seja, **1em = 16px**

A medida dos editores de texto (word) é em pontos (pt)

**Aula 5 – Peso, estilo e shorthand font – Capítulo 14 do material**

**Font-weight:** 🡪 Peso: faz a fonte ficar mais “magra” ou “gorda”

**Tipos:** (em algumas fontes, ela não tem todos os pesos, tipo arial, verdana)

Lighter

Normal

Bold

Bolder

OU de 100 até 900

**Font-style:**

**Tipos:**

Normal

Italic

Você pode combinar o weight com o style.

Nem todas as fontes tem todas essas possibilidades

O sublinhado é **text-decoration: underline**;

**TRANSFORMANDO TUDO EM 1 LINHA SÓ: shorthand font (também funciona para border)**

Sequência:

**Font-style 🡪 font-weight 🡪 font-size 🡪 font-family**

Exemplo:

**font: italic bolder 3em ‘work sans’, sans-serif;**

**Aula 6 – Usando Google Fonts – Capítulo 14 do material**

Exercício 18

Você pode usar o filtro por categoria

Clique em select this style , copie o código entre as tags <style>, e cole logo abaixo da tag <style> do seu código

**Link:** Dentro do HTML

**@import:** Dentro do CSS

No Fonts - Em Setence: Type something, escreva algo e veja como ficaria nas fontes.

Você vai adicionando as variações que essa fonte tem.

**Aula 7 – Usando Fontes Externas Baixadas – Capítulo 14 do material**

<https://www.dafont.com/pt/>

Escolha uma fonte que tem todos os glifos com acentos e maiúscula e minúscula.

Baixe a fonte

Abaixo da tag <style> use

**@font-face {**

**Font Family: ’’ ;** 🡪 Aqui você pode colocar o nome que quiser entre as ‘’

**Src: url**(**‘**coleque aqui o nome EXATO do arquivo.mídia**’** dele) **format ();**

Após fechar o format (), você pode adicionar outra url 🡪 É sempre recomendável devido aquele negócio dos navegadores.

Tipos de format ()

- **opentype** (otf)

- **truetype** (ttf)

- embed-opentype (etf)

- truetype-aat (apple advanced typography)

- svg

**Aula 8 – Capturando as Fontes usadas em um site – Capítulo 14 do material**

**Extensão Fonts Ninja** no Chrome

É só entrar no site e abrir a extensão, você pode passar o mouse em cima da fonte e ele te dará todas as características dela.

**Aula 9 – Detectando Fontes dentro de uma imagem – Capítulo 14 do material**

IMPORTANTE: Os resultados não são tão precisos quando é dentro de uma imagem

<https://www.whatfontis.com/> -- O melhor

<https://www.fontsquirrel.com/> - Vá em font identifire

Aumente o contraste e reduza o brilho entre as letras para torna-las mais identificáveis para o algoritmo.

**Aula 10 – Alinhamento de textos com CSS – Capítulo 14 do material**

Dentro do Exercício 17 🡪 fonte02.html

**Text-align: ;**

Pode ser center, left, right ou justify

**Text-indent: Zpx**; 🡪 Da aquele afastamento no início de um parágrafo (tipo a regra da ABNT nos artigos).

**Aula 1 – Usando o ID com CSS – Capítulo 15 do material**

Exercício 19

Use o ID quando você quer 2 formatações diferentes em uma mesma tag, EX: 1 h2 formatado de um jeito e outro h2 formatado de outro jeito.

**APENAS 1 TAG PODE RECEBER ESSE PARÂMETRO ID**

No Exemplo – No HTML: <h1 **id=”principal”**>

No Exemplo – No CSS: **h1#principal { OU SÓ #principal {**

Você escolhe o nome que quiser na “” após o ID.

Tudo que em HTML é **id** em CSS é **#**

Tudo que em HTML é **class** em CSS é **. 🡪 Ponto**

**Aula 2 – As diferenças entre ID e CLASS – Capítulo 15 do material**

Exercício 19

Dentro do HTML **só pode ter 1 código com ID**

Não dê classificação pela forma no “” e sim pela função.

