

JORNADA TECH



Fab
Lab

JORNADA TECH

OFICINA 1: LÓGICA DE
PROGRAMAÇÃO



JORNADA
TECH



BEM-VINDOS À JORNADA TECH!

O QUE É O JORNADA TECH?

- UMA INICIATIVA PARA DEMOCRATIZAR O ACESSO AO CONHECIMENTO EM ENGENHARIA DA COMPUTAÇÃO.
- QUEREMOS DESPERTAR SEU INTERESSE PELA TECNOLOGIA E INCENTIVAR O RACIOCÍNIO LÓGICO, A CRIATIVIDADE E O PROTAGONISMO.

OBJETIVO GERAL DO CURSO:

- OFERECER UMA FORMAÇÃO INTRODUTÓRIA, PRÁTICA E MOTIVADORA NAS ÁREAS DA ENGENHARIA DA COMPUTAÇÃO.
- AMPLIAR SEUS HORIZONTES ACADÊMICOS E PROFISSIONAIS.

PARA QUEM É ESTE CURSO?

- JOVENS DE CUIABÁ E REGIÃO COM INTERESSE EM TECNOLOGIA!
- ESPECIALMENTE PARA QUEM AINDA NÃO TEVE UMA FORMAÇÃO PRÁTICA NO SETOR.



A OFICINA DE HOJE: LÓGICA DE PROGRAMAÇÃO (4 HORAS)

O QUE VAMOS APRENDER JUNTOS? O QUE SÃO ALGORITMOS E COMO PENSÁ-LOS.

- COMO TOMAR DECISÕES EM PROGRAMAS USANDO A LINGUAGEM C.
- COMO CRIAR REPETIÇÕES (LAÇOS) EM C.
- COMO APLICAR A LÓGICA PARA RESOLVER PROBLEMAS COM C.

POR QUE LÓGICA DE PROGRAMAÇÃO É IMPORTANTE?

- É A BASE PARA CRIAR QUALQUER PROGRAMA OU APLICATIVO! APRENDER COM C CONSTRÓI UMA BASE SÓLIDA.

NOSSA METODOLOGIA:

- TEORIA INTRODUTÓRIA COM LINGUAGEM ACESSÍVEL.
- MÃO NA MASSA COM C: ATIVIDADES PRÁTICAS! ("APRENDER FAZENDO").
- ESPAÇO PARA DÚVIDAS, TROCAS E COMPARTILHAR DESCOBERTAS.



O QUE É LÓGICA? E LÓGICA DE PROGRAMAÇÃO?

LÓGICA NO NOSO DIA A DIA:

EXEMPLO: SE O SINAL ESTÁ VERMELHO, EU PARO. SE ESTÁ VERDE, EU SIGO.

É A NOSSA CAPACIDADE DE PENSAR DE FORMA COERENTE PARA TOMAR DECISÕES E RESOLVER PROBLEMAS.

LÓGICA DE PROGRAMAÇÃO:

É UM CONJUNTO DE REGRAS E PROCESSOS PARA CRIAR PROGRAMAS DE COMPUTADOR QUE REALIZAM TAREFAS ESPECÍFICAS.

ENVOLVE ANALISAR PROBLEMAS E DEFINIR SOLUÇÕES EFICIENTES.

ESSENCIAL PARA DESENVOLVER QUALQUER SOFTWARE, SEJA PARA SISTEMAS, JOGOS OU APLICATIVOS.



O QUE É?

A HABILIDADE DE RESOLVER PROBLEMAS DE FORMA ESTRUTURADA E LÓGICA, USANDO CONCEITOS DA COMPUTAÇÃO.

COMO FUNCIONA?

- DECOMPOSIÇÃO: QUEBRAR UM PROBLEMA GRANDE EM PARTES MENORES E MAIS FÁCEIS DE RESOLVER.
- EXEMPLO: PARA FAZER UM BOLO, VOCÊ DECOMPÕE EM: SEPARAR INGREDIENTES, MISTURAR, ASSAR.
- RECONHECIMENTO DE PADRÕES: IDENTIFICAR SIMILARIDADES ENTRE OS PROBLEMAS.
- ABSTRAÇÃO: FOCAR NO QUE É IMPORTANTE, IGNORANDO DETALHES DESNECESSÁRIOS.
- CRIAÇÃO DE ALGORITMOS: DESENVOLVER UM PASSO A PASSO PARA SOLUCIONAR CADA PARTE DO PROBLEMA.



ALGORITMOS: A RECEITA DO PROGRAMA!

O QUE SÃO ALGORITMOS?**

SÃO SEQUÊNCIAS DE PASSOS OU INSTRUÇÕES CLARAS E FINITAS QUE VISAM RESOLVER UM PROBLEMA OU EXECUTAR UMA TAREFA.

EXEMPLOS DO COTIDIANO:

- UMA RECEITA DE BOLO.
- INSTRUÇÕES PARA MONTAR UM MÓVEL.
- O CAMINHO QUE VOCÊ FAZ DE CASA PARA A ESCOLA.

CARACTERÍSTICAS DE UM BOM ALGORITMO:

- FINITO: TEM UM COMEÇO E UM FIM.
- BEM DEFINIDO: CADA PASSO É CLARO E NÃO DEIXA DÚVIDAS.
- EFICAZ: RESOLVE O PROBLEMA PROPOSTO.



ESTRUTURAS BÁSICAS DE UM ALGORITMO

TODO ALGORITMO, E CONSEQUENTEMENTE UM PROGRAMA EM C, POR MAIS SIMPLES OU COMPLEXO, GERALMENTE POSSUI:

ENTRADA:

- OS DADOS QUE O PROGRAMA PRECISA PARA COMEÇAR (EX: VIA TECLADO).
- EXEMPLO (SOMAR DOIS NÚMEROS): OS NÚMEROS 5 E 7.

PROCESSAMENTO:

- AS OPERAÇÕES OU CÁLCULOS QUE O PROGRAMA REALIZA COM OS DADOS DE ENTRADA.
- EXEMPLO: A OPERAÇÃO DE SOMA $5 + 7$.

SAÍDA:

- O RESULTADO GERADO PELO PROGRAMA APÓS O PROCESSAMENTO (EX: MOSTRADO NA TELA).
- EXEMPLO: O RESULTADO DA SOMA, QUE É 12.



O QUE É?

UMA REPRESENTAÇÃO GRÁFICA DE UM ALGORITMO.

UTILIZA SÍMBOLOS GEOMÉTRICOS PARA REPRESENTAR DIFERENTES AÇÕES E DECISÕES.

AJUDA A VISUALIZAR O "FLUXO" DA SOLUÇÃO DO PROBLEMA.

SÍMBOLOS COMUNS:

- INÍCIO/FIM: (OVAL)
- PROCESSO: (RETÂNGULO) - EX: "SOMAR A + B"
- ENTRADA/SAÍDA: (PARALELOGRAMO) - EX: "LER NÚMERO", "MOSTRAR RESULTADO"
- DECISÃO: (LOSANGO) - EX: "NÚMERO > 0?"
- SETAS: INDICAM A DIREÇÃO DO FLUXO.



INTRODUÇÃO À LINGUAGEM C

O QUE É A LINGUAGEM C?

- UMA LINGUAGEM DE PROGRAMAÇÃO ESTRUTURADA, PODEROSA E FUNDAMENTAL, CRIADA NOS ANOS 70.
- É A BASE PARA MUITAS OUTRAS LINGUAGENS MODERNAS (C++, JAVA, C#, PYTHON EM PARTES).
- PERMITE UM BOM NÍVEL DE CONTROLE SOBRE O HARDWARE DO COMPUTADOR.

POR QUE APRENDER C?

- DESENVOLVE UMA BASE SÓLIDA EM LÓGICA E ESTRUTURA DE PROGRAMAS.
- ENSINA CONCEITOS CRUCIAIS COMO TIPOS DE DADOS, VARIÁVEIS, OPERADORES, CONTROLE DE FLUXO (DECISÕES, LAÇOS) E FUNÇÕES.



FUNDAMENTOS DA LINGUAGEM C: VARIÁVEIS, ENTRADA E SAÍDA

O QUE SÃO VARIÁVEIS?

- ESPAÇOS NA MEMÓRIA DO COMPUTADOR RESERVADOS PARA ARMAZENAR DADOS QUE SEU PROGRAMA UTILIZARÁ.
- CADA VARIÁVEL POSSUI UM TIPO, UM NOME E PODE CONTER UM VALOR.
- COMO DECLARAR (CRIAR) UMA VARIÁVEL: TIPO NOME_DA_VARIAVEL;
- EXEMPLO: INT IDADE_DO_ALUNO;

PRINCIPAIS TIPOS DE VARIÁVEIS EM C:

- INT: ARMAZENA NÚMEROS INTEIROS (SEM CASAS DECIMAIOS).
- EX: INT GOLS_MARCADOS = 3;
- FLOAT: ARMAZENA NÚMEROS COM CASAS DECIMAIOS (PONTO FLUTUANTE DE PRECISÃO SIMPLES).
- EX: FLOAT ALTURA_METRO = 1.75;
- CHAR: ARMAZENA UM ÚNICO CARACTERE (LETRA, NÚMERO COMO SÍMBOLO, PONTUAÇÃO). USA ASPAS SIMPLES ' '.
- EX: CHAR OPCAO_MENU = 'A';

MOSTRANDO DADOS NA TELA: A FUNÇÃO PRINTF()

- USADA PARA EXIBIR TEXTO E VALORES DE VARIÁVEIS NO CONSOLE (SAÍDA PADRÃO).
- SINTAXE BÁSICA PARA TEXTO: `PRINTF("QUALQUER TEXTO AQUI!\n");`
- PARA EXIBIR VARIÁVEIS, USAMOS ESPECIFICADORES DE FORMATO:
- %D OU %I PARA INT: `PRINTF("IDADE: %d ANOS\n", IDADE_DO_ALUNO);`
- %F PARA FLOAT: `PRINTF("PREÇO: R$ %f\n", VALOR_PRODUTO);`
- %C OU %S PARA CHAR: `PRINTF("LETRA ESCOLHIDA: %c\n", MINHA_LETRA);`
- O \N NO FINAL DO TEXTO PULA PARA UMA NOVA LINHA

LENDODADOS DO TECLADO: A FUNÇÃO SCNF()

- USADA PARA LER DADOS QUE O USUÁRIO DIGITA NO TECLADO E ARMAZENÁ-LOS EM VARIÁVEIS.
- SINTAXE: SCNF("ESPECIFICADOR_DE_FORMATO", &NOME_DA_VARIAVEL);
- IMPORTANTE: NOTE O & ANTES DO NOME DA VARIÁVEL. ELE INDICA O ENDEREÇO DA VARIÁVEL NA MEMÓRIA.
- SCNF("%D", &GOLS_MARCADOS);
- SCNF("%F", &ALTURA_METRO);
- SCNF("%C OU %S", &OPCAO_MENU);

INCLUSÃO NECESSÁRIA: PARA USAR PRINTF() E SCNF(), VOCÊ DEVE INCLUIR A BIBLIOTECA PADRÃO DE ENTRADA/SAÍDA NO INÍCIO DO SEU ARQUIVO C:

```
#INCLUDE <STDIO.H>
```



INTRODUÇÃO À LINGUAGEM C: PRIMEIRO PROGRAMA

- ESTRUTURA BÁSICA DE UM PROGRAMA EM C: TODO PROGRAMA C EXECUTA A PARTIR DA FUNÇÃO MAIN().
- COMANDOS SÃO GERALMENTE TERMINADOS COM PONTO E VÍRGULA ;.
- BLOCOS DE CÓDIGO SÃO DELIMITADOS POR CHAVES { }.

```
#include <stdio.h> // Biblioteca para entrada (scanf) e saída (printf) de dados

int main() { // Função principal onde o programa começa
    // Declaração de variáveis
    int numero1, numero2, soma;

    printf("Digite o primeiro número: ");
    scanf("%d", &numero1); // Lê o primeiro número do teclado

    printf("Digite o segundo número: ");
    scanf("%d", &numero2); // Lê o segundo número do teclado

    // Processamento
    soma = numero1 + numero2; // Realiza a soma

    // Saída
    printf("A soma de %d e %d é: %d\n", numero1, numero2, soma); // Mostra o resultado

    return 0; // Indica que o programa terminou com sucesso
}
```



AMBIENTE DE PRÁTICA: ONLINEGDB (COMPILADOR C ONLINE)

COMO USAR (VISÃO GERAL):

- ACESSE O SITE (EX: WWW.ONLINEGDB.COM).
- NO CANTO SUPERIOR DIREITO, SELECIONE A LINGUAGEM "C".
- DIGITE SEU CÓDIGO NA JANELA PRINCIPAL (EDITOR).
- CLIQUE NO BOTÃO VERDE "RUN" (EXECUTAR) NO MENU SUPERIOR.
- VEJA A SAÍDA DO SEU PROGRAMA NA JANELA INFERIOR ("OUTPUT"). SE HOUVER ERROS, ELES TAMBÉM APARECERÃO LÁ.

VAMOS COMEÇAR! 

```
#include <stdio.h>

int main() {
    printf("Olá, Jornada Tech!\n");
    return 0;
}
```



O QUE SÃO INSTRUÇÕES CONDICIONAIS (COMANDOS IF/ELSE)?

- PERMITEM QUE O PROGRAMA EXECUTE DIFERENTES BLOCOS DE CÓDIGO DEPENDENDO SE UMA CONDIÇÃO É VERDADEIRA (TRUE) OU FALSA (FALSE).
- É COMO DIZER AO COMPUTADOR: "SE (ALGO FOR VERDADE), FAÇA ISSO. SENÃO (CASO CONTRÁRIO), FAÇA AQUILO."

```
if (condicao) {  
    // Bloco de código se a 'condicao' for VERDADEIRA  
} else {  
    // Bloco de código se a 'condicao' for FALSA (o 'else' é opcional)  
}
```



EXEMPLO DE DECISÃO

SITUAÇÃO:

VERIFICAR SE UM ALUNO FOI APROVADO (MÉDIA PARA APROVAÇÃO: 6.0).

```
#include <stdio.h>

int main() {
    float media;

    printf("Digite a média do aluno: ");
    scanf("%f", &media);

    if (media >= 6.0) {
        printf("Aluno Aprovado!\n");
    } else {
        printf("Aluno Reprovado.\n");
    }

    return 0;
}
```



ATIVIDADE PRÁTICA 1: HORA DE DECIDIR!

DESAFIO:

1. CRIE UM PROGRAMA EM C QUE PEÇA A IDADE DO USUÁRIO.
2. SE A IDADE FOR MAIOR OU IGUAL A 18, MOSTRE A MENSAGEM "VOCÊ É MAIOR DE IDADE."
3. SENÃO, MOSTRE "VOCÊ É MENOR DE IDADE."

DICA: LEMBRE-SE DE INCLUIR <STDIO.H>, USAR PRINTF PARA MENSAGENS, SCANF PARA LER A IDADE, E A ESTRUTURA IF/ELSE.



LAÇOS DE REPETIÇÃO: REPETINDO TAREFAS

O QUE SÃO LAÇOS DE REPETIÇÃO?

- ESTRUTURAS QUE PERMITEM EXECUTAR UM MESMO BLOCO DE CÓDIGO VÁRIAS VEZES, ATÉ QUE UMA CONDIÇÃO DE PARADA SEJA SATISFEITA OU UM NÚMERO DEFINIDO DE VEZES SEJA ATINGIDO.
- EVITAM QUE A GENTE PRECISE ESCREVER O MESMO COMANDO REPETIDAMENTE.

EXEMPLOS:

- CONTAR DE 1 ATÉ 10.
- PEDIR UMA SENHA AO USUÁRIO REPETIDAMENTE ATÉ QUE ELE ACERTE.

6 TIPOS DE LAÇOS (CONCEITUAL)

LAÇO FOR:

USADO GERALMENTE QUANDO SABEMOS QUANTAS VEZES QUEREMOS REPETIR UM BLOCO DE CÓDIGO.

ESTRUTURA EM C:

```
for (inicializacao; condicao_de_continuacao; incremento_decremento) {  
    // Código a ser repetido  
}
```

LAÇO WHILE:

USADO QUANDO A REPETIÇÃO DEPENDE DE UMA CONDIÇÃO SER VERDADEIRA. O CÓDIGO REPETE ENQUANTO A CONDIÇÃO FOR VERDADEIRA.

```
while (condicao_de_continuacao) {  
    // Código a ser repetido  
    // (Importante: algo dentro do laço deve eventualmente tornar a condição falsa)  
}
```



ATIVIDADE PRÁTICA 2: MÃOS À OBRA COM REPETIÇÕES!

DESAFIO 1 (LAÇO FOR):

- CRIE UM PROGRAMA EM C QUE MOSTRE OS NÚMEROS DE 1 A 5 NA TELA, CADA UM EM UMA NOVA LINHA.

DESAFIO 2 (LAÇO WHILE):

- IMAGINE UM JOGO ONDE O USUÁRIO TEM QUE ADIVINHAR UM NÚMERO SECRETO (EX: 7).
- ESCREVA UM PROGRAMA EM C QUE PEÇA UM NÚMERO AO USUÁRIO. ENQUANTO O NÚMERO DIGITADO NÃO FOR 7, ELE DEVE DIZER "ERROU, TENTE NOVAMENTE!" E PEDIR OUTRO NÚMERO. SE ACERTAR, DIZ "PARABÉNS!".



RESOLVENDO PROBLEMAS: A ABORDAGEM CORRETA

COMPREENDER O PROBLEMA:

- LEIA ATENTAMENTE, ENTENDA O QUE É PEDIDO, QUAL O OBJETIVO.

