Para pegar o valor de algum elementos, id ou classe ou os 3 juntos, o código abaixo também mostra como pegar a frase toda e separala por espaços vazios para sabermos quantas palavras temos

var frase = $(".frase").text();

var numPalavras = frase.split(" ").length;

Evento de click, a expressão reglar é para não bugar a contagem quando tiver espaços vazios na string

var campo = $('.campo-digitacao');

campo.on('click', function(){

var conteudo = campo.val();

var qtdPalavras = conteudo.split(/\S+/).length - 1;

$("#contador-palavras").text(qtdPalavras);

var qtdCaracteres = conteudo.length;

$("#contador-caracteres").text(qtdCaracteres);

});

Há tbm o evento input que executa a função quando o input recebe uma entrada

var campo = $('.campo-digitacao');

campo.on(input, function(){

var conteudo = campo.val();

var qtdPalavras = conteudo.split(" ").length;

$("#contador-palavras").text(qtdPalavras);

var qtdCaracteres = conteudo.length;

$("#contador-caracteres").text(qtdCaracteres);

});

O código abaixo serve para fazer um cronometro para digitar alguns textos

O evento one serve para que a função seja chamada somente uma vez ao ser dado o foco, não perdendo a lógica, a função setInterval retorna um id esse id é usado na função clearInterval, essa função serve para que a execução possa ser encerrada da forma correta, para colocar um atributo a um elemento é usado o attr, porem no caso do disabled não temos valor para ele então passamos true como segundo parametro

var tempoRestante = $('#tempo-digitacao').text();

campo.one('focus', function(){

var cronometroId = setInterval(function(){

tempoRestante--;

$('#tempo-digitacao').text(tempoRestante);

if(tempoRestante < 1){

campo.attr('disabled', true);

clearInterval(cronometroId);

}

}, 1000)

});

As funções principais, ja tem as funções proprias para serem chamadas

$('#reiniciar').click(function(){

console.log('Botão clicado');

})

Para executar algum código assim que a pagina terminar de ser carregada

$(document).ready(function(){ //codigo aqui}

Ha um atalho tbm

$(function(){ //codigo aqui}

Para adicionar um estilo inline no código, porem nao é recomendado. Com essa propriedade tambem da pra pegar o valor que esta setado atualmente não passando o segundo parametro

campo.css('background-color', 'lightgray');

É possivel pegar mais de um valor de uma vez

var valores = $("div").css(["background-color","width"]);

O mias correto seria inserir uma classe

campo.addClass('campo-desativado');

Remover classe

campo.removeClass('campo-desativado');

Essa função serve para ativar ou desativar a classe automaticamente setando o valor inverso que esta atualmente

campo.toggleClass('campo-desativado');

Ja existe uma função mais nova que verifica se uma string é igual a um pedaço de outro text, segue exemplo abaixo

var digitouCorreto = frase.startsWith(digitado);

Para verficar se o navegador tem suporte a essa tecnologia

"ECMA Script 6".startsWith("ECMA")

A função append coloca o conteudo sempre no final do elemento, a prepeend insere no inicio

corpoTabela.append(linha);

corpoTabela.prepend(linha);

Para buscar algum elemento dentro de outro usa-se o find

var corpoTabela = $('.placar').find('tbody');

Função para remover alguem da lista, o parent esta duplicado para poder subir dois niveis

$('.botao-remover').click(function(event){

event.preventDefault();

$(this).parent().parent().remove();

});

Efeitos não são adicionados a elementos criados como string

Para criar um elemento com jquery

var linha = $('<tr>');

Criar e atribuir valores

var colunaUsuario = $("<td>").text(usuario);

Dado o seguinte HTML:

<p id="descricao">jQuery - write less, do more</p>

e este código Javascript(com jQuery importado previamente):

$('#descricao').click(function() {

//AQUI});

Como podemos selecionar o texto do parágrafo no lugar de //AQUI com jQuery?

Resposta = var texto = $(this).text();

Temos que lembrar que dentro de um evento do Javascript e do jQuery, a palavra reservada this sempre se refere ao elemento que sofreu o evento, neste caso ao elemento clicado. Como o elemento clicado é um elemento do HTML, ele se torna um objeto tradicional do Javascript dentro de nosso código. E como já sabemos, a função **.text()** é exclusiva de objetos jQuery, logo temos de convertê-lo antes.

Para converter um objeto tradicional do Javascript em um objeto jQuery, devemos fazer uso da função jQuery, passando o this para ela, deste modo:

var texto = $(this).text();