No Exemplo

No HTML: <h2 **class=”basico”**> <h2 **class=”intermediario”**> <h2 **class=”avancado”**> . Tanto na H2 Aprendendo HTML quanto no Aprendendo CSS

No CSS:

**.basico {**

**.intermediario {**

**.avancado {**

**Você pode colocar + de 1 classe em um elemento, os 2 nomes devem estar dentro da “”;**

Você também pode colocar classe no elemento que tem ID, mas o ID irá se sobrepor a class

**Primeiro configure o genérico (EX: <h1>) e depois os termos com classe (variações).**

O **ID se sobrepõe ao termo genérico e à class** (veja a H1 do exemplo)

**Ordem: Class, Genérico < ID**

**Aula 3 – Pseudo-classes em CSS – Capítulo 15 do material**

Exercício 20 🡪 VERR

Abra tags <div> no <body> e depois vá ajustando o CSS dela entre a tag style ou no arquivo.css

A Div ocupa a 1 linha de ponta a ponta

**Shorthand border: Largura Tipo Cor**

O parâmetro **display: inline-block;** fará as divs ficarem uma do lado da outra, em linha reta

As pseudoclasses são representadas pelos 2 pontos 🡪 **:**

Elas devem estar relacionadas a um elemento ou uma classe (relacionada ao **estado** de um determinado elemento)

Entre a <style> ele fez **div:hover** { 🡪 Significa quando passar o mouse por cima da div)

Você pode iluminar o elemento quando passar com o mouse em cima dele com o parâmetro acima.

Veja o exemplo da pasta Exercicio019 – Pasta hover.html

Para mexer APENAS no elemento que está dentro de outro 🡪 Na tag style abra o parâmetro elemento 1 > elemento 2 { }

No exercício:

Existem 2 tags <p> no body e ele quer alterar apenas o <p> que está dentro da <div>, então:

<style>

div > p { **display: none** }

</style>

O **display: none** faz o “TEXTO ESCONDIDO” sumir, para que esse texto volte a aparecer quando passar o mouse em cima, faça:

<style>

Div:hover > p { **display: block** }

</style>

**Essas configurações são feitas no menu, quando passa o mouse em cima, vai abrir várias opções clicáveis.**

No capítulo 15 do material tem várias pseudoclasses além do hover.

**Aula 4 – Pseudo-elementos em CSS – Capítulo 15 do material**

Exercício 20 🡪 links.html

Para criar um link dentro do texto entre <li>, selecione o texto e envelope ele com a tag <a>

Mudar a cor de um **link visitado**:

**a:visited { color:**

Para retirar o sublinhado dos links:

**a { text-decoration: none;**

Para sublinhar e ficar preto quando passar o mouse em cima:

**a:hover { text-decoration: underline; e color:black}**

Para o link ficar amarelo **quando você clicar** nele:

**a:active { color: yellow;**

Os pseudo-elementos podem funcionar nas **classes**, elementos ou **Ids**, eles são representados por 🡪 **::**

Para indicar [LINK] **após** todos os links:

**a::after { content: ‘ (link)’;**

**RESUMÃO DOS SELETORES PERSONALIZADOS**

**# 🡪 ID**

**. 🡪 Class**

**: 🡪 Pseudo-classe**

**:: 🡪 Pseudo-elemento**

**> 🡪 Children**

**Aula 1 – Modelo de Caixas: primeiros passos (Box Model) – Capítulo 16 do material**

A estrutura HTML é como se fosse uma caixa, e podem ter várias caixas encaixadas uma dentro da outra (tipo aquela boneca russa kkk), isso é chamado de **aninhamento**

A caixa seria tipo aquele seletor retangular que fica circundado as frases.