DIVIDIR EM SUBPROBLEMAS (DECOMPOR):

- QUEBRE O PROBLEMA GRANDE EM PARTES MENORES E MAIS GERENCIÁVEIS.

DESENVOLVER A LÓGICA E ESCREVER O CÓDIGO:

- PARA CADA SUBPROBLEMA, PENSE NA SEQUÊNCIA DE PASSOS (PODE USAR FLUXOGRAMA COMO APOIO VISUAL) E COMECE A ESTRUTURAR SEU CÓDIGO EM C.

IMPLEMENTAR, TESTAR E DEPURAR:

- ESCREVA O CÓDIGO EM C.
- COMPILE E EXECUTE SEU PROGRAMA (USANDO O ONLINEGDB).
- TESTE COM DIFERENTES ENTRADAS PARA VER SE FUNCIONA CORRETAMENTE E CORRIJA OS ERROS (BUGS).



EXEMPLO PRÁTICO: SOMA DE 1 A N

- PROBLEMA: CALCULAR A SOMA DE TODOS OS NÚMEROS INTEIROS DE 1 ATÉ UM NÚMERO N, FORNECIDO PELO USUÁRIO.
- 1. COMPREENDER: QUERO SOMAR $1 + 2 + 3 + \dots + N$.
- 2. DECOMPOR: PRECISO PEDIR O NÚMERO N AO USUÁRIO (ENTRADA: SCANF).
- PRECISO DE UMA VARIÁVEL PARA GUARDAR A SOMA, COMEÇANDO EM 0.
- PRECISO REPETIR A SOMA PARA CADA NÚMERO DE 1 ATÉ N (LAÇO FOR).
- PRECISO MOSTRAR O RESULTADO FINAL (SAÍDA: PRINTF).

```
#include <stdio.h>

int main() {
    int N, soma = 0;
    int i; // Variável contadora para o laço

    printf("Digite um número inteiro positivo (N): ");
    scanf("%d", &N);

    if (N < 1) {
        printf("Por favor, insira um número maior ou igual a 1.\n");
    } else {
        for (i = 1; i <= N; i++) {
            soma = soma + i; // ou soma += i;
        }
        printf("A soma de 1 até %d é: %d\n", N, soma);
    }
    return 0;
}
```

★ POR QUE A LÓGICA DE PROGRAMAÇÃO É TÃO IMPORTANTE? (REVISÃO)

- BASE FUNDAMENTAL: É O ALICERCE PARA QUALQUER DESENVOLVIMENTO NA ÁREA DA COMPUTAÇÃO.
- ESTRUTURA O PENSAMENTO: ENSINA VOCÊ A PENSAR DE FORMA ORGANIZADA, CLARA E EFICIENTE PARA RESOLVER PROBLEMAS.
- HABILIDADE UNIVERSAL: OS CONCEITOS DE LÓGICA SÃO OS MESMOS, NÃO IMPORTA A LINGUAGEM DE PROGRAMAÇÃO QUE VOCÊ USARÁ NO FUTURO.
- ABRE PORTAS: PERMITE QUE VOCÊ CRIE SUAS PRÓPRIAS SOLUÇÕES, AUTOMATIZE TAREFAS E ENTENDA O MUNDO DIGITAL AO SEU REDOR.



ONDE A LÓGICA DE PROGRAMAÇÃO É USADA?

A LÓGICA DE PROGRAMAÇÃO ESTÁ EM TODO LUGAR NO MUNDO DA TECNOLOGIA!

- DESENVOLVIMENTO DE SOFTWARE: CRIAÇÃO DE APLICATIVOS PARA CELULAR, PROGRAMAS DE COMPUTADOR, SISTEMAS PARA EMPRESAS.
- DESENVOLVIMENTO WEB: CONSTRUÇÃO DE SITES E PLATAFORMAS ONLINE.
- JOGOS DIGITAIS: DESDE A MECÂNICA DO PERSONAGEM ATÉ AS REGRAS DO JOGO.
- AUTOMAÇÃO DE PROCESSOS: ROBÔS QUE REALIZAM TAREFAS REPETITIVAS.
- INTELIGÊNCIA ARTIFICIAL (IA): ALGORITMOS QUE PERMITEM QUE MÁQUINAS "APRENDAM" E TOMEM DECISÕES.
- ANÁLISE DE DADOS: ENCONTRAR INFORMAÇÕES IMPORTANTES EM GRANDES VOLUMES DE DADOS.
- ...E MUITO MAIS!



DESAFIO DE REVISÃO

CRIE UM PROGRAMA EM LINGUAGEM C QUE:

- LEIA O NOME E DUAS NOTAS DE 3 ESTUDANTES
- CALCULE A MÉDIA DE CADA ESTUDANTE
- MOSTRE SE ELE FOI APROVADO ($\text{MÉDIA} \geqslant 7$) OU REPROVADO

O PROGRAMA DEVE USAR:

- UM LAÇO DE REPETIÇÃO
- PELO MENOS UMA ESTRUTURA CONDICIONAL
- PREFERENCIALMENTE UMA FUNÇÃO PARA CALCULAR A MÉDIA

 DÚVIDAS? PERGUNTAS? IDEIAS?



SUA JORNADA TECH CONTINUA!

ESTA FOI APENAS A PRIMEIRA OFICINA! VEJA O QUE MAIS TEREMOS:

- OFICINA 2: DESENVOLVIMENTO WEB BÁSICO HTML, CSS, ESTRUTURA DE PÁGINAS, CONSTRUÇÃO DE UM SITE PESSOAL.
- OFICINA 3: INTRODUÇÃO AO ARDUINO CIRCUITOS SIMPLES, COMPONENTES, PROTOBOARD E PRIMEIROS PASSOS COM ARDUINO.
- OFICINA 4: CARREIRA EM ENGENHARIA DA COMPUTAÇÃO ATUAÇÃO PROFISSIONAL, MERCADO, ÁREAS DA ENGENHARIA, BATE-PAPO COM ENGENHEIRO.

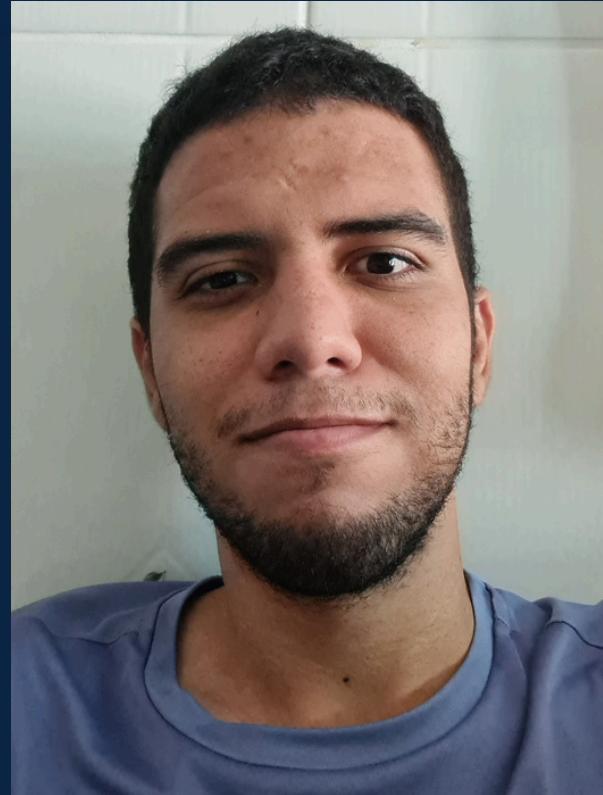
EQUIPE



ANA BEATRIZ SCHUINDT



NAMEM RACHID JAUDY



CARLOS MIGUEL
OLIVEIRA



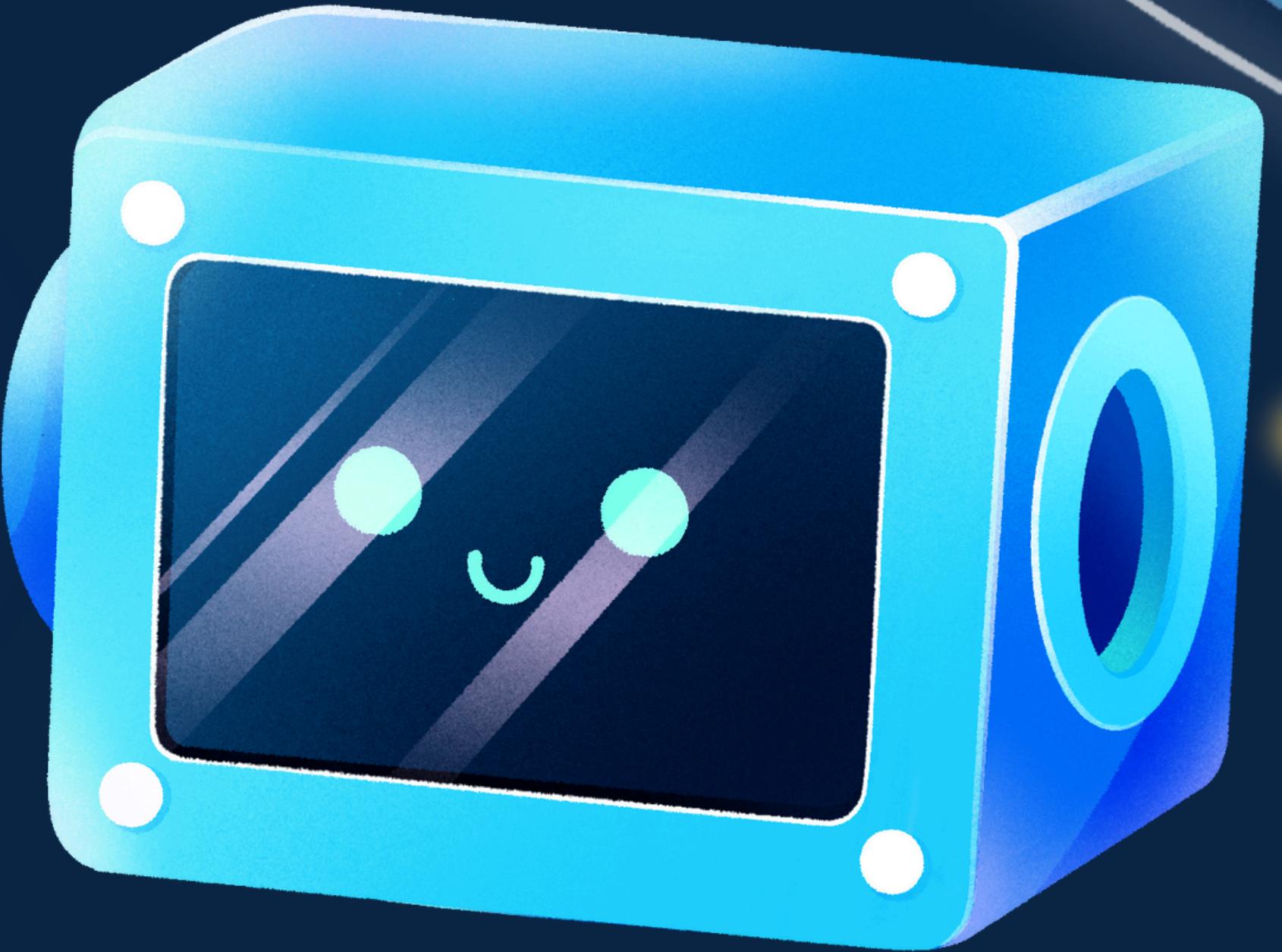
LEONARDO ANTUNES

Fab
Lab

OBRIGADA!

Alguma dúvida?

Vamos trocar ideias e testar
algumas coisas!



JORNADA TECH



EQUIPE



ANA BEATRIZ SCHUINDT



NAMEM RACHID JAUDY



CARLOS MIGUEL
OLIVEIRA



LEONARDO ANTUNES

O QUE VAMOS APRENDER?

O QUE É A INTERNET E COMO ELA FUNCIONA?



HTML - QUE TRABALHA A ESTRUTURA DE UM SITE

LINGUAGENS UTILIZADAS PARA DESENVOLVIMENTO WEB



CSS - QUE Torna o ESTILO DO SITE BONITO



FERRAMENTAS NECESSÁRIAS PARA COMEÇAR A DESENVOLVER



JAVASCRIPT (JS) COMO DEIXAR SEU SITE INTERATIVO

Fab
Lab

SERÁ QUE EU DOU CONTA?

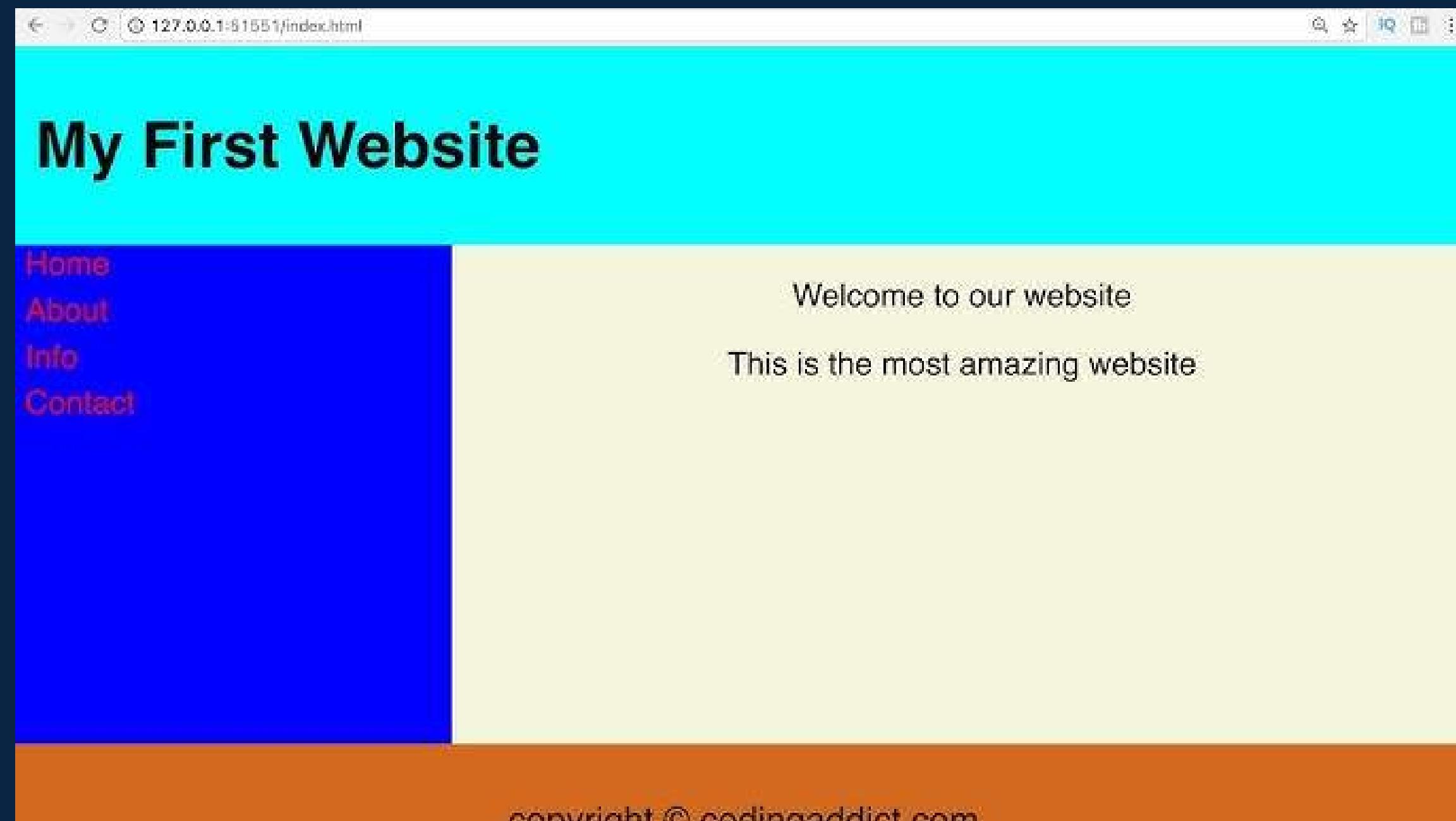
BEBÊS
ANTIGAMENTE:
'GUGU-DADA?'



