**Sisfrete - Desafios para o programador**

**Atenção!! Leia aqui antes de realizar os desafios:**

- **PHP** deve ser a linguagem utilizada para resolver os desafios.

- Você deve escolher pelo menos **3** dos 4 desafios abaixo para resolver.

- O código deve ser enviado para um **repositório público no Github** na sua conta

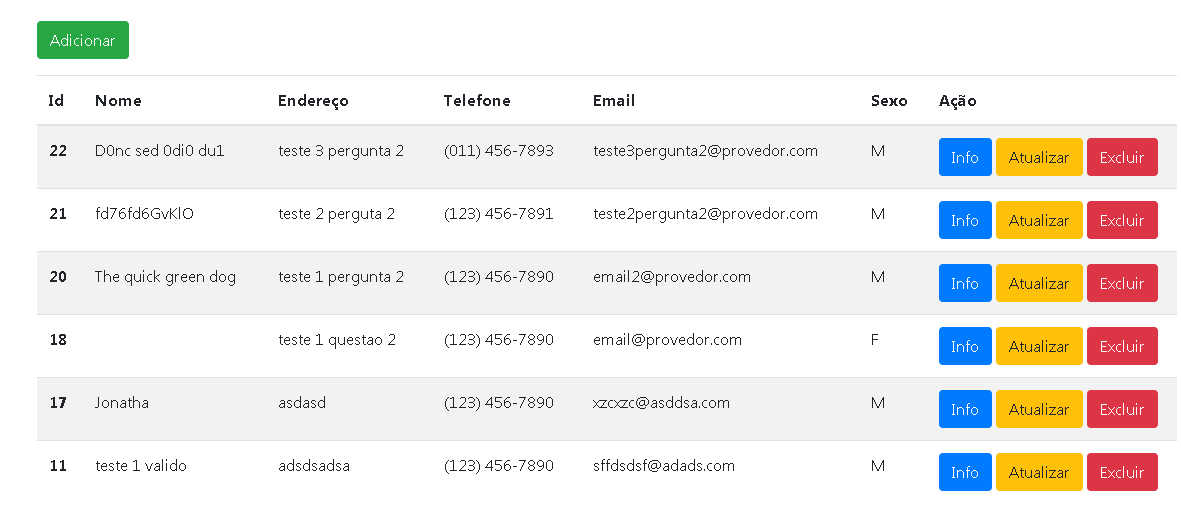
pessoal e o link do repositório(somente o link) deve ser enviado ao email de contato

informado. Anexos enviados por email **serão desconsiderados** .

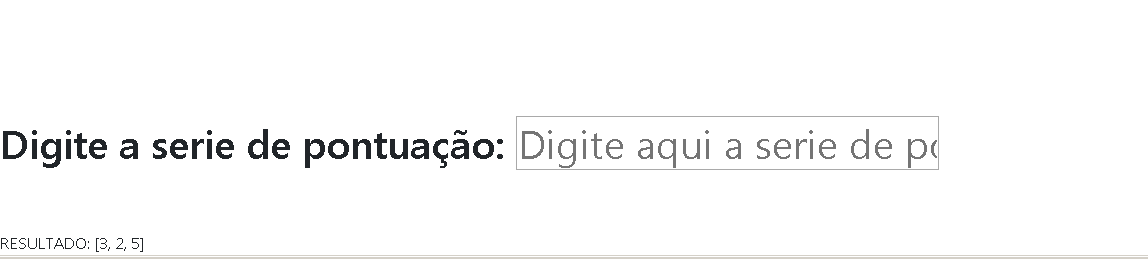
- Que a Força esteja com você ;)