**TODAS AS MEDIDAS ABAIXOS SÃO EM PX**

**Height** 🡪 Altura da caixa

**Widht** 🡪 Largura da caixa

**Border** 🡪 É a borda do conteúdo (ela tem o parâmetro **widht, solid, color, etc.)**

**Padding 🡪** É usado para aumentar a amplitude da caixa, aumentando o **espaço interno**, fazendo ela não ficar tão colada no texto; é como se fosse a engorda da caixa (distancia a borda do conteúdo). Possui o parâmetro **top, right, bottom e left (siga o sentido horário)**

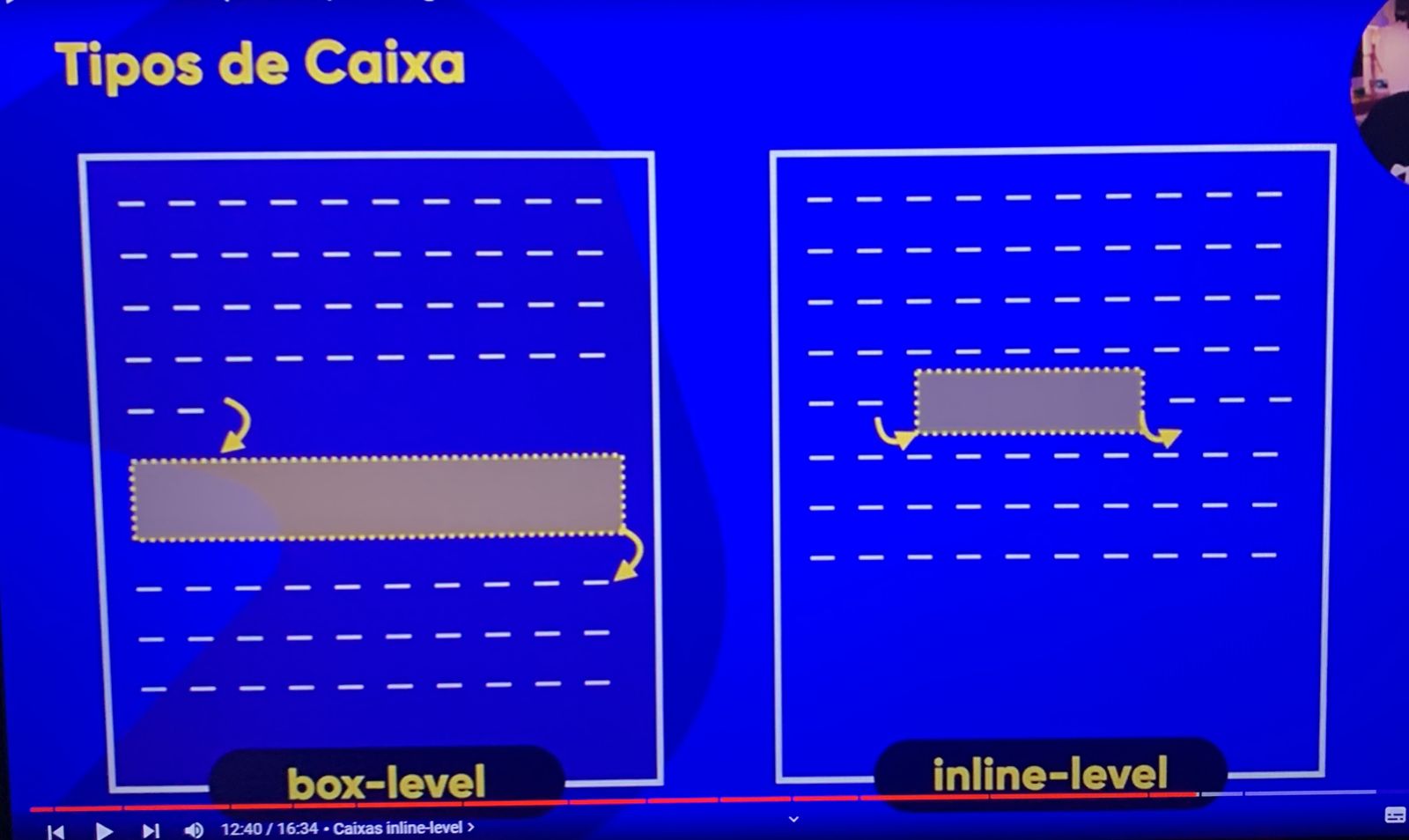
**Margin** 🡪 É o **espaço externo** que fica ao redor da borda, esse parâmetro será utilizado para as bordas entre os elementos não se encostarem. Possui o parâmetro **top, right, bottom e left (siga o sentido horário) E auto (explicado abaixo, aula 3 desse módulo)**