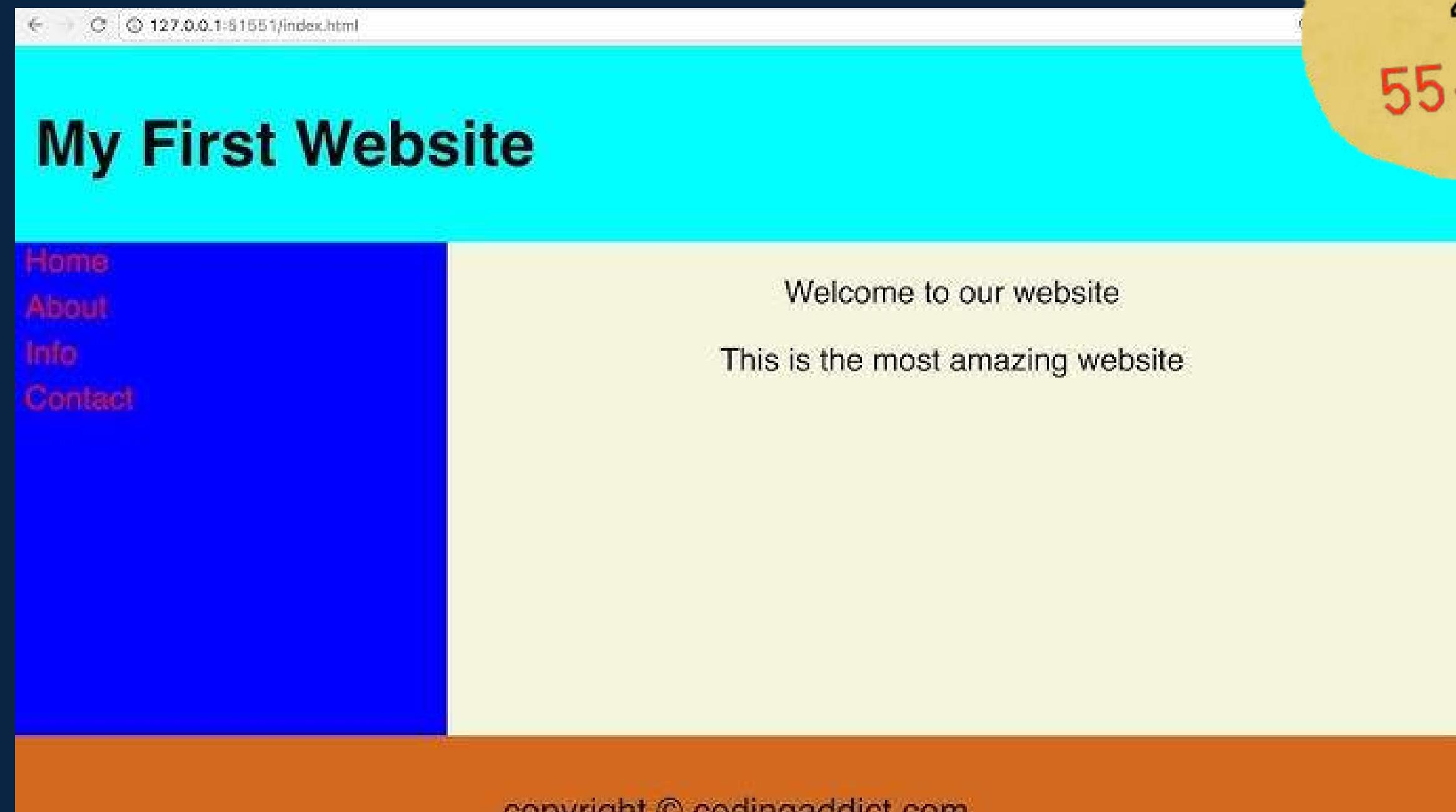
BEBÊS
ANTIGAMENTE:
'GUGU <H1>?'



SERÁ QUE EU DOU CONTA?



SERÁ QUE EU DOU CONTA?



30-40(HTML) +
25-30(CSS)

55-70 LINHAS!

CONTEÚDOS LEGAIS

NETFLIX

Inicio Séries Filmes Jogos Bombando Minha lista Navegar por idiomas

Descubra suas próximas histórias

TOP 10 THE ROOKIE Novidade

TOP 10 HOMEM COM H Novidade

TOP 10 OLÍMPO Novidade

TOP 10 N A ULTIMA DOTA Novidade

TOP 10 N SANMAN Novidade

TOP 10 N GINNY E GEORGIA Nova temporada

Brasil: top 10 em filmes hoje

1 GUERRERAS DO K-POP Novidade

2 HOMEM COM H Novidade

3 DESASTRE TOTAL CRUZEIRO DO CÓDOR Novidade

4 ULTIMA DOTA

A14 1h 47min HD Áudio Especial

Sombrio • Suspense no ar • Contra o sistema

5 SOMA DOS MEDOS

Brasil: top 10 em séries hoje

1 O PÍER

2 RICO MAS Pobre

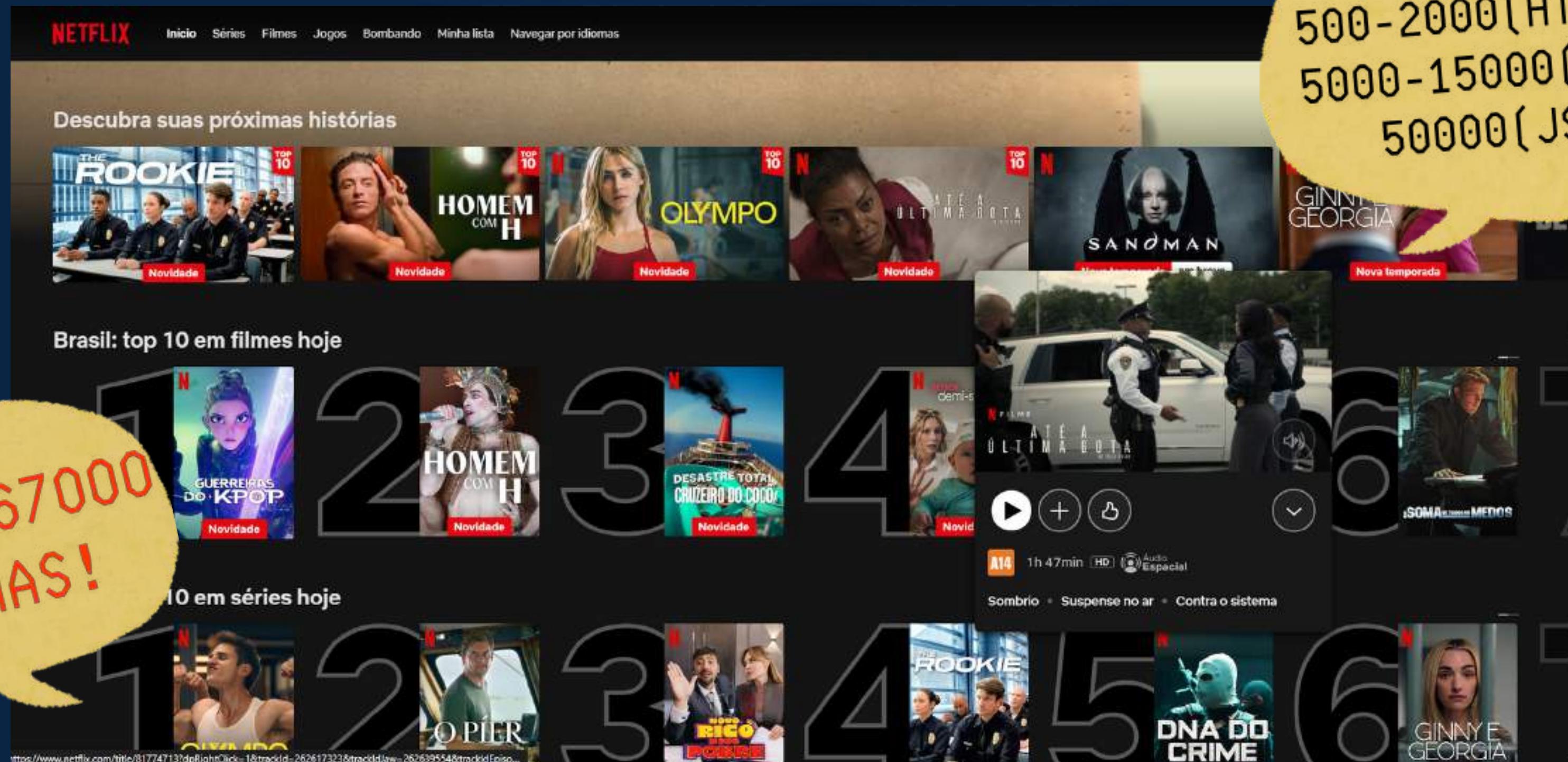
3 THE ROOKIE

4 DNA DO CRIME

5 GINNY E GEORGIA

<https://www.netflix.com/title/81774713?dpRightClick=1&trackId=262617323&trackIdLow=262639554&trackIdEpi...>

CONTEÚDOS LEGAIS



Fab
Lab

**Sua[®]
Logo**

[PRINCIPAL](#) [PÁGINA 1](#) [PÁGINA 2](#) [PÁGINA 3](#) [PÁGINA 4](#) [CONTATO](#)

Título na página inicial

Lorèm ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritas et quasi architecto beatis vite dicta sunt explicata. Nemo enim ipsam voluptatem quia voluptas sit aspernare aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatum sequi resipiunt. Neque porro quisquam est, qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit; sed quia non numquam eius modi tempora incidunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim ad minimam veniam, quis nostrud exercitationem ullam corporis suscipit laboriosam, nisi ut aliquid ex ea commodi consequat? Quis autem vel eum llore reprehendit, qui in ea voluptate velit esse quam nihil molestiae consequatur, vel illum qui dolorem eum fugiat, quo voluptas nulla pariatur?

[Título Link](#)

At vero eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque conseptus dolores et quas molestias excepturi sint occaecati cupiditate non provident, similique sunt in culpa qui officia deserunt mollitia animi, id est laborum et dolorum fuga.

TODOS OS DIREITOS RESERVADOS
Endereço • Telefone

www.programaria.org/cursos/euprograma/

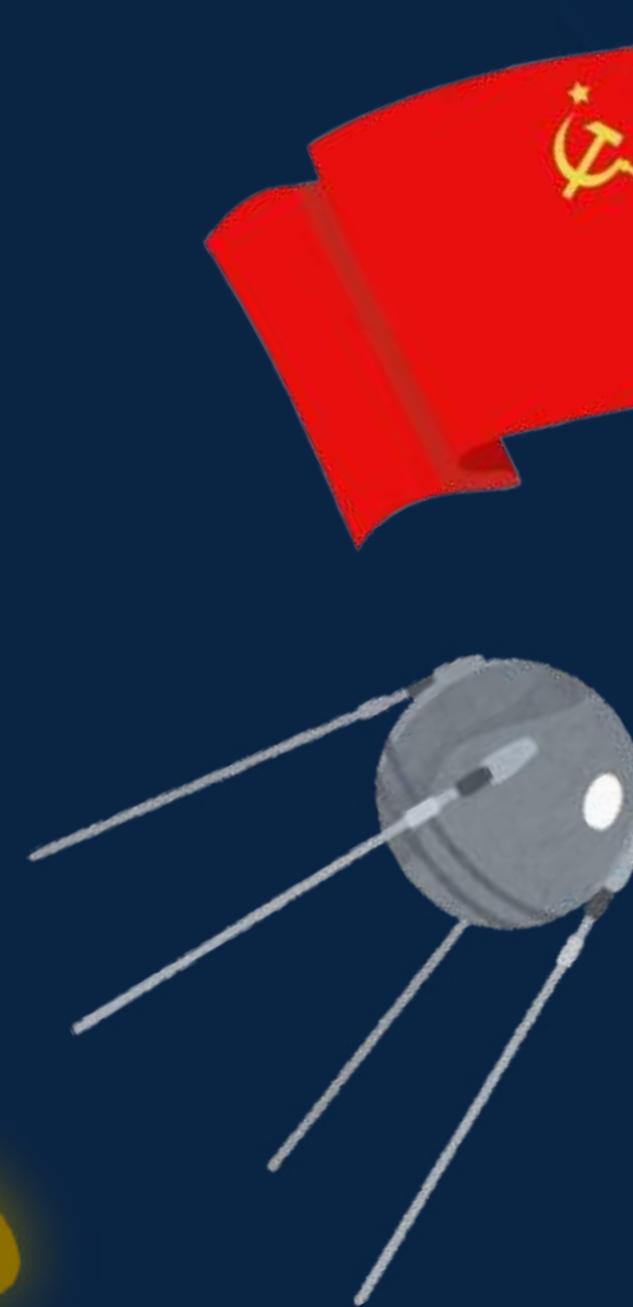
programaria

A missão

Desbrave o universo dos códigos com este curso de introdução à programação com foco em desenvolvimento web. Nele, você vai dar o primeiro passo da sua jornada! Vai aprender conceitos básicos de programação, fazer páginas web e ter noções de como escrever programas para resolver problemas reais. Para embarcar com a gente, você precisa de apenas um pré-requisito: vontade de aprender!

Durante o curso, você planejará, desenvolverá e publicará seu primeiro website!

COMO A INTERNET SURGIU?



SPUTNIK
1957



GUERRA FRIA
USA X URSS
1949



1958

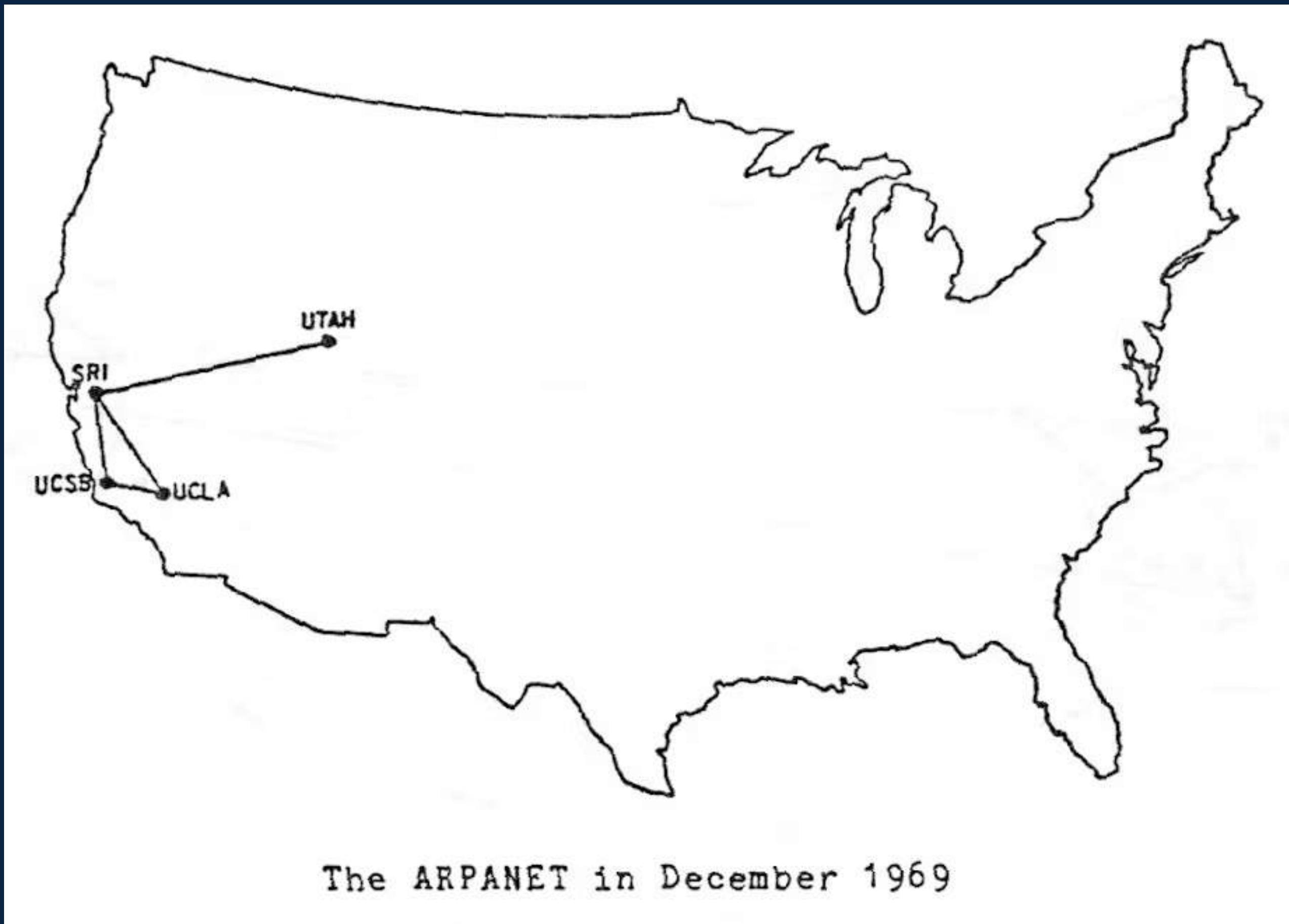
Fab
Lab]

O QUE É A INTERNET?

