Curso de Java Script – Roger Mello

# Aula 001

40:01

Aula inaugural

Visão do Roger sobre como realizar um curso, suas ideias e regras.

Comece e não pare, devagar ou rápido, não pare, não pule as aulas e não faça diferente do professor perdendo o foco

Entender é diferente de executar, pratique, mesmo que vc tenha entendido tudo

Nove marcos para chegar na fluência

Alicerce

Marco 1 – Assistir aula inaugural

Marco 2 – Realizar todas as atividades – Postar resumo/fazer perguntas das aulas no repositório github = roger-melo-treinamentos/curso-de-js-roger-melo

Marco 3 – Como realizar as atividades: 0 – Faça os exercícios – 2 – Faça os exercícios corrigidos junto com o professor até que vc entenda os procedimentos – Implemente novamente, sozinho, do jeito do professor – 3 – Se tiver dúvidas, procure tirá-las usando suas anotações, e tente fazer sozinho novamente do zero.

Fazer uma revisão raio-x antes de cada implementar cada aplicação – Tem link explicando

Validação

Marco 4 – Análise das aplicações pelo professor

Marco 5 – Montar um portifólio com as 3 primeiras aplicações, no github e website. Pelo jeito o Roger não sabe da existência do github pages por citou que as pessoas podem não saber mexer com o github e baixar os seus projetos

Marco 6 – Se candidatar as vagas – Quando o medo é maior que o desejo vc paralisa, senão vc age. A cura do medo é ação

Evolução

Marco 7 – Implementar os feedbacks dos recrutadores

Marco 8 – Ser contratado

Comemoração

Marco 9 – Postar sua comemoração no repositório do treinamento

Plantão da fluência. atendimento individualizado de questões não técnicas

# Aula 002

01:05

Objetivos do treinamento

# Aula 003

03:01

Definindo JS

# Aula 004

07:10

Configurando o ambiente – VSCode – A extensão Live Server

<html lang=”pt-br”>

Instalar o f.flux inibe a luz azul do Monitor para descansar a visão

VSCode ctrl + shift + p -> Atalho para todas as funções

VSCode ctrl + b -> Fecha/Abre a aba de arquivos

Emmet ! <tab> -> Estrutura html

# Aula 005

03:36

alert(‘Mensagem’)

Mensagem popup no browser com botão OK

<script src=”App.js”></script>

# Aula 006

02:28

Chrome f12 -> Abrir aba Devtools

Configurar o Chrome DevTools com temas diferentes

DevTools -> ‘3 pontinhos’ -> Setings -> Aparencia -> Theme

console.log(‘mensagem’) -> exibir mensagens em terminais ou no console do devtools no chrome

Javascript é executado de forma sequencial de cima para baixo

# Aula 007

09:11

Constante, variáveis e comentários

let nomeVar

const nomeVar2

var nomeVar3 -> var não é mais usado

console.log(nomeVar, ‘Texto’, nomeVar2, nomeVar3)

let e const vieram no JS5

regras de nomeação de variáveis - camelCase

// Comentário de 1 linha

/\*

Cometários de várias linhas

\*/

VSCode ctrl + ; Coloca/retira comentários de linha

# Aula 008

04:44

Primeiros exercícios

No github – Em um olhinho escrito watch vc pode clicar nele e selecionar uwatch para ser avisado de alteração naquele repositório

Instalação do git

Instalação do terminal Hyper feito em html css js

hyper.is

git –version

git version 1.9.1

VSCode shift + ctrl + e -> Só abre a aba arquivos, não fecha

# Aula 009

08:04

Correção dos exercícios da aula 001

# Aula 010

03:21

Overview sobre tipos de dados

Number, String, Boolean, Undefined, Null, Object, Symbol, BigInt

Null -> quando nós queremos deixar explicitamente uma variável sem valor

Undefined -> O JS declara quando uma variável não possui valor e nem foi declarada com null

# Aula 011

12:52

Strings

“um texto normal”

‘um texto normal’

`texto ${variável} texto texto ${variável}`

‘texto’ + “texto2” + ‘texto’

textotexto2texto

Concatenação de strings

“texto”.length -> exibe a quantidade de caracteres

let novo = ‘texto’

console.log(novo[1])

e

JS – Zero based

console.log(“texto”.toUpperCase())

TEXTO

console.log(“TEXTO”.toLowerCase())

texto

Strings possui métodos que agem em cima do objeto original e outros em cópias destas strings

O Professor declarou que as Strings são imutáveis porém depois citou que existe métodos que agem no original e outros que agem na cópia

console.log(“xxx@xxx.com”.indexOf(‘@’));

3

# Aula 012

08:06

Métodos comuns de strings

‘texto’.lastIndexOfA(‘t’)

3

Retorna o índice da última ocorrência do parâmetro na string

const email = 'laracroft@gmail.com'

console.log(email.slice(0, 9))

laracroft

Retorna a string entre o primeiro parâmetro até o segundo parâmetro

Pode ser feito desta forma tumém

const email = 'laracroft@gmail.com'

console.log(email.slice(0, email.indexOf('@')))

laracroft

Não usando o Segundo parâmetro é retornado a string entre o valor do parâmetro usado até o fim da string

const email = 'laracroft@gmail.com'

console.log(email.slice(9))

@gmail.com

string.replace(‘string’,’string’)

const email = 'laracroft@gmail.com'

console.log(email,'\n',email.replace('lara', 'sarada'))

laracroft@gmail.com

saradacroft@gmail.com

Troca a primeira ocorrência da string do primeiro parâmetro pela do segundo parâmetro

# Aula 013

11:34

Numbers

Operadores aritméticos

/ -> Divisão

% -> Modulo

Área de um círculo

const radius = 10

const pi = 3.14

const area= pi \* radius \*\* 2

‘pi vezes (\*) radius Elevado (\*\*) ao quadrado (2)’

Ordem de precendência

1 -> Parêntese

2 -> expoentes (exponenciação) ou raízes

3 -> Multiplicação e divisão

4 -> Adição e subtração

Ou seja

radius \*\* 2 -> Foi resolvido primeiro

pi \* ‘resultado da expressão acima’

Incremento e decremento

let temp = 10;

console.log(temp++);

console.log(temp)

console.log('--------------')

temp = 10;

console.log(++temp);

console.log(temp)

console.log('--------------')

Resultado:

10

11

--------------

11

11

--------------

-------------------------------------------

Addition/Subtraction/mult/div assignment

destination assignment

let temp = 10

temp += 10

cosole.log(temp)

20

= temp = temp + 10

------------------------------------

NaN

# Aula 014

09:12

Template strings ou templates literals

`${variavel} texto texto `${variavel} texto `${variavel}`

\’ -> Escapando aspa simples

|” -> Escapando aspando enters duplas

\n ->Escapando enter

\t -> Escapando tabulação

Crase ou Back strict?

Templates html

const html =

`

<h2>${posTitle}$</h2>

<p>Autor: ${posAutor}$</p>

<span>Este post contém: ${postComents} comentarios</span>

`

Ai é só usar no html com .textHtml = html

# Aula 015

1,29

Exercícios

# Aula 016

27:56

EMMET script[src=”./app.js”] <tab>

# Aula 017

10:10

# Aula 018

# Aula 019

# Aula 020

# Aula 021