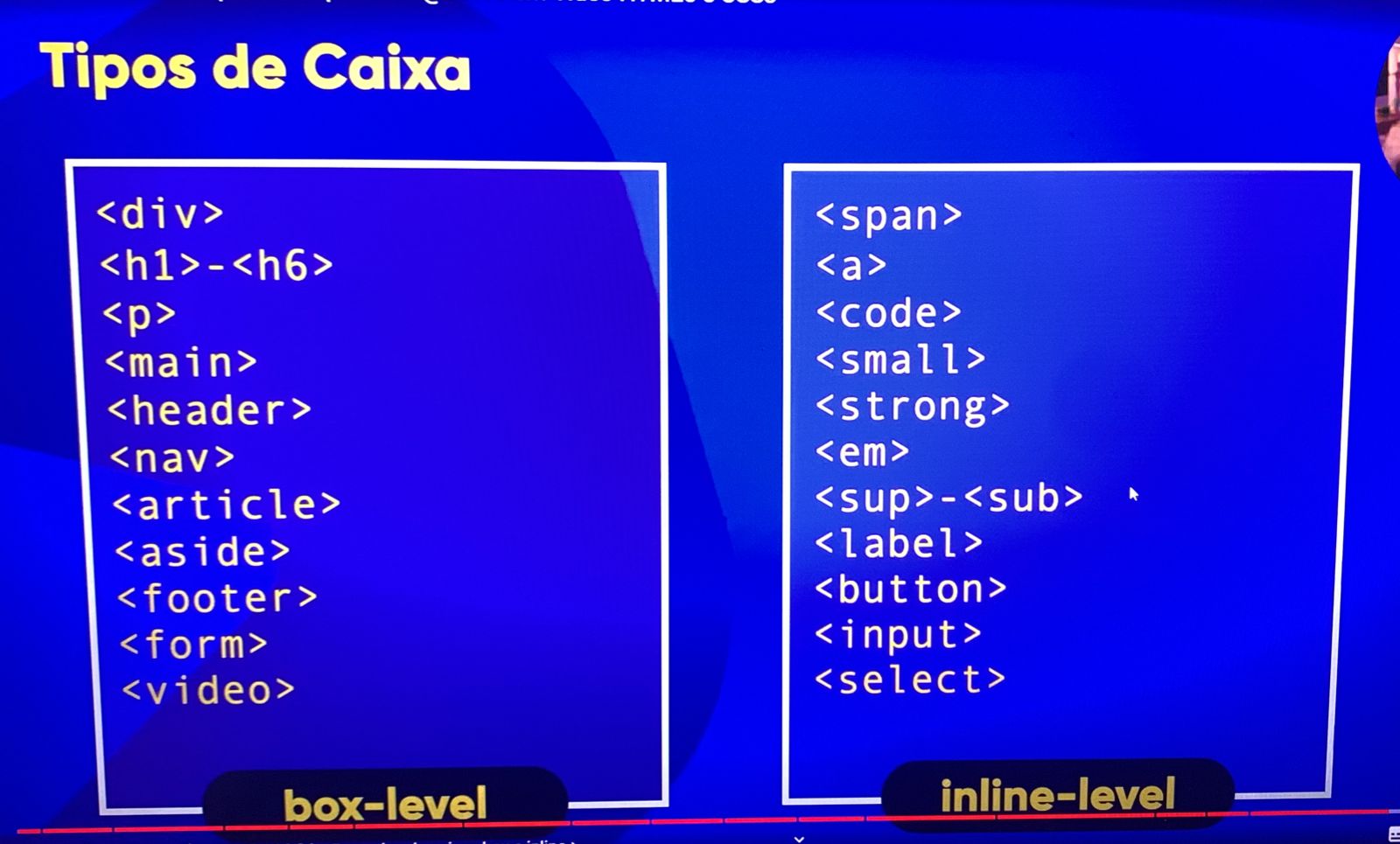
**Outline 🡪** É o espaço que fica entre a borda da caixa e o espaçamento externo da margem. Chamado de contorno ou traçado. CONFIGURADO SEMELHANTE A BORDA.