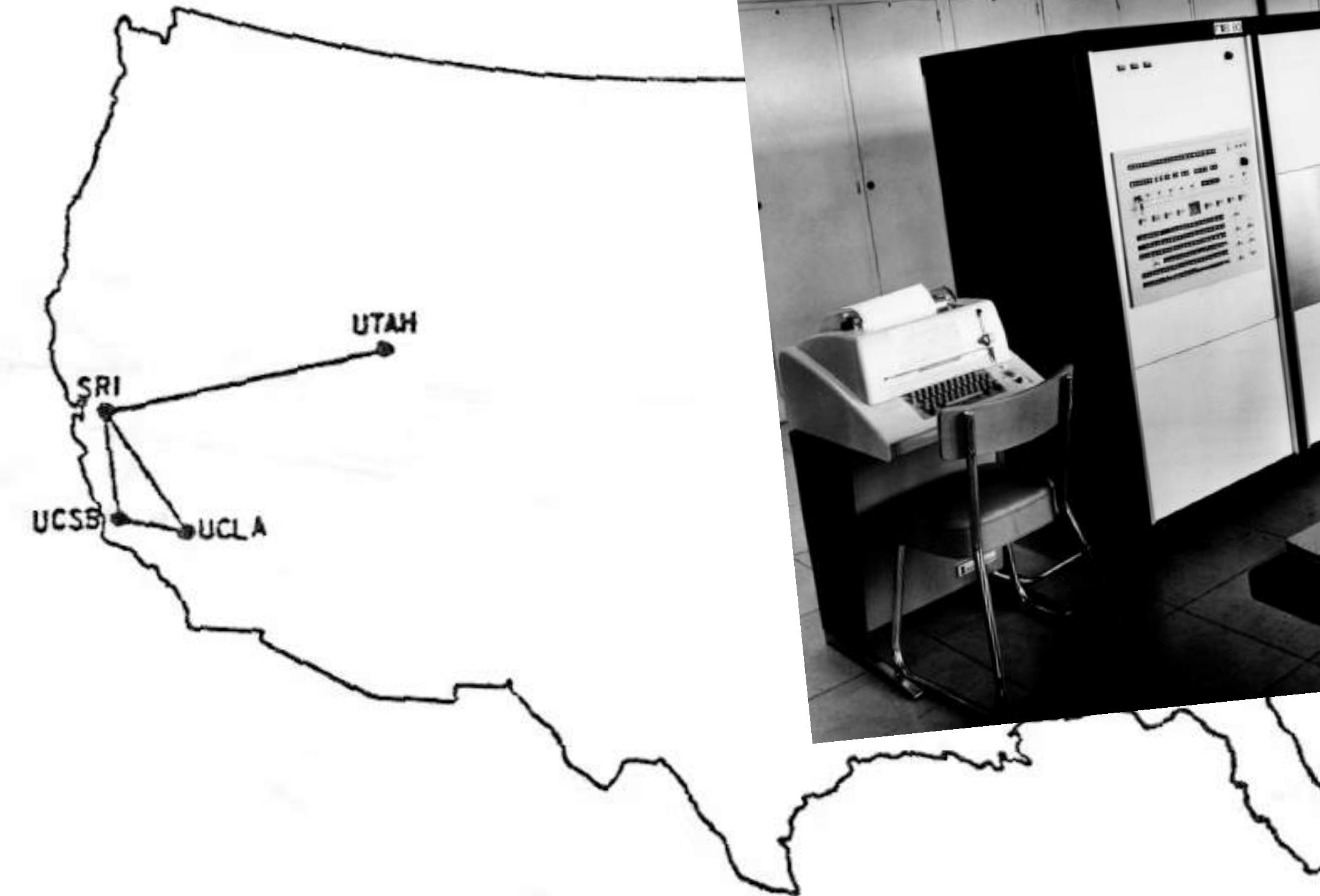
1969



Fab
Lab]



Fab
Lab]

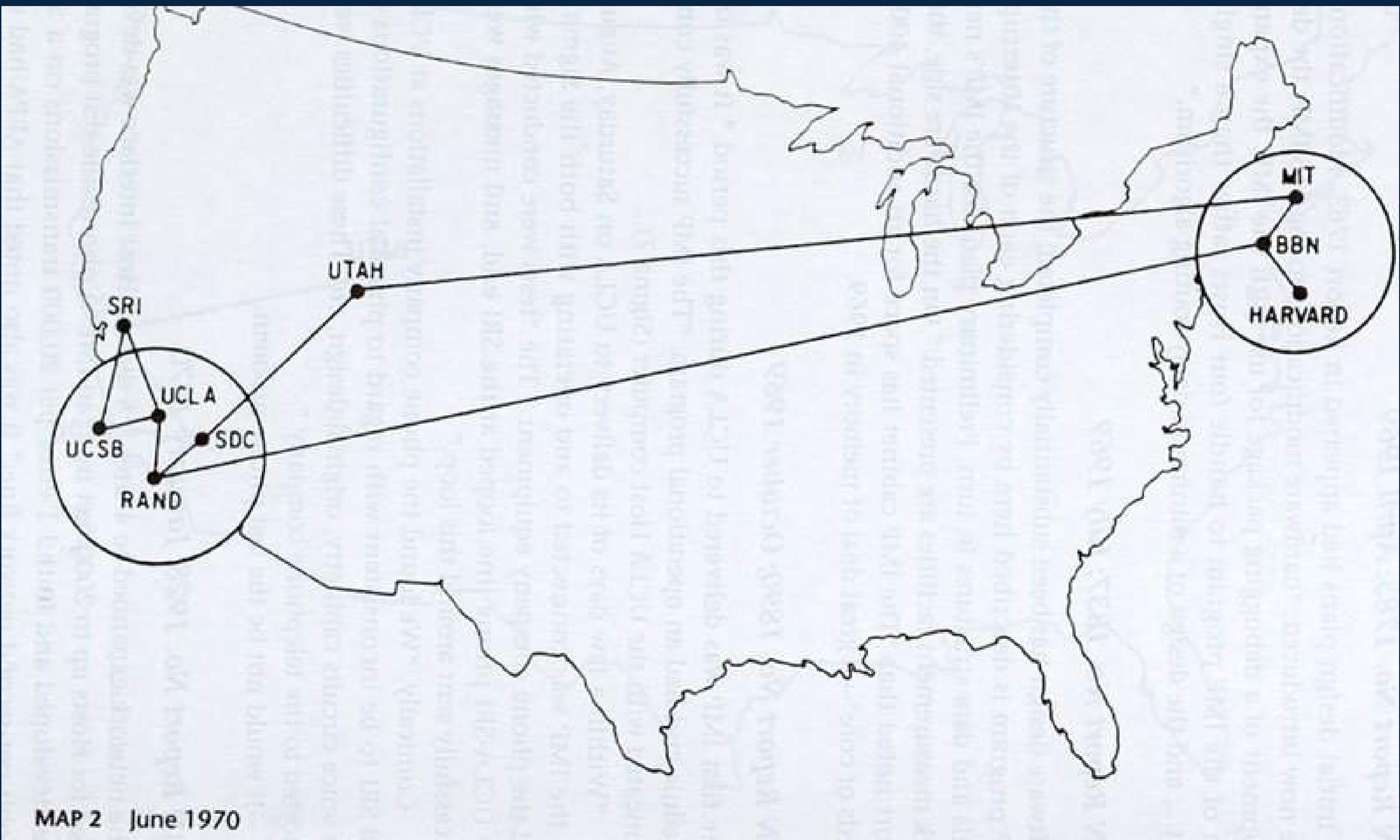


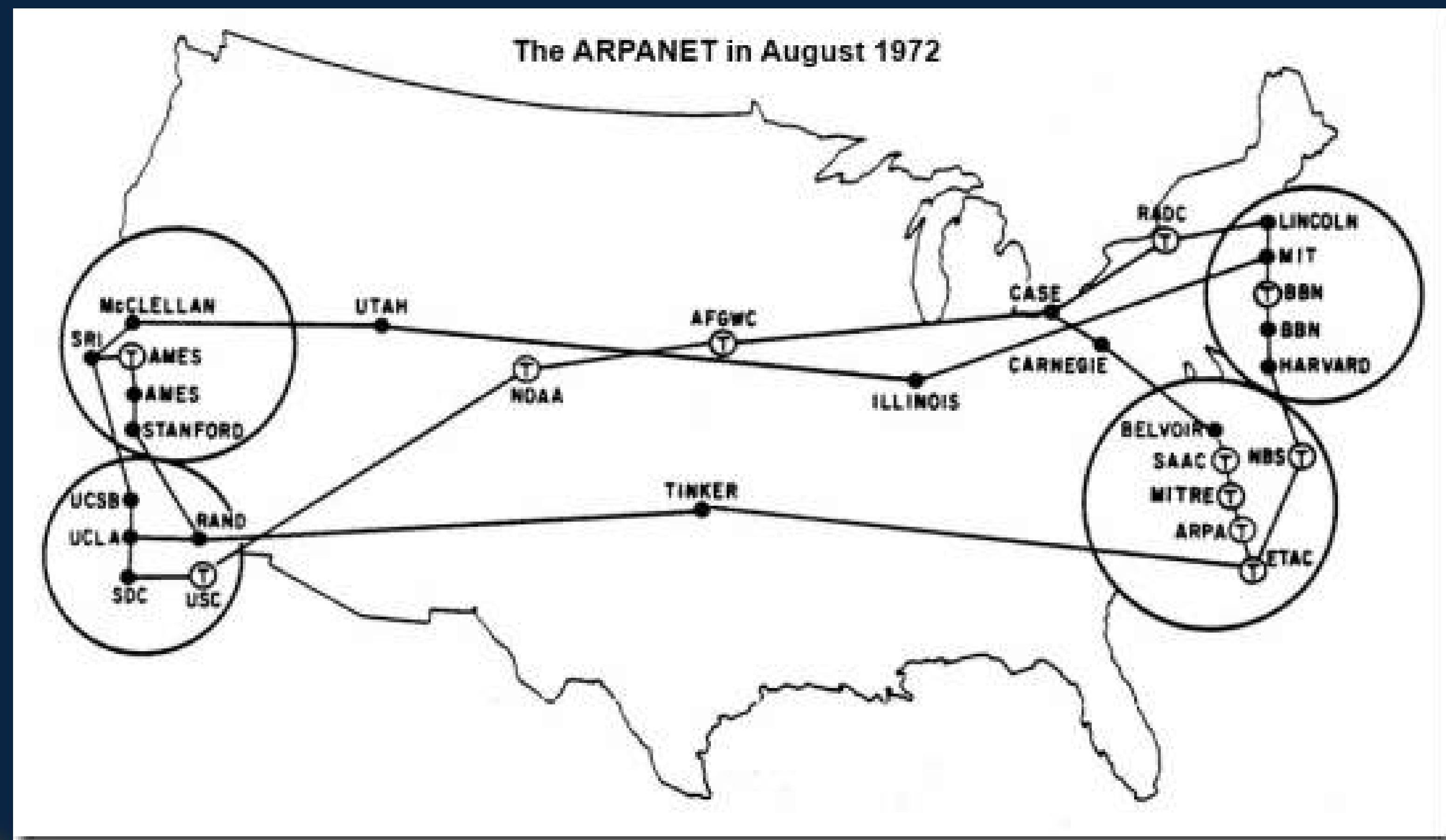
The ARPANET in December 1969



SDS SIGMA 7

Fab
Lab]





COMO A INTERNET FUNCIONA?



+



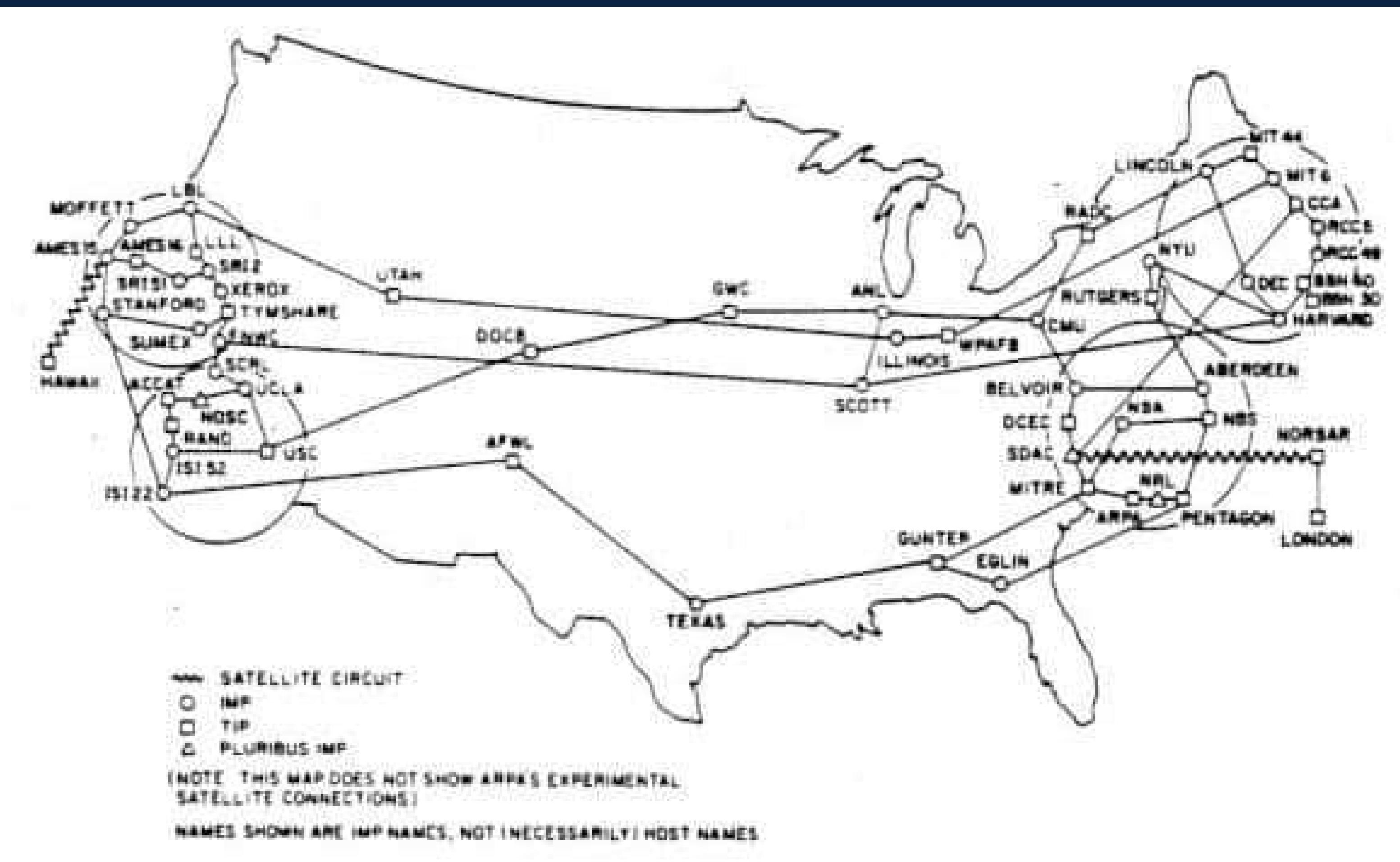
=

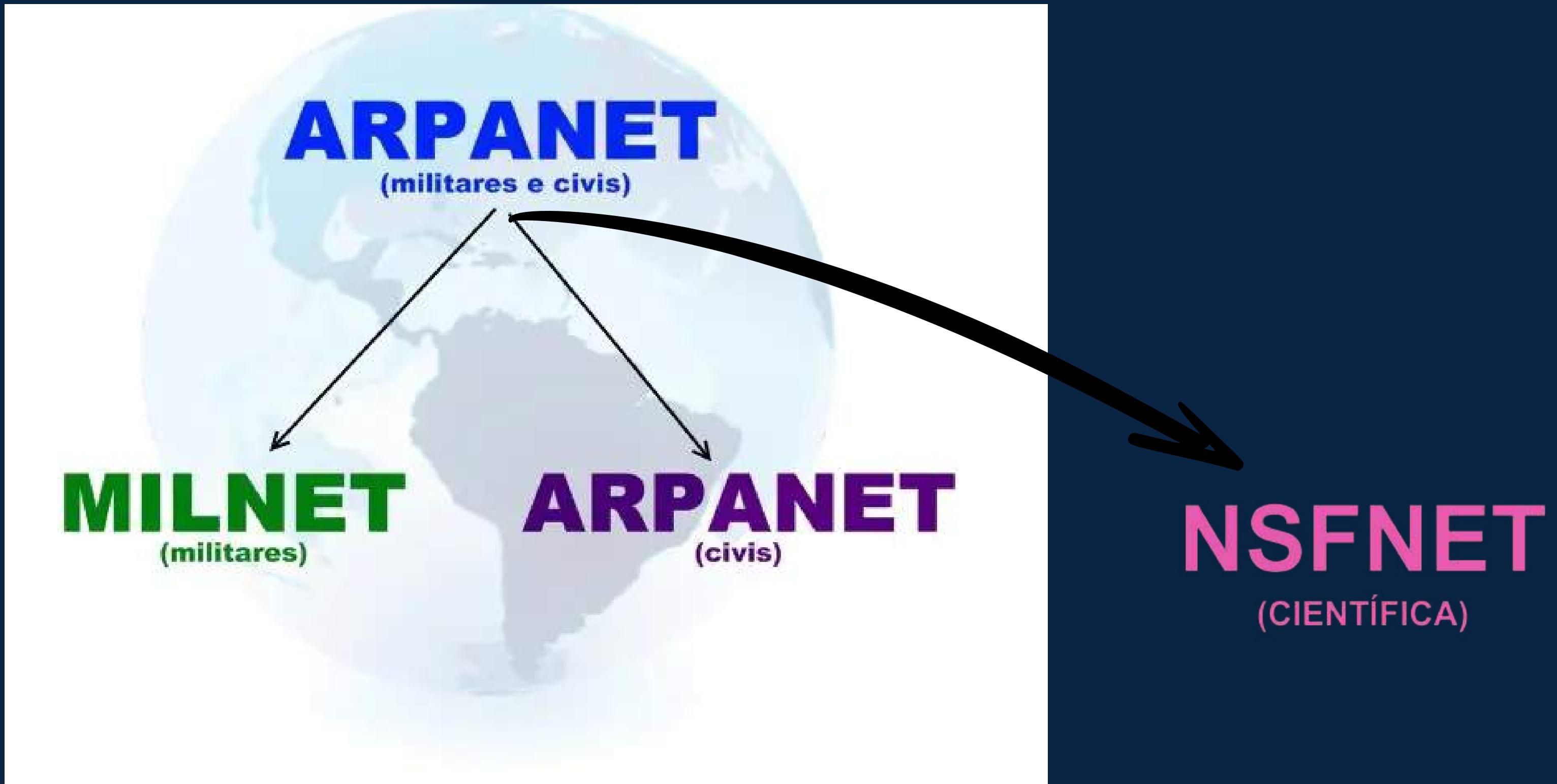


BOB KAHN
TCP
(SEM ESPERA)

VINT CERF
IP
(ENDEREÇAMENTO)

Fab
Lab]





INTERCONNECT NETWORKING



INTERNETWORKING



CONJUNTO DE
MUUIITAS
REDES!

