***Selecionando tag html.***

Para podermos pegar um elemento por Id devemos colocar um ID na tag html e com a função  document.*getElementById(‘nome do id’).*

com isso podemos capturar todo o elemento do html por ID.

Outra forma que temos para elementos é pelo nome da classe, com o document.*getElementsByClassName(‘nome da classe’).*

assim podemos selecionar vários elementos de uma vez.

Podemos usar o document.*getElementsByTagName(‘nome da tag’)*

para selecionar as tags que desejamos selecionar;

o *querySelector(‘.one’);* faz uma busca pelo seletor podemos passar tanto o nome da class como tags e atributos esse método captura o primeiro elemento que ele encontra no documento já o *querySelectorAll([src])* ele traz uma coleção de nós ou seja ele traz todos os elementos que têm o seletor.

***Manipulando elemento e estilos***

Manipulação de conteúdo com a propriedade. **textContent** podemos adicionar ou modificar um elemento selecionado já o **innerText** troca o texto interno do seu html. para podermos inserir tags html usamos o **innerHtml.**  o **.value** podemos manipular o valor de inputs por exemplo.

podemos manipular atributos tato setar e remover  com o **.setAttribute** e **.removeAttribute**

-Mas como podemos alterar os estilos do documento com JS, podemos usar a função **.style** esse método irá inserir css inline para podermos mudar uma cor por exemplo nós não dividimos as propriedades com hífen(-) como fazemos no css nos utilizaremos o camelCase EX: backgroundColor: red,

Também para manipulação de estilos temos o **.classList** que é bem parecido com um array de estilos essa função podemos fazer muitas coisas com usar o método **.classList.add(‘classe css’)** para podermos adicionar uma classe temos o **.classList.remove(‘classe css’)** para remover a classe desejada o **.classList.toggle(‘classe css’)** serve com um interruptor se a tag faz uso de determinada classe css ele tira se a tag não faz uso ele adiciona a classe a tag sem contar que essa função é composta por vários outros métodos.

***Navegando pelos elementos***

Para selecionamos o elemento pai de determinada tag html usamos o método **.parentNode** ou podemos usar o **.parentElement** que ele irá retornar o pai do elemento selecionado

já para pegarmos os elementos filhos temos os **.children** traz um HTMLCollection com todos os filhos, temos o **.firstElementChild** vai pegar o primeiro filho da tag e o **.lastElementChild** vai pegar o último elemento filho.

para buscar um irmão mais próximo temos o método **.nextElementSibling**

***Criando e adicionando elementos no html***

Para criar elementos devemos criar uma constante e usarmos a função document.createElement('elemento') assim que esse elemento é criado vamos selecionar onde queremos colocar esse elemento criado ou seja o qual será o pai desse elemento apos selecionar com um querySelector por exemplo temos duas maneiras de inserir esse novo elemento no pai com a função

.append(elemento) // para posicionar o elemento no inicio

.prepend(elemento) // para posicionar o elemento no final

Já o .insertBefore é utilizado para colocar o elemento antes de uma determinada tag que você passa por parâmetro quando chama esse método **.insertBefore(elemento, elemento que vc quer que venha depois)**