**Tipos de caixas:**

**Box-level:** Sempre se inicia em uma **nova linha**. Ele **ocupa a largura total** da tela.

**Inline-level:** Não pula a linha e não ocupa o tamanho inteiro e dá continuidade ao conteúdo na mesma linha.





**Aula 2 – Modelo de Caixas na prática (parte 1) – Capítulo 16 do material**

Exercício 21

H1 é box level e o <a> é inline-level. Para transformar Inline-level em box level use **display: block;**

Toque no inspecionar uma página e desça o dev tools – Ele já mostra o box level, e quando você clica no elemento (ex na h1), ele já mostra todos os parâmetros dele

**User agent** é a configuração nativa do navegador.

**Display: block;** é um elemento do tipo box-level.

**Você consegue alterar as coisas do site no dev tools sem mexer no código. Basta clicar no elemento e alterar diretamente no dev tools. Você também pode desligar algum parâmetro de um elemento.**

**Aula 3 – Modelo de Caixas na prática (parte 2) – Capítulo 16 do material**

Cont Exercício 21

**Margin: auto;** 🡪 irá centralizar a caixa no meio, independentemente do tamanho do navegador. Se for pra centralizar um texto use o text-aligment: center.

O outline fica dentro da margem e colado na borda.

**SIMPLIFICANDO A BORDER, PADDIN, MARGIN E OUTLINE C/ SHORTHAND E SUA ORDEM**

**Border** 🡪 width, style, color

**Paddin** 🡪 topo, right, bottom & left. Se todas forem **iguais**, basta informar apenas 1 medida. Se for inserido apenas **2 valores**: O 1º dita TOPO & BOTTOM e o outro RIGHT & LEFT

**Margin** 🡪 topo, right, bottom & left. Se quiser **centralizar** horizontalmente a margem coloque **auto** nos valores de right e left.

**Outline** 🡪 width, style, color

**Aula 4 – Grouping Tags em HTML5 – Capítulo 16 do material**

Pasta Exercício 21 🡪 Arquivo caixa02.html

Todo site tem um conteúdo principal (**<main>**), o cabeçalho (**<header>**) e o rodapé (**<footer>**)

Para **menu** 🡪 Dentro do <header>, abre uma <h1> e em baixo use **<nav>** e entre este último vários links <a>.

Dentro do **<main>** 🡪 Pode ter várias **<sections é legal ter id ou class para cada>** e dentro de cada section pode ter vários **<article>** e entre eles podem ter **<h2>, <p>** e **<aside>** (texto ao lado).

**Aula 5 – Sombra nas caixas – Capítulo 16 do material**

Sombras devem ser suaves, use o parâmetro **box-shadow: ;**

Elas já funcionam como shorthand, se acordo com a ordem:

Deslocamento horizontal – deslocamento vertical – espalhamento – cor da sombra

EX: 1px 1px 1px black;

Inset 🡪 Sombra interna.

Dentro do dev tools, ao selecionar a tag que deseja adicionar a sombra, vá no “+” e lá te a opção add box-shadow

Sempre reduza a saturação do preto da sombra.