INTERNET!



Fab
Lab

COMO A INTERNET FUNCIONA?



[SETE MAPAS INCRÍVEIS DA INTERNET](#) |

[BLOG OFICIAL DA KASPERSKY](#)

COMO A INTERNET FUNCIONA?



CLIENTE
(VOCÊ)



SERVIDOR
YOUTUBE

Fab
Lab J

COMO A INTERNET FUNCIONA?



COMO A INTERNET FUNCIONA?

<https://info.cern.ch>



TIM BERNERS-LEE

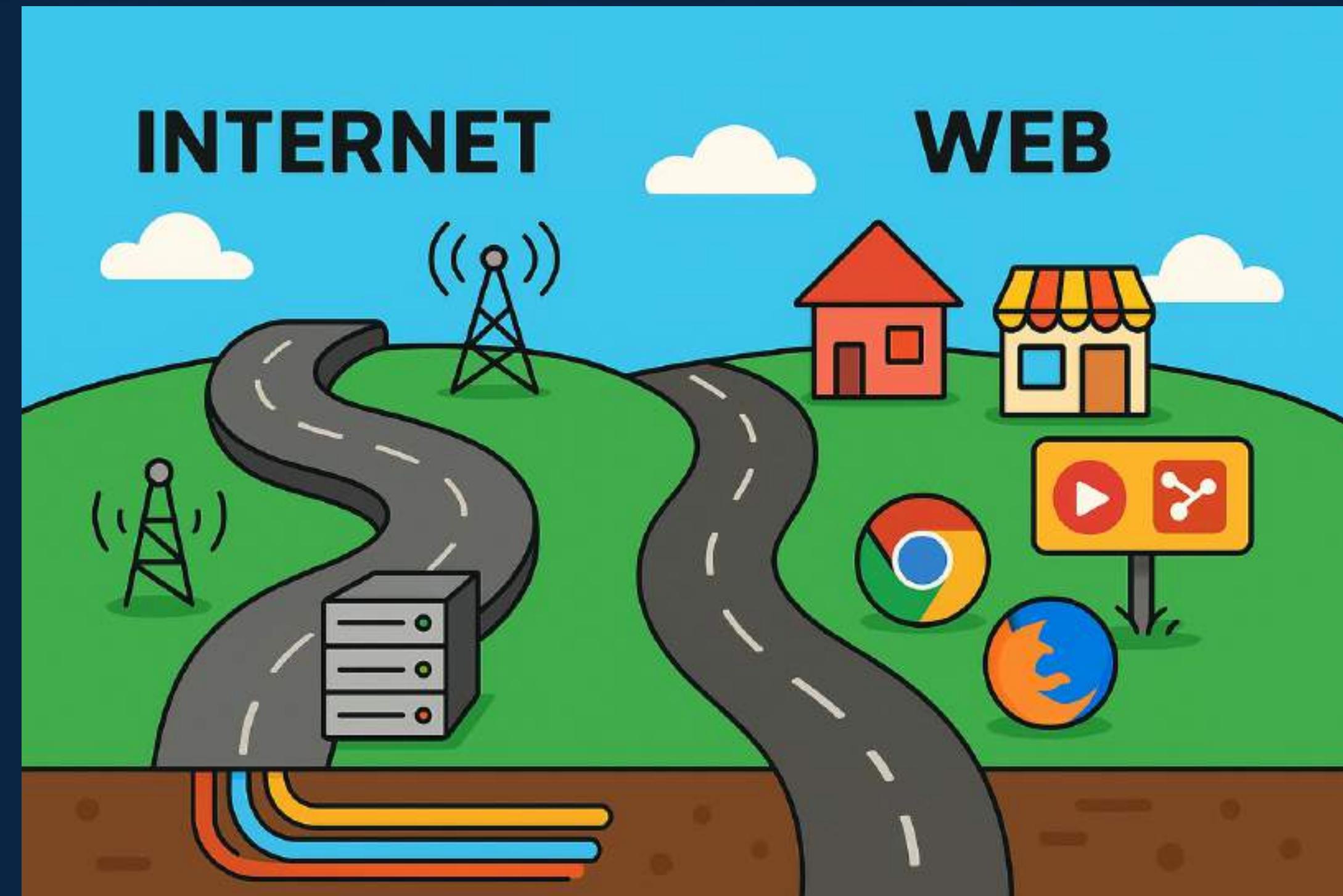


COMO A INTERNET FUNCIONA?



NAVEGADORES WEB

COMO A INTERNET FUNCIONA?



COMO A WEB FUNCIONA? O QUE É UMA URL?



A URL É O ENDERECO DE UMA
PÁGINA NA WEB!!!

O QUE É UMA PÁGINA WEB?

UMA PÁGINA WEB É COMO UMA CASA:

- O HTML CONSTRÓI A BASE E AS PAREDES (CONTEÚDO E ESTRUTURA)
- O CSS PINTA, ORGANIZA E DEIXA BONITO (ESTILO)
- O JAVASCRIPT LIGA A CAMPAINHA, ACENDE A LUZ, ETC (INTERATIVIDADE)

HTML



ESTRUTURA COM
TEXTO E IMAGENS

CSS



CORES E LAYOUT

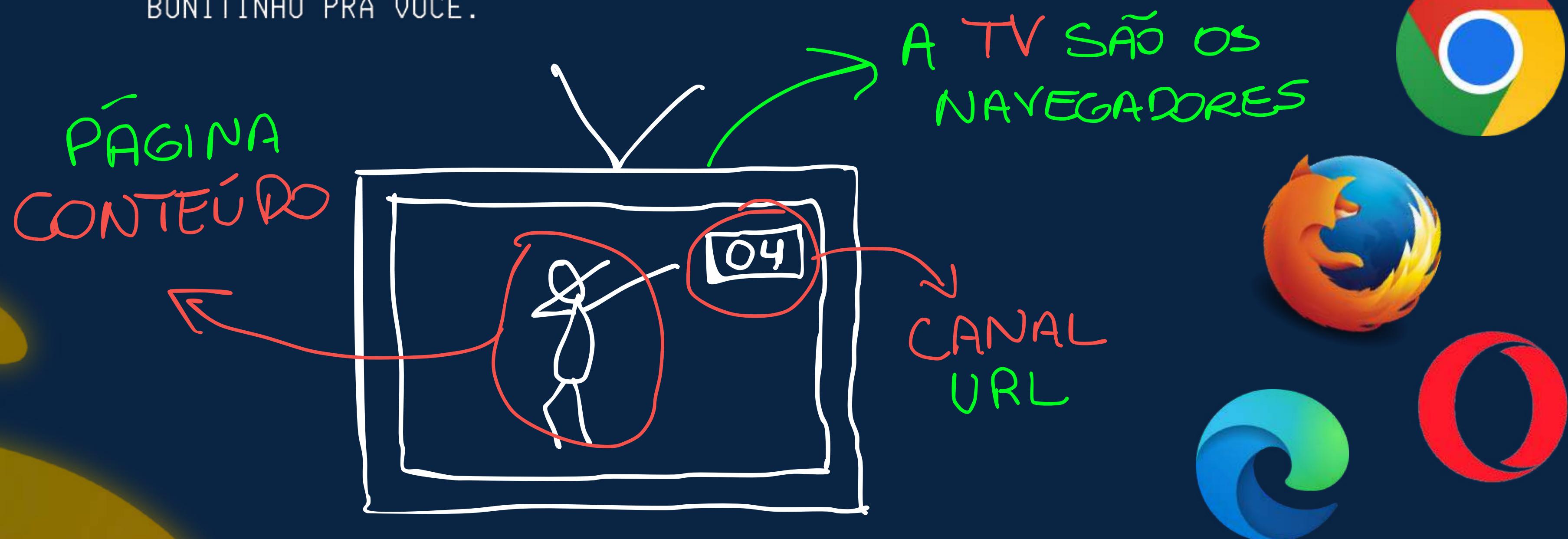
JavaScript



BOTÃO QUE APARECE
OU REAGE

O QUE É UM NAVEGADOR WEB?

- O NAVEGADOR (BROWSER) É UM PROGRAMA QUE ACESSA E EXIBE PÁGINAS DA WEB.
- ELE “FALA” COM OS SERVIDORES, ENTENDE HTML, CSS E JS, E MOSTRA TUDO BONITINHO PRA VOCÊ.



FERRAMENTAS DE DESENVOLVEDOR (DEVTOOLS)

- FERRAMENTAS DO NAVEGADOR PARA VER E EDITAR O CÓDIGO DE UM SITE
- ACESSE COM F12 OU BOTÃO DIREITO → “INSPECIONAR”
- VOCÊ PODE:
 - VER HTML E CSS DO SITE
 - TROCAR TEXTOS E CORES AO VIVO
 - SIMULAR CELULARES
 - VER ERROS NO CÓDIGO



FERRAMENTAS DE DESENVOLVEDOR (DEVTOOLS)

AGORA É SUA VEZ! EXPLORE UM SITE COM DEVTOOLS



- TROCAR UM TEXTO
- MUDAR UMA COR
- BRINCAR COM O LAYOUT

FERRAMENTAS



VSCODE (VISUAL
STUDIO CODE)

DENTRO DO VSCODE



EXTENSÃO LIVE
SERVER



UM NAVEGADOR DA
SUA ESCOLHA



O QUE É O HTML?

```
<h1>h1</h1>
<h2>h1</h2>
<h3>h1</h3>
<h4>h1</h4>
<h5>h1</h5>
<h6>h1</h6>
<p>p</p>
```

h1

h1

h1

h1

h1

h1

p

- HTML = HYPERTEXT MARKUP LANGUAGE
- É A ESTRUTURA DE UMA PÁGINA WEB
- DEFINE O QUE VAI APARECER NA TELA
(TEXTOS, IMAGENS, LINKS, LISTAS...)



Fab
Lab

Editar HTML

Conteúdo Avançado

Código HTML

Código HTML Code with AI

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>Título da Página
6   </title>
7   <link rel="stylesheet"
8     href="styles.css">
9   <script src="script.js"
10    ></script>
11 </head>
12 <body>
13   <header>
14     <nav>
15       <ul>
16         <li><a href="#">
17           Página Inicial
18         </a></li>
19         <li><a href="#">
20           Sobre</a></li>
21         <li><a href="#">
22           Contato</a>
23         </li>
24       </ul>
25     </nav>
26   </header>
27   <main>
28     <h1>Título Principal
29     </h1>
30     <p>Este é um parágrafo
31       de exemplo.</p>
32     
35     <a href="#">Clique aqui
36       </a> para saber
37       mais.
38   </main>
```

• Página Inicial
• Sobre
• Contato

Título Principal

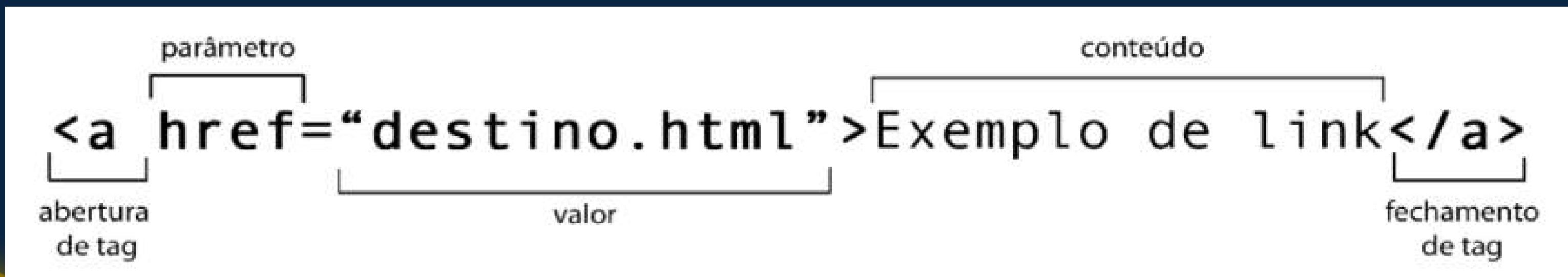
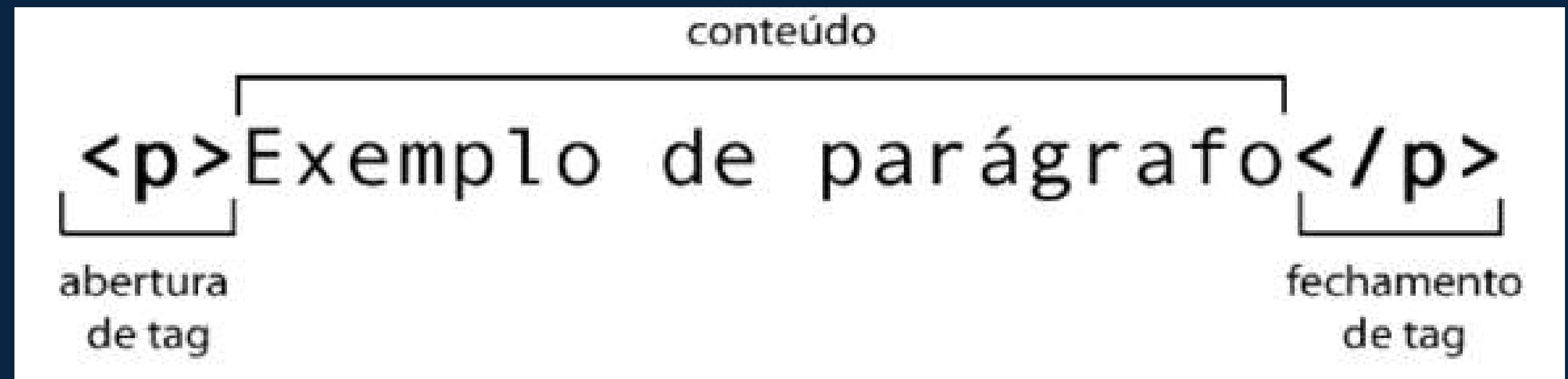
Este é um parágrafo de exemplo.

Descrição da imagem [Clique aqui](#) para saber mais.
© 2023 Nome da Empresa. Todos os direitos reservados.

Solte o widget aqui

o que você procura?

ANATOMIA DE UMA TAG HTML



TAGS <HTML>: OS H

ESTRUTURA DE HEADING TAGS DE UMA PÁGINA

H1: Título do texto (único)

H2: Primeiro tópico do assunto

H3: Um tópico dentro do H2

H4: Subtópico dentro do H3

H5: Subtópico dentro do H4

H6: Dentro do subtópico do subtópico H5

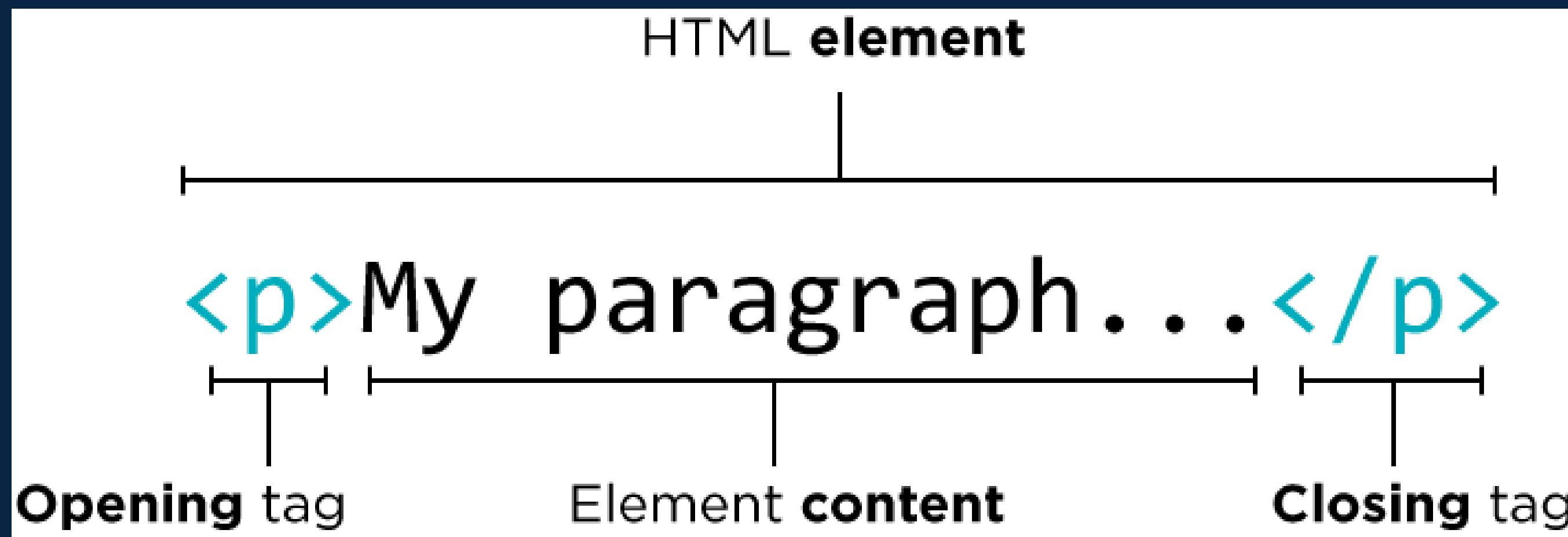
H3: Outro tópico dentro do H2

H2: Segundo tópico do assunto

H3: Um tópico dentro do H2

H2: Terceiro tópico do assunto

TAGS <HTML>: 0 <P>



TAGS <HTML>: 0 <A>

ESTRUTURA DE UMA TAG HTML

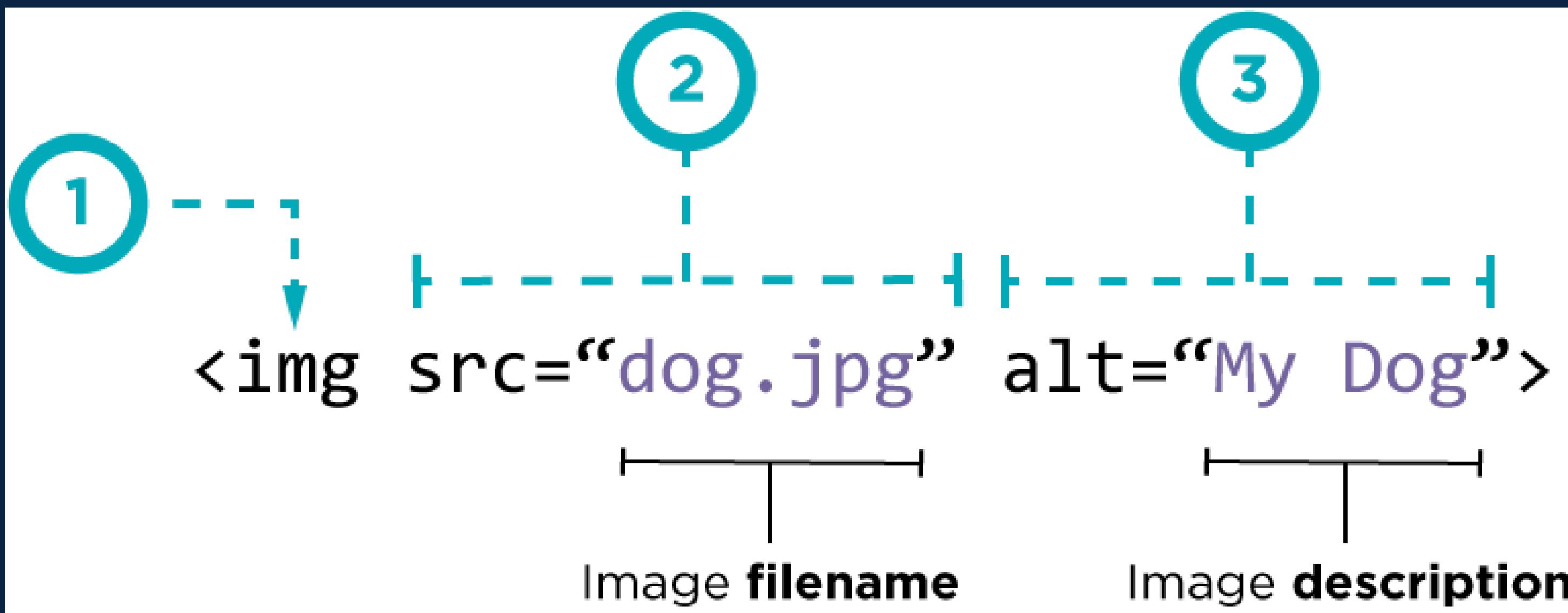
ABERTURA ATRIBUTO

VALOR

GCFAprendeLivre } CONTEÚDO

FECHAMENTO

TAGS <HTML>: 0



TAGS <HTML>: LISTAS

HTML Demo:

RESET OUTPUT

Title of the document.

https://www.w3docs.com

Ordered list

1. Appetizers
2. Main Course
3. Salads

Unordered list

- Cold Drinks
- Hot Drinks
- Ice-Creams

HTML

CSS