**Nome: Jônatha Mendonça dos Santos**

**Observaçãp: hospedado e banco crud**



Caminho local host <http://localhost/projeto.crud.php-master/index.php>



Caminho local host <http://localhost/projeto.crud.php-master/sistemap/index.php>

**1 - O número de telefone está formatado corretamente?**

**Crie uma função que aceite uma string e retorne true se estiver no formato de um número**

**de telefone adequado e false se não estiver. Assuma que qualquer número entre 0 e 9**

**(nos locais apropriados) produzem um número de telefone válido.**

**É assim que um número de telefone válido se parece: (123) 456-7890**

**Exemplos:**

**isValid("(123) 456-7890") ➞ true**

**isValid("1111)555 2345") ➞ false**

**isValid("098) 123 4567") ➞ false**

**Nota:**

**Não esqueça o espaço após os parênteses.**

Rascunho

function celular($telefone){

$telefone= trim(str\_replace('/', '', str\_replace(' ', '', str\_replace('-', '', str\_replace(')', '', str\_replace('(', '', $telefone))))));

$regexTelefone = "^[0-9]{11}$";

//$regexCel = '/[0-9]{2}[6789][0-9]{3,4}[0-9]{4}/'; // Regex para validar somente celular

if (preg\_match($regexCel, $telefone)) {

return true;

}else{

return false;

}

}

<https://pt.stackoverflow.com/questions/144362/como-validar-telefone-em-php>

**Resposta**

if(empty($telefone))

{

$telefoneErro = 'Por favor digite o número do telefone!';

$validacao = false;

}

else

{

if(!preg\_match('^\(+[0-9]{3}\) [0-9]{3}-[0-9]{4}$^', $telefone)){

echo "Telefone inváildo.";

$telefoneErro = 'formato errado';

$validacao = false;

}

}

// isValid("(123) 456-7890") ➞ true $telefone = "(82) 5555-5555";

<https://www.diegomacedo.com.br/validacao-de-dados-no-php/>

<https://regex101.com/r/hB8dH7/1>

<https://github.com/jonathadev/projeto.crud.php>

**2 - Remover os caracteres especiais de uma string**

**Crie uma função que recebe uma string e remove todos os caracteres "especiais" (por**

**exemplo,!@#$%^&\\*) E retorne a nova string. Os únicos caracteres não alfanuméricos**

**permitidos são traços -, underscore \_ e espaços.**

**Exemplos:**

**removeCharacters("The quick green dog!") ➞ "The quick green dog"**

**removeCharacters("%fd76$fd(-)6GvKlO.") ➞ "fd76fd-6GvKlO"**

**removeCharacters("D0n$c sed 0di0 du1") ➞ "D0nc sed 0di0 du1"**

**Rascunho:**

<?php

function sanitizeString($str) {

$str = preg\_replace('/[áàãâä]/ui', 'a', $str);

$str = preg\_replace('/[éèêë]/ui', 'e', $str);

$str = preg\_replace('/[íìîï]/ui', 'i', $str);

$str = preg\_replace('/[óòõôö]/ui', 'o', $str);

$str = preg\_replace('/[úùûü]/ui', 'u', $str);

$str = preg\_replace('/[ç]/ui', 'c', $str);

// $str = preg\_replace('/[,(),;:|!"#$%&/=?~^><ªº-]/', '\_', $str);

$str = preg\_replace('/[^a-z0-9]/i', '\_', $str);

$str = preg\_replace('/\_+/', '\_', $str); // ideia do Bacco :)

return $str;

}

?>

<https://pt.stackoverflow.com/questions/858/refatora%C3%A7%C3%A3o-de-fun%C3%A7%C3%A3o-para-remover-pontua%C3%A7%C3%A3o-espa%C3%A7os-e-caracteres-especiais>

**Resposta:**

function sanitizeString($str) {

$str = preg\_replace('/[áàãâä]/ui', 'a', $str);

$str = preg\_replace('/[éèêë]/ui', 'e', $str);

$str = preg\_replace('/[íìîï]/ui', 'i', $str);

$str = preg\_replace('/[óòõôö]/ui', 'o', $str);

$str = preg\_replace('/[úùûü]/ui', 'u', $str);

$str = preg\_replace('/[ç]/ui', 'c', $str);