**Aula 6 – Caixa com vértice arredondado – Capítulo 16 do material**

Use o parâmetro **border-radius: Zpx (também pode ser em %)**; para dar o arredondamento da borda. Aqui você também pode seguir a lógica do shorthand de sentido horário

**Aula 7 – Bordas decoradas – Capítulo 16 do material**

Exercício 19 – arquivo caixa03.html

A borda decorada deve ser criada no GIMP e o parâmetro border deve ser inserido antes.

Use os parâmetros a seguir para fazer a borda decorada com uma imagem:

**Border-image-sourcer: url(‘nomedoarquivodaimagem.png’);**

**Border-image-slice: número qualquer;** esse número qualquer você ajusta no dev tools para ficar tudo OK e depois ajusta no VSC

**Border-image-repeat: repeat;**

Você também pode usar o shorthand e usar os 3 parâmetros seguidos

EX: **border-image: url(‘borda.png’) 27 repeat;**

**XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX**

**Aula 3 – Planejando a Estrutura do Site do Desafio – Capítulo 17 do material**

Desafio 10 – VEJA A APOSTILA DO CAP 17

Antes de começar a programar um site você deve:

Planejar o Layout, as cores e as fontes! Para isso, use o site abaixo: e vá em UI Drowings

[www.mockflow.com](http://www.mockflow.com)

As cores serão planejadas no adobe color

Fonte: <https://fonts.google.com/specimen/Bebas+Neue?query=bebas>

**Aula 6 – Variáveis em CSS– Capítulo 17 do material**

Para facilitar a colocação das cores ele criou no css uma psudo-classe

**:root {**

**--**nomedavariável: Valor;

**--** 🡪 representa a variável

Você atribui um nome e um valor para alguma variável como por exemplo a #numeração de uma cor.

O **..**/ antes da url com a fonte Android, significa que o style.css q está dentro da página estilo vai ter q **voltar uma pasta** para poder acessar a pasta que contém as fontes.

Ele usa o global **\* { margin: 0px; padding: 0px}** 🡪 Para facilitar

Com o dev tools aberto no site e você for testar aumentar ou reduzir a tela (responsividade), perceba q nem cima vai dizer o valor em pixels da altera e largura.

**Defina um valor mínimo e máximo de largura, no qual o conteúdo é lido com facilidade**

Ele definiu o valor máximo de 800px de largura e o mínimo é 400px (natural do navegador). Sua interface tem que ficar bem em 800px (valor do máximo definido) e 400px (valor mínimo definido).

**Aula 7 – Responsividade para sites – Capítulo 17 do material**

**Faça as configurações de min-widht e max-widht no MAIN ou div principal**

Limite o tamanho mínimo e máximo de uma interface.

O ideal é colocar de 1.000 a 1.200px de largura, acima disso você deve criar um bloco que seja adaptável, mas que não vai ser muito largo.

Ele notou que quando reduzia abaixo de 800px na largura, a imagem começava a quebrar e vazar

Dentro da pasta do desafio ele criou o arquivo responsivo.html

**Na responsividade, você não deseja a criação da barra lateral de baixo.**

Baixamos o plugin do Chrome: Window Resiezer; quando você abre ela e aperta no “ I “ q tem, você verá o **tamanho da viewport.**

Ele vê o tamanho máximo que a imagem não quebra e coloca o parâmetro widht:

Quando a tela for muito grande você pode **alinhar** o conteúdo ao **centro** com **margin: auto**

O fator acima resolve o fator tela grande, mas não a tela pequena, então use no **main {:**

**Min-widht: 280px;** 🡪 Largura mínima dos celulares antigos

**Max-widht: 800px;**

O conteúdo está sendo responsivo, mas a imagem ainda está quebrando, então faça na <style>

**Img { widht: 100%;**

**PASSO A PASSO DA RESPONSIVIDADE:**

1. Defina um tamanho máximo **(max-widht)** no qual o conteúdo fica bem legível.
2. Defina um tamanho mínimo **(min-widht: 280px;)**, caso ainda quebre você deve avaliar o px do padding e da margin.
3. Use **margin: auto** para centralizar o conteúdo principal em grandes telas;
4. Nas imagens faça **img { widht: 100%**
5. A imagem p no max-widht ele escolhe um tamanho de largura que quando trocar a imagem (pequena) não ficará muito grande.