```
2 <li>Item 1</li>
3 <li>Item 2
4   <ul>
5     <li>Subitem 2.1</li>
6     <li>Subitem 2.2</li>
7   </ul>
8 </li>
9 <li>Item 3
10  <ol>
11    <li>Subitem 3.1</li>
12    <li>Subitem 3.2</li>
13  </ol>
14 </li>
15 </ul>
16
```

• Item 1
 • Item 2
 • Subitem 2.1
 • Subitem 2.2
 • Item 3
 1. Subitem 3.1
 2. Subitem 3.2

TAGS <HTML>: A TAG <DIV>

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>DivExample</title>
</head>

<div class="header">
    <h1>My Website</h1>
</div>

<div class="main-content">
    <p>This is the main content of the page.</p>
</div>

<div class="footer">
    <p>Footer information goes here.</p>
</div>
</html>
```

My Website

This is the main content of the page.

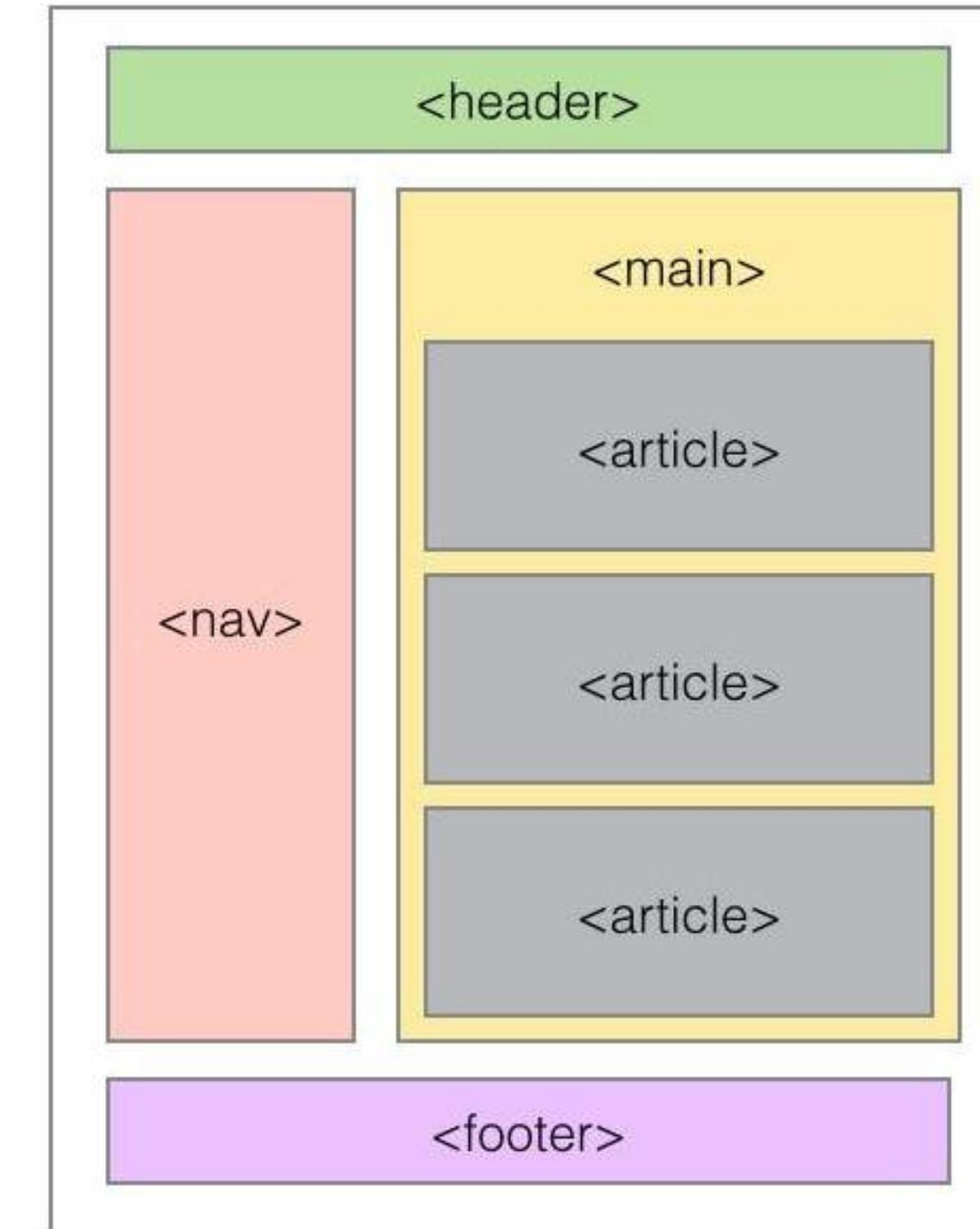
Footer information goes here.

CONVENÇÃO DO HTML

CASA AS TAGS).

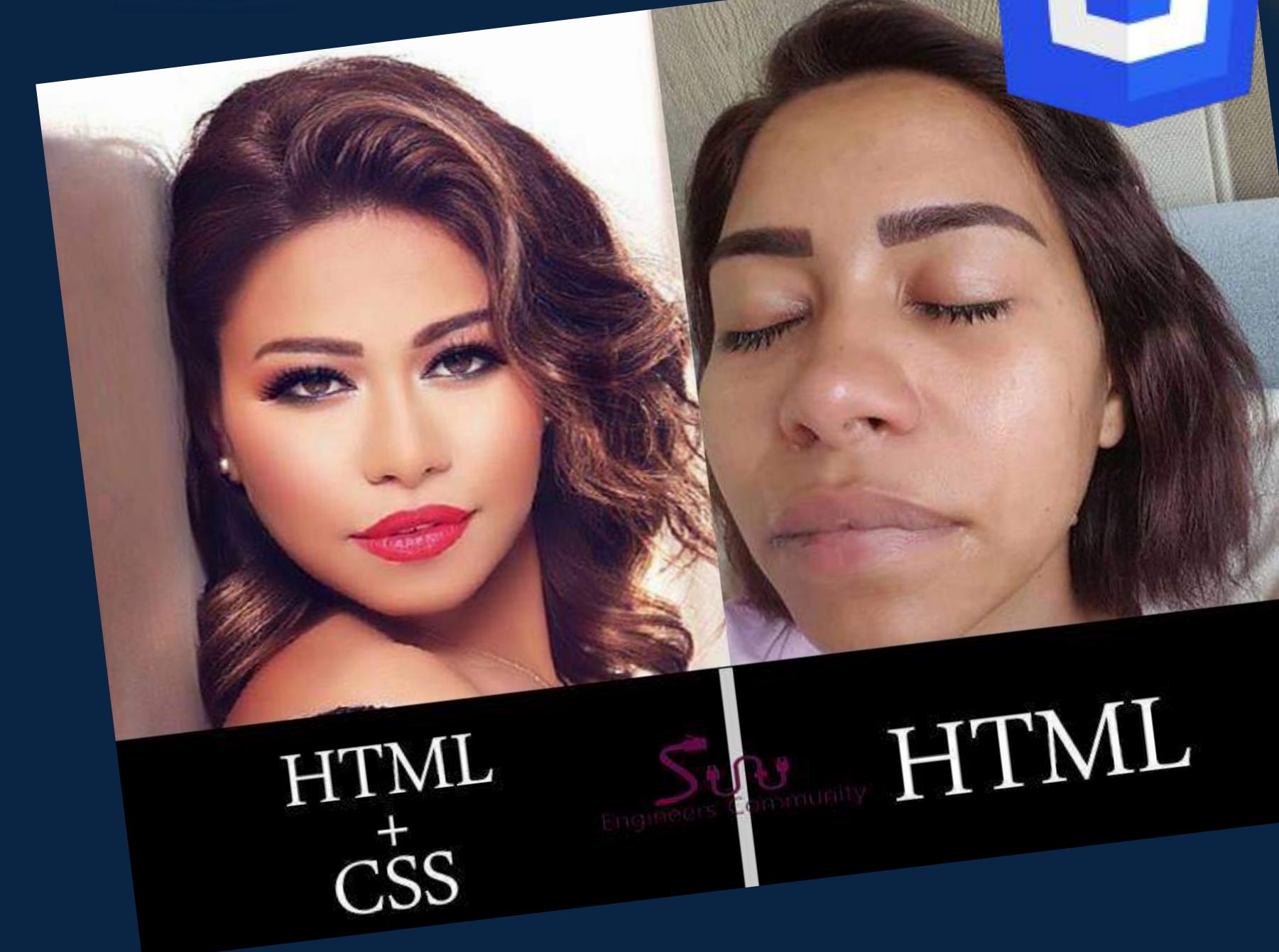
HTTPS://WWW.
SMASHINGMAGAZINE.COM/

HTML5: Semantic Tags/Sections



O QUE É O CSS?

- CSS = CASCADING STYLE SHEETS (FOLHAS DE ESTILO EM CASCATA)
- DEFINE COMO OS ELEMENTOS HTML VÃO PARECER (CORES, TAMANHOS, LAYOUT...)



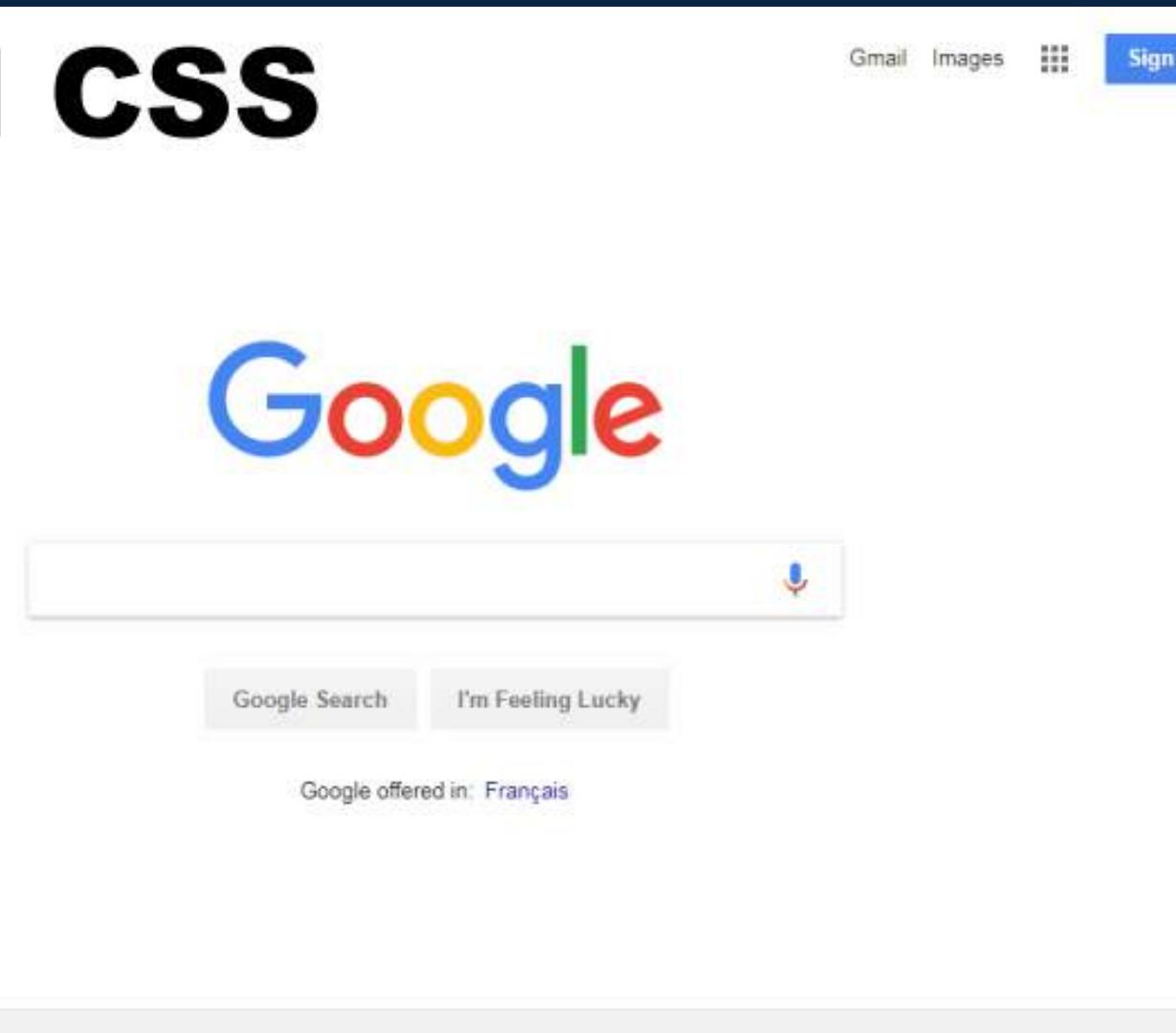
HTML
+
CSS

Sun
Engineers Community

HTML

O QUE É O CSS?

WITH CSS



The screenshot shows the Google homepage with a clean, modern design. It features the iconic Google logo at the top left. Below it is a large, white search bar with a microphone icon for voice search. Underneath the search bar are two buttons: "Google Search" and "I'm Feeling Lucky". A small note indicates that the site is offered in French ("Google offered in: Français"). At the bottom of the page, there's a navigation bar with links for "France", "Advertising", "Business", "About", "Consumer Information", "Privacy", "Terms", and "Settings".

WITHOUT CSS



The screenshot shows the same Google homepage but without CSS. The layout is much more basic and lacks visual structure. The Google logo is present but lacks its characteristic color gradient. The search bar is a simple white rectangle. The "Google Search" and "I'm Feeling Lucky" buttons are also less distinct. The language note at the bottom is smaller. The overall appearance is flat and lacks the polished look provided by CSS.

COMO APLICAR O CSS?

```
<p style="color:red;">  
Paragraph</p>
```

```
<head>  
<style>  
    p {color:red;}  
</style>  
</head>
```

```
<head>  
    <link rel="stylesheet" href="styles.css">  
</head>
```

COMO APLICAR O CSS?