//$str = preg\_replace('/[,(),;:|!"#$%&/=?~^><ªº-]/', '\_', $str);

$str = preg\_replace('/[^a-z0-9 ]/i', '', $str);

$str = preg\_replace('/\_+/', '\_', $str); // ideia do Bacco :)

return $str;

}

$nome = sanitizeString($nome);

**3 - Sistema de pontuação**

**Ana, Bruna e Caio estão jogando um jogo de tabuleiro. Os três decidiram criar um novo**

**sistema de pontuação. A primeira inicial de um jogador ("A", "B" ou "C") indica que o jogador**

**marcou um único ponto. Dada uma série de letras maiúsculas, retorne uma série de**

**pontuações dos jogadores.**

**Por exemplo, se ABBACCCCAC for escrito quando o jogo terminar, Ana marcou 3 pontos,**

**Bruna marcou 2 pontos e Caio marcou 5 pontos, pois existem 3 ocorrências da letra A, 2**

**ocorrências da letra B e 5 ocorrências da letra C. Portanto, a matriz [3, 2, 5] deve ser**

**retornada.**

**Exemplos:**

**calculaPontuacao("A") ➞ [1, 0, 0]**

**calculaPontuacao("ABC") ➞ [1, 1, 1]**

**calculaPontuacao("ABCBACC") ➞ [2, 2, 3]**

**Rascunho:**

$texto = "00513|01\_BRANCO|3\_4|8";

$QtdTxt = substr\_count($texto,'|');

echo $QtdTxt;

calculaPontuacao("A") ➞ [1, 0, 0]

$texto = "A";

$QtdTxt = substr\_count($texto,'A''B''C');

echo $QtdTxt;

$texto = "ABBACCCCAC";

$QtdTxt = substr\_count($texto,'A');

echo $QtdTxt;

<https://forum.imasters.com.br/topic/469145-contar-ocorrencia-do-caracter-na-string/>

**Resposta:**

function calculaPontuacao($str) {

$QtdTxtA = substr\_count($str,'A');

$QtdTxtB = substr\_count($str,'B');

$QtdTxtC = substr\_count($str,'C');

$valores[0] = $QtdTxtA;

$valores[1] = $QtdTxtB;

$valores[2] = $QtdTxtC;

echo $resultado = 'RESULTADO: ';

echo '[', $valores[0],', ';

echo $valores[1],', ';

echo $valores[2], ']';

return $str;

}

?>

<form method="POST" action="">

<label>Digite a serie de pontuação: </label>

<input type="text" name="nome" placeholder="Digite aqui a serie de pontuacao"><br><br>

</form>

</h1>

<?php

if(!empty($\_POST))

{

$texto = $\_POST['nome'];

//Validaçao dos campos:

$validacao = true;

if(empty($nome))

{

$nomeErro = 'Por favor digite o seu a serie de puntuacao!';

$validacao = false;

}

calculaPontuacao($texto);

}

?>

**4 - Dividir string por caracteres idênticos**

**Crie uma função que divida uma string em um array de grupos idênticos.**

**Exemplo:**

**dividirGrupos("555") ➞ ["555"]**

**dividirGrupos("5556667788") ➞ ["555", "666", "77", "88"]**

**dividirGrupos("aaabbbaabbab") ➞ ["aaa", "bbb", "aa", "bb", "a", "b"]**

**dividirGrupos("abbbcc88999&&!!!\_") ➞ ["a", "bbb", "cc", "88", "999",**

**"&&", "!!!", "\_"]**

**Notas:**

**- Cada grupo deve ter o mesmo caractere.**

**- O array retornado deve estar na mesma ordem que a sequência da string passada a**

**função.**

**- Deve funcionar com letras, números e outros caracteres.**

**Rascunho:**

for( $i=0; $i < $n\_caracteres ; $i++ )

for( $j=0; $j < $i ; $j++ )

{  
   print " $nome[$i] \n ";

i==[$i-1]

nome2[$j]= $nome[$i]  
}

for ou Split?