**Aula 8 – Responsividade na prática – Capítulo 17 do material**

Ele coloca no **main { min-widht: 300px** Conteúdo adapta, mas imagem ainda não

**Max-widht: 1000px**

P/ adaptar imagem faça 🡪 **main img { widht: 100%;** 🡪 Adapta para **tela pequena** e

**Main { margin:auto;** 🡪 Adapta para **telas grandes e centralizar conteúdo**

Para que a imagem do símbolo do homem com o Android não fique distorcida em grandes telas, ele **adicionou a classe pequena na <img> dele.**

No CSS: main img.pequena {**max-widht: 350px; 🡪** 50px menor do q o tamanho da imagem.

**Margin: auto;**

**Display: block** 🡪 Ele usa esse parâmetro para o margin: auto; funcionar, já que imagem não é um bloco (box-size) e não tem margem, então esse parâmetro adiciona a margem e consequentemente centraliza a imagem

**Aula 9 – Configurando o Header e o Menu do site – Capítulo 17 do material**

Ele colocou altura mínima (**min-height**) no header para distanciar entre o <p> e o <nav>

Ele usou padding top no header, pq quando fez **margin-top** vazou a cor do background da **<main>.**

Ele fez **max-widht:500px** no **<p>** do header para alinhar com H1, após isso o <p> virou um bloco, e para centralizar esse bloco ele fez **margin:auto** para centralizar.

Ele fez padding nas laterais do <p> e também sombra no texto.

No **nav > a {transition-duration: 0.5s** para aparecer as alterações de leves que foram criadas no **nav > a:hover**.

**Aula 10 – Melhorando o Formato do Conteúdo – Capítulo 17 do material**

Ele usou main p, invés de main > p, porque dentro do <main> ainda tem um <article>; a ausência do > significa que irá mudar todos os <p>, independente se ele estiver dentro de outras tags.

Ele também usou dentro do **main p {line-height: 2em;** 🡪 Espaçamento entre linhas

Macete do gradiente nas H2:

**background-image: linear-gradient(to right, var(--cor1), transparent);**

**Aula 11 – Conteúdo periférico e o rodapé – Capítulo 17 do material**

Ele usa uma **\** (para dar espaço entre o check da UL e os links externos com emojis) antes do código do emoji que fica depois do +.

Caso queira mais espaço use **\00A0\** 🡪 Se quer a direita faça antes de código e se quer a direita, faça depois do código do emoji.

Ele fez **a.externo::after (em cima do body)** e depois teve que criar essa classe no html

Content: ‘’; 🡪 código do emoji

**Aula 12 – Tornando um vídeo responsivo – Capítulo 17 do material**

Ele começou tirando os negritos dos h1 (header e main) e h2 (main), q já vem em negrito como padrão.

Para tornar o vídeo responsivo, ele começa envelopando-o com uma **<div class=’video’),** assim o vídeo **(<iframe>)** fica dentro de uma caixa maior, que é a **<div>**

Ele começa a configurar o padding, background e margem da **div.video {**

Depois ele via configurar **div.video > iframe {**

Ele adiciona o parâmetro:

**div.video {**

**position: relative;**

**padding-bottom: 59%;** 🡪 esse 59% foi pra não ficar a barra lateral quando dá o play no vídeo.

**Margin left e right: -20px;** 🡪 para preencher o bg color até a largura do espaço branco (bg color da main)

**Margin-bottom: -15px;** 🡪 Para dar um distanciamento da caixa do vídeo até a seção inferior (aside)

**div.video > iframe {**

**position: absolute; top: 5%; left: 5%; widht: 90%; height: 90%;**

**top e left:** foi para descolar de cima e da esquerda

**widht e height:** foi testando até não ficar uma tarja preta no vídeo quando ele é reproduzido.