Inline CSS

```
<p style="color: blue;">This is a paragraph.</p>
```



A screenshot of a mobile browser window. At the top, there are three colored dots (red, yellow, green) representing the browser's navigation buttons. Below them, the text "Inline CSS Styles" is displayed in a light gray font. Underneath this, there is a sample of HTML code with inline CSS styling applied to an

element and a element. ``` <!-- Inline CSS Styles --> <h1 style="font-weight: bold; font-size: 42px;"> I will be bold and 42 pixels! </h1> <p style="color: blue;">I will be blue!</p> ```

COMO APLICAR O CSS?

Internal CSS

```
<head>
  <style type = text/css>
    body {background-color: blue;}
    p { color: yellow;}
  </style>
</head>
```

```
<head>
  <style>
    h1 {
      color: red;
      font-size: 40px;
    }
    body {
      background-color: salmon;
    }
  </style>
</head>
```

COMO APLICAR O CSS?

External CSS

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

The screenshot shows a code editor with two files open. On the left is the `index.html` file, which contains the following code:

```
<? index.html 3 >
<? index.html > <html> <body> <p>
1   <html>
2   <head>
3     <link rel="stylesheet" type="text/css" href="main.css">
4   </head>
5   <body>
6     <h1>DataFlair</h1>
7     <p>Welcome to DataFlair.</p>
8   </body>
9   </html>
10
```

On the right is the `main.css` file, which contains the following CSS rules:

```
# main.css > ...
# main.css > ...
1 h1 {
2   color: cyan;
3   font-size: 36px;
4 }
5 p {
6   font-family: Arial;
7 }
```

The screenshot shows a browser window displaying the result of the CSS application. The page features a large cyan-colored `h1` heading "DataFlair" and a smaller black `p` tag "Welcome to DataFlair." The browser interface includes tabs for `index.html`, `main.css`, and a preview area.

DataFlair

Welcome to DataFlair.

SELETORES CSS

Seletor de elemento

Seleciona elementos HTML

Com base no nome desses elementos

`p { ... }`

Seletor de id

Seleciona um elemento HTML específico

Com base no atributo id desse elemento

`#id { ... }`

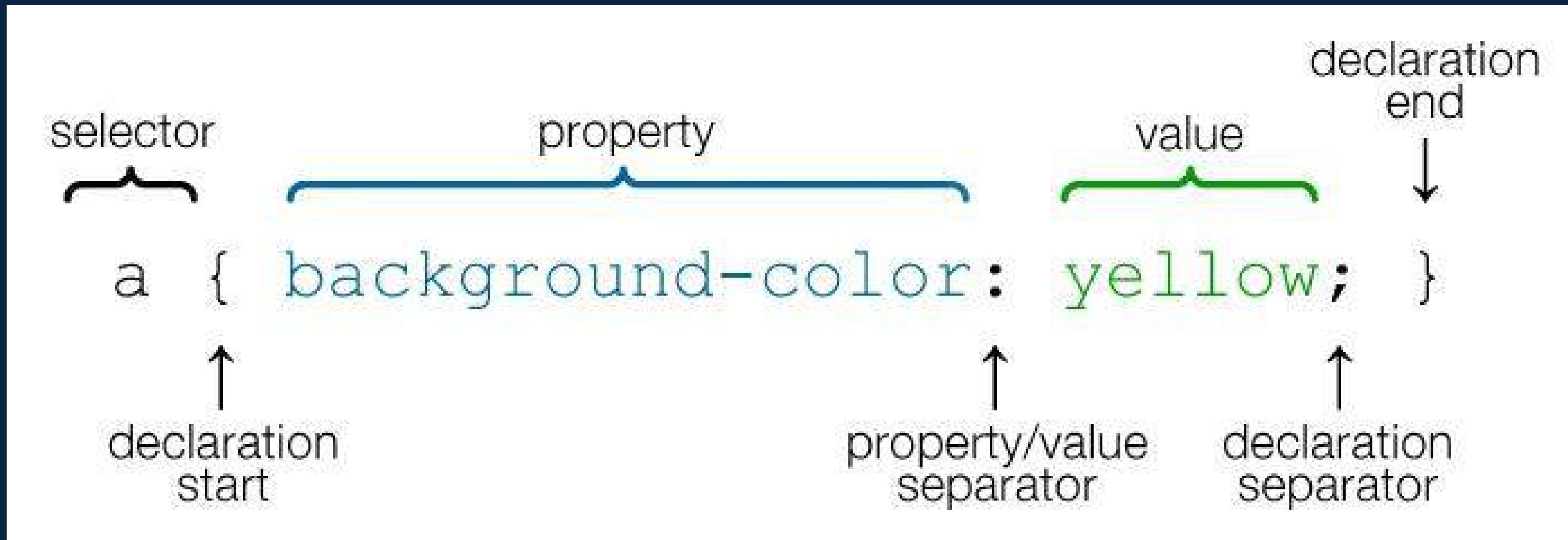
Seletor de classe

Seleciona elementos HTML

Com base no atributo de classe

`.classe { ... }`

SELETORES CSS



PRINCIPAIS SELETORES CSS

Propriedade	O que faz	Exemplo
<code>background-color</code>	Define a cor de fundo de um elemento	<code>background-color: lightblue;</code>
<code>color</code>	Define a cor do texto	<code>color: darkblue;</code>
<code>font-size</code>	Define o tamanho da fonte	<code>font-size: 20px;</code>
<code>text-align</code>	Alinha o texto (esquerda, centro, direita...)	<code>text-align: center;</code>
<code>margin</code>	Espaço fora do elemento	<code>margin: 10px;</code>
<code>padding</code>	Espaço dentro do elemento	<code>padding: 15px;</code>
<code>border</code>	Adiciona uma borda ao redor do elemento	<code>border: 2px solid black;</code>
<code>border-radius</code>	Deixa a borda arredondada	<code>border-radius: 10px;</code>
<code>font-weight</code>	Define o peso (espessura) da fonte	<code>font-weight: bold;</code>
<code>width / height</code>	Define largura/altura do elemento	<code>width: 300px;</code>

PRINCIPAIS SELETORES CSS



Google Fonts

html

Copiar Editar

```
<link href="https://fonts.googleapis.com/css2?family=Roboto&display=swap" rel="stylesheet">
```

css

Copiar Editar

```
body {  
  font-family: 'Roboto', sans-serif;  
}
```



O QUE É O JAVA SCRIPT?

- JAVASCRIPT É UMA LINGUAGEM USADA PARA DAR VIDA À PÁGINA: AÇÕES, REAÇÕES, ANIMAÇÕES, VALIDAÇÕES.

```
<!DOCTYPE html>
<html>
<head>
<title>Intro To Javascript</title>
</head>
<body>
<script>
    // My First Javascript
</script>
</body>
</html>
```



O QUE É O JAVA SCRIPT?

Conceito	Em C	Em JavaScript
variável	<code>int x = 5;</code>	<code>let x = 5;</code>
função	<code>void ola()</code>	<code>function ola() {}</code>
imprimir	<code>printf("oi");</code>	<code>console.log("oi");</code>
condição	<code>if (x > 0)</code>	<code>if (x > 0)</code>
comentário	<code>// isso é um comentário</code>	<code>// também aqui</code>



TESTANDO O JAVA SCRIPT

The screenshot shows a window titled "Untitled - TextEdit" with a menu bar containing "File", "Edit", "View", and "Help". The main text area contains the following JavaScript code:

```
<SCRIPT>

    ALERT("OLÁ, MUNDO!");

    CONSOLE.LOG("MENSAGEM NO CONSOLE");

</SCRIPT>
```



TESTANDO O JAVA SCRIPT

The screenshot shows a window titled "Untitled - TextEdit" with a menu bar containing "File", "Edit", "View", and "Help". The main text area contains the following HTML code:

```
<BUTTON ONCLICK="ALERT( 'VOCÊ CLICOU!' )">  
    CLIQUE AQUI  
</BUTTON>
```

FRONT-END BACK-END FULL STACK



Fab
Lab

CONTEÚDOS LEGAIS

M mdn web docs

w3
schools



Fab
Lab

OBRIGADA!

Alguma dúvida?

Vamos trocar ideias e testar
algumas coisas!



JORNADA TECH



Fab
Lab

JORNADA TECH

OFICINA DE AUTOMAÇÃO COM
ARDUINO



JORNADA
TECH

INTRODUÇÃO

O POR QUE APRENDER ELETRONICA

O O QUE VAI APRENDER ?

- ENTENDA CORRENTE, TENSÃO E RESISTÊNCIA COM A LEI DE OHM.
- CONHEÇA RESISTORES, LEDS, CAPACITORES E MUITO MAIS.
- DOMINE A PROTOBOARD PARA CIRCUITOS SEM SOLDA.
- UTILIZE O MULTÍMETRO PARA TESTAR COMPONENTES.

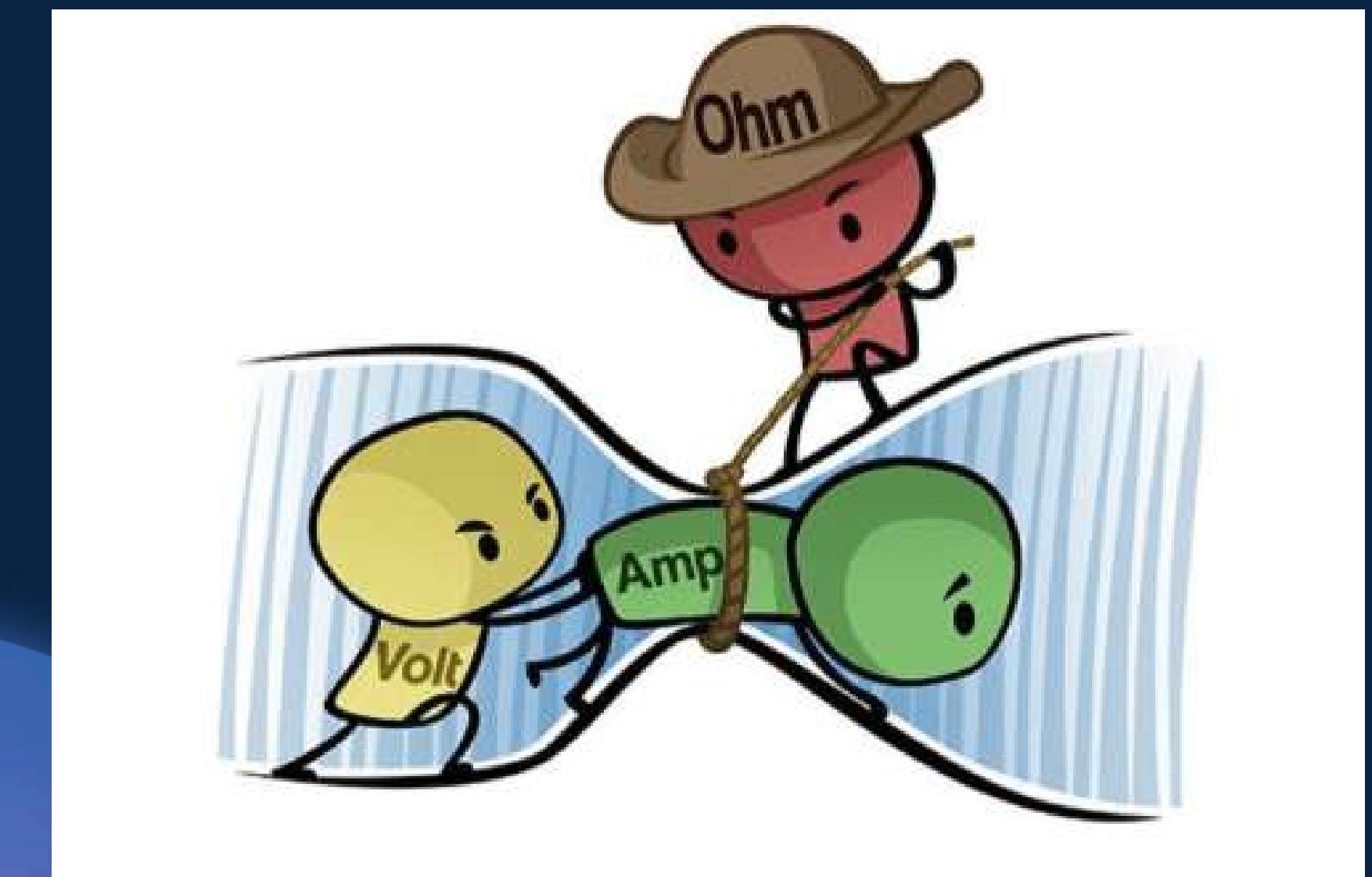
CONCEITOS BASICOS

CORRENTE ELETRICA

TENSÃO

RESISTENCIA ELETRICA

POTENCIA ELETRICA



EQUIPE



ANA BEATRIZ SCHUINDT



NAMEM RACHID JAUDY



CARLOS MIGUEL
OLIVEIRA



LEONARDO ANTUNES

CONCEITOS FUNDAMENTAIS

CORRENTE ELETRICA

LEI DE OHM

TRIANGULO DE POTENCIA

TESTE E MEDIÇÃO



SEGURANÇA

BOAS PRATICAS NA MONTAGEM DE
CIRCUITOS ELETRICOS

SEGURANÇA AO TRABALHAR COM
ELETRICIDADE



ELETRÔNICA BÁSICA PARA ARDUINO

FUNDAMENTOS ESSENCIAIS
COMPONENTES CHAVE
MONTAGEM PRÁTICA
TESTE E MEDIÇÃO

- ENTENDA CORRENTE, TENSÃO E RESISTÊNCIA COM A LEI DE OHM.
- CONHEÇA RESISTORES, LEDS, CAPACITORES E MUITO MAIS.
- DOMINE A PROTOBOARD PARA CIRCUITOS SEM SOLDA.
- UTILIZE O MULTÍMETRO PARA TESTAR COMPONENTES.

INTRODUÇÃO AO ARDUINO

- O QUE É ARDUINO?
- PLATAFORMA DE PROTOTIPAGEM ELETRÔNICA DE CÓDIGO ABERTO. IDEAL PARA PROJETOS INTERATIVOS.
- MODELOS POPULARES CONHEÇA O UNO, NANO E MEGA. CADA UM COM SUAS CARACTERÍSTICAS.
- PROGRAMANDO O ARDUINO APRENDA A IDE DO ARDUINO E A LINGUAGEM C++. SEU PRIMEIRO PASSO: PISCAR UM LED.

PLATAFORMAS ONLINE: TINKERCAD E FRITZING

- TINKERCAD
- SIMULE CIRCUITOS ELETRÔNICOS ONLINE. EXPERIMENTE SEM CUSTOS. RECURSO INTERATIVO PARA INICIANTES.
- FRITZING
- CRIE DIAGRAMAS ESQUEMÁTICOS E LAYOUTS DE PCB. FACILITE O DESIGN DO SEU PROJETO. COMPARTILHE SEUS PROJETOS COM FACILIDADE.
- VANTAGENS
- EXPERIMENTE SEM HARDWARE. COMPARTILHE PROJETOS FACILMENTE. IDEAL PARA APRENDIZADO E COLABORAÇÃO.

PROJETO 1: SEMÁFORO

MONTAGEM DA PARTE FÍSICA

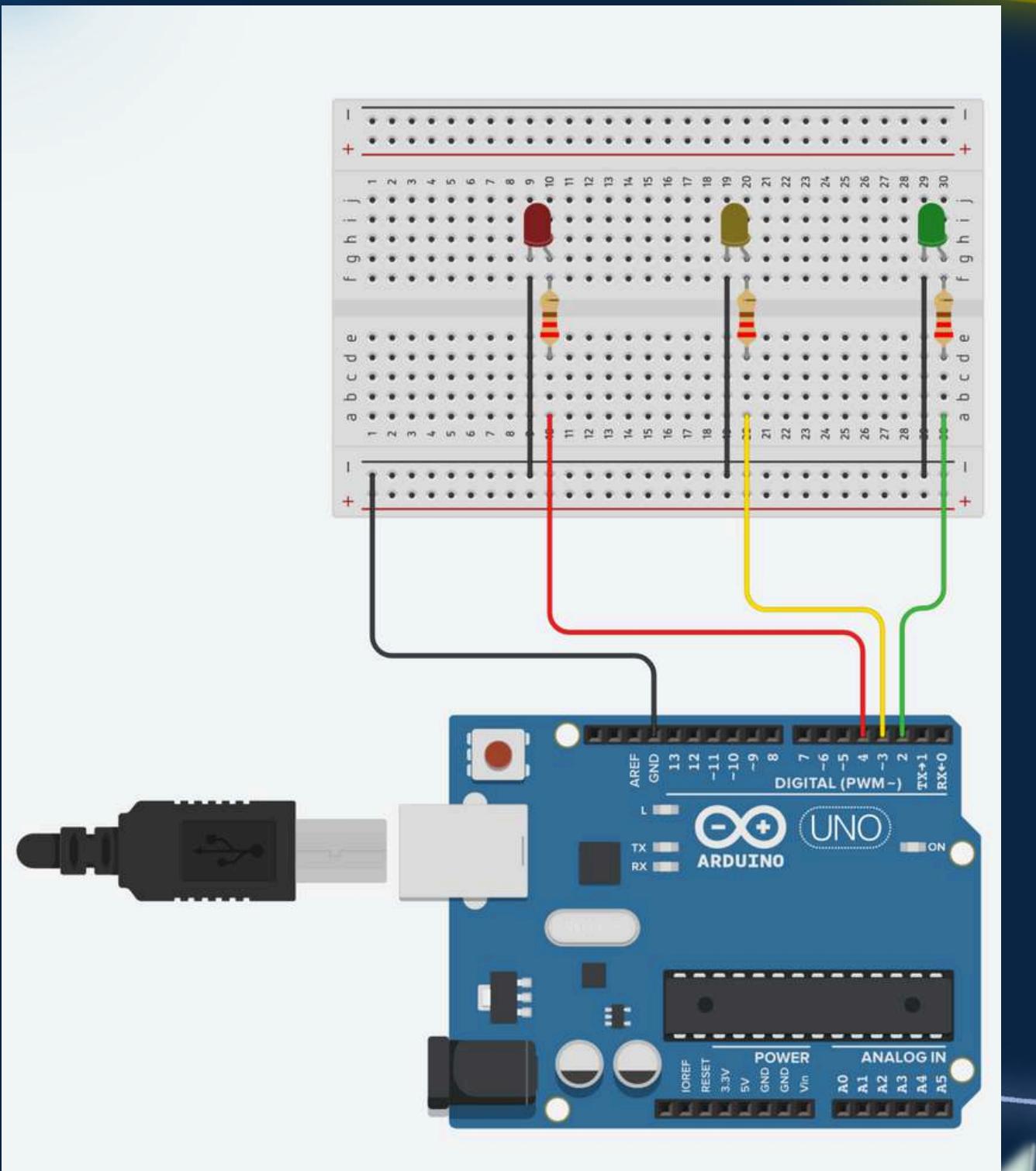
- CONECTAR OS LED E RESISTORES

PROGRAMAÇÃO

- CONTROLE O SEMÁFORO COM ARDUINO. DEFINA A LÓGICA DE CORES.

SINCRONIZAÇÃO

- IMPLEMENTE TEMPOEIZADORES. AJUSTE OS DELAYS



PROJETO 2: SENSOR LDR

MONTAGEM DA PARTE FISICA

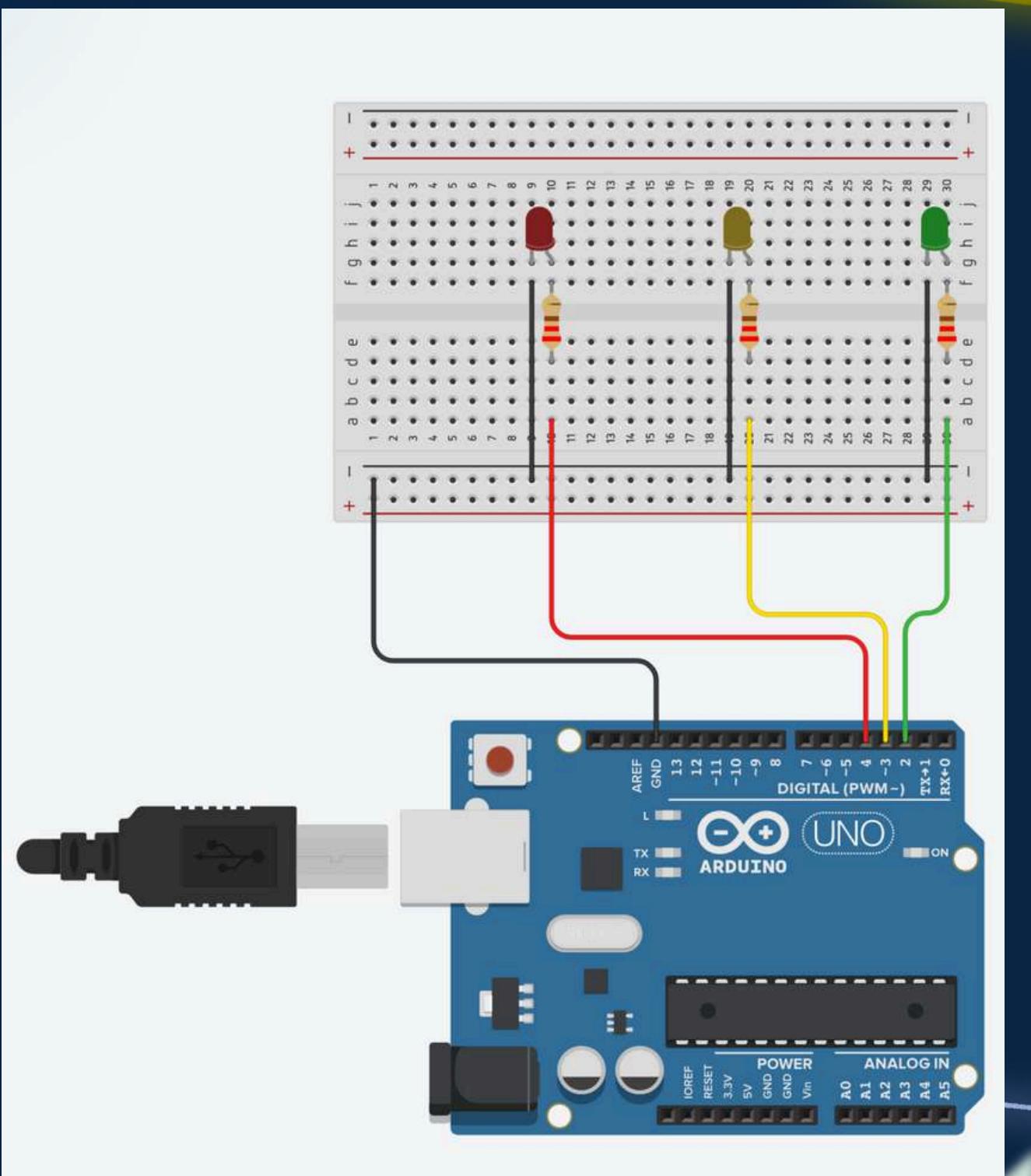
- SENSORES DE LUZ E COMO ATUAM

PROGRAMAÇÃO

- MAPEIE VALORES PARA LIGAR UM LED AUTOMATICAMENTE

SINCRONIZAÇÃO

- CONTROLE DE NO ARDUINO

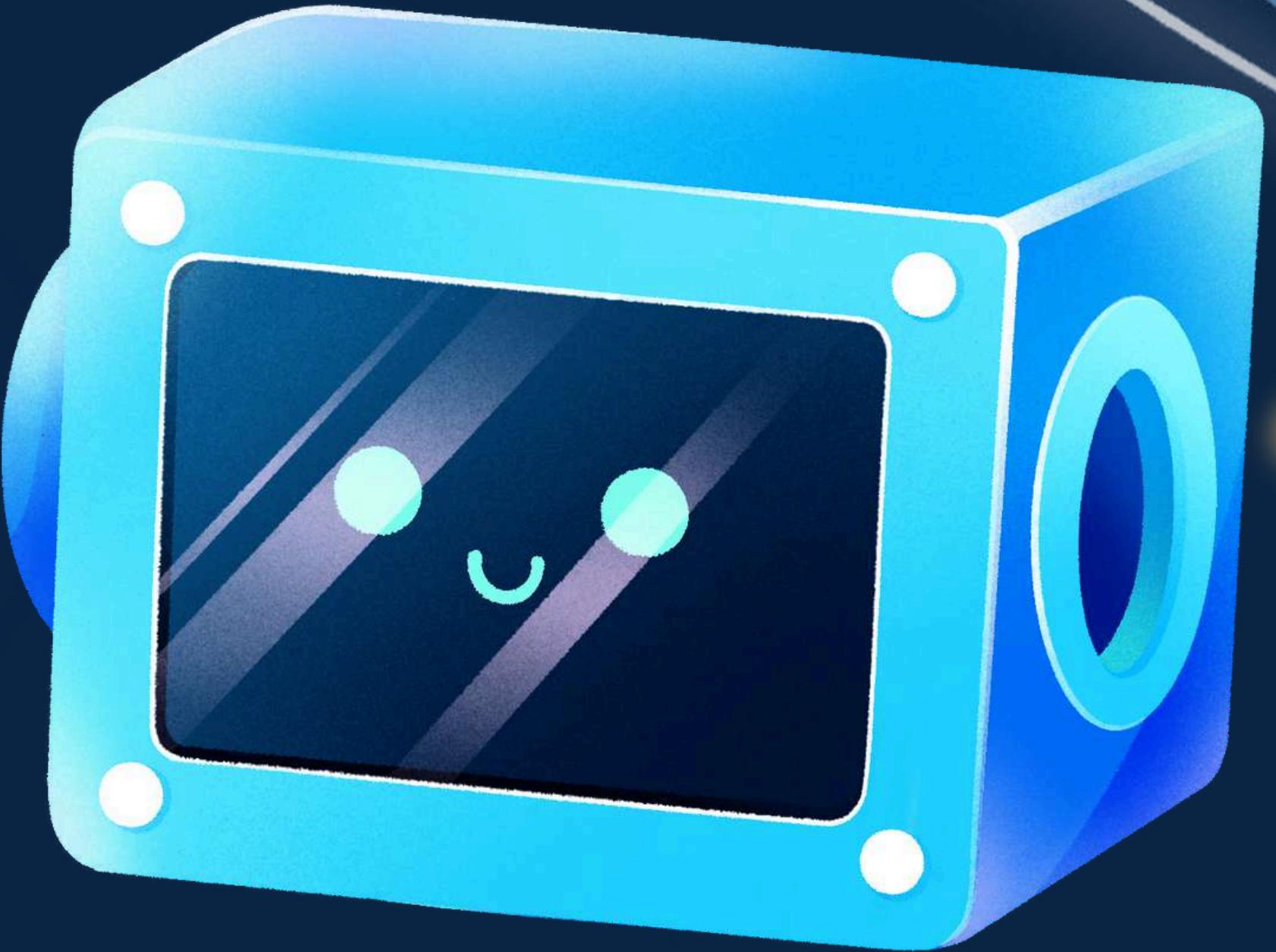


Fab
Lab

OBRIGADA!

Alguma dúvida?

Vamos trocar ideias e testar
algumas coisas!



JORNADA TECH



Fab
Lab

JORNADA TECH

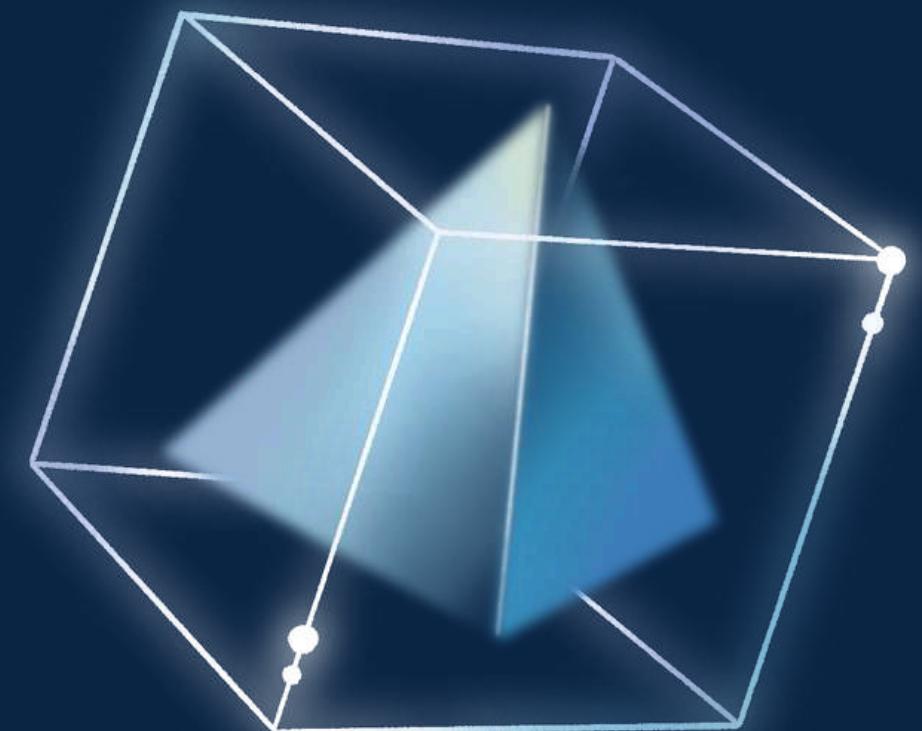
UMA PRÉVIA DO CURSO DE
ENGENHARIA DA COMPUTAÇÃO



JORNADA
TECH

RETROSPECTIVA

- LOGICA DE PROGRAMAÇÃO.
- DESENVOLVIMENTO WEB.
- ELETRONICA BASICA E PROJETOS DE AUTOMAÇÃO.
- CARREIRAS E UM PANORAMA DA CARREIRA.



CONVIDADOS



ICARO ALENCAR - ENG. DA COMPUTAÇÃO
PROGRAMADOR SENIOR GENEXUS



RUBIA MARA - ENG. DA COMPUTAÇÃO
PROGRAMADORA JR. GENEXUS
PROGRAMADORA EM BLOCOS SENIOR

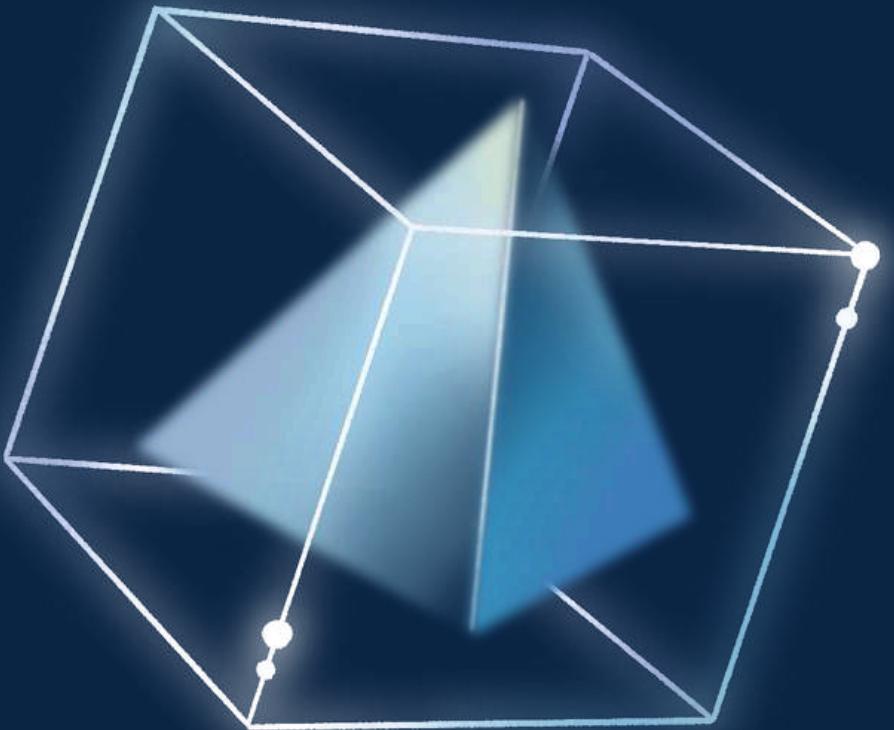


REGRAS BASICAS

- "AQUI NÃO TEM PERGUNTA BOBA! ESSE É UM ESPAÇO SEGURO PRA TROCAR IDEIA, CONTAR HISTÓRIAS, TIRAR DÚVIDAS E SE INSPIRAR."
- "A IDEIA É SER UM BATE-PAPO. PODE LEVANTAR A MÃO, PERGUNTAR, CONTAR UMA HISTÓRIA. QUEREMOS TE OUVIR TAMBÉM!"
- "TODO MUNDO AQUI TEM UMA TRAJETÓRIA DIFERENTE - ENTÃO VAMOS OUVIR COM ATENÇÃO E FALAR COM EMPATIA."

RODA DE CONVERSA

- O QUE É ENGENHARIA DA COMPUTAÇÃO, AFINAL?
- COMO FOI COMEÇAR NA ÁREA ?
- QUAIS FORAM SUAS MAIORES DIFICULDADES ?
- O QUE MAIS TE EMPOLGAVA?



RODA DE CONVERSA

- QUAIS MATERIAS VOCÊS TIVERAM QUE MAIS CURTIRAM OU MAIS SOFRERAM?
- COMO É APRENDER A PROGRAMAR? PRECISA SABER MUITO DE MATEMÁTICA?
- JÁ CRIARAM ALGUM APP, JOGO OU SISTEMA LEGAL DURANTE A FACULDADE?

MERCADO DE TRABALHO E FUTURO

- COMO ESTÁ O MERCADO DE TECNOLOGIA HOJE?
- QUAIS ÁREAS ESTÃO BOMBANDO? (IA, SEGURANÇA, EMBARCADOS, ETC.)
- É POSSÍVEL EMPREENDER OU TRABALHAR COM JOGOS, POR EXEMPLO?
- COMO VOCÊS LIDAM COM A COMPETITIVIDADE? TEM MAIS COOPERAÇÃO OU RIVALIDADE?
- O QUE DEFINE O CRESCIMENTO NA ÁREA? EXPERIÊNCIA, PROJETOS, CERTIFICAÇÕES?

DICAS PARA QUEM QUER COMEÇAR

- QUE CURSOS GRATUITOS OU PLATAFORMAS VOCÊS INDICAM?
- COMO APRENDER PROGRAMAÇÃO SOZINHO?
- O QUE VOCÊS DIRIAM PARA O "VOCÊS MESMOS" DE 5 ANOS ATRÁS?
- O QUE VOCÊS GOSTARIAM DE TER APRENDIDO ANTES DE ENTRAR NO CURSO?

EQUIPE



ANA BEATRIZ SCHUINDT



NAMEM RACHID JAUDY



CARLOS MIGUEL
OLIVEIRA



LEONARDO ANTUNES

Fab
Lab

OBRIGADA!

Alguma dúvida?

Vamos trocar ideias e testar
algumas coisas!

