



solados coloniais franceses de origem muçulmana?  
\* UM TENENTE BRASILEIRO LIDEROU uma revolta de  
organizou o Batalhão Feminino da morte?  
\* A RUSSA MARIA BOCHKAREVA  
Você sabia que...



- 1) [2,0] Considere o exemplo de arquivo texto "pereciveis.csv" abaixo, que poderia guardar dados de diversos produtos perecíveis, no formato CSV (formato de valores separados por vírgula ou ponto-e-vírgula), cuja data de validade pode estar ultrapassada. Considere que o dia atual é "05/05/2023".

Descrição;Validade

Iogurte desnatado 1L;06/05/2023

Cerveja Acme 600ml;04/05/2023

Leite Integral Mimosa 1L;05/05/2023

Queijo Minerim 250g;03/05/2023

Faça um script que leia o arquivo CSV, que pode ter produtos diversos do que está no exemplo acima, e então imprima quais produtos estão fora do prazo de validade (vencidos) e há quantos dias.

- 2) [2,5] Considere o banco de dados "mma", localizado em "localhost", com *charset* "utf8", usuário "dev", senha "123456" e que tenha a tabela "lutador", cuja estrutura é indicada abaixo:

```
CREATE TABLE lutador (  
  id INT NOT NULL AUTO_INCREMENT PRIMARY KEY,  
  nome VARCHAR(100) NOT NULL,  
  peso_em_quilos DECIMAL(5,2) NOT NULL,  
  altura_em_metros DECIMAL(3,2) NOT NULL  
) ENGINE=INNODB;
```

- a) [2,0] Faça um script que consulte e liste cada lutador, com todos os seus dados.
- b) [0,5] Após a listagem realizada no script anterior, imprima os seguintes dados, consultados do banco de dados: número total de lutadores, média de alturas, a maior altura e o maior peso.

- 3) [4,5] Considere o arquivo "lutador.php" abaixo, que contém a classe Lutador:

```
class Lutador {  
  public $id = 0;  
  public $nome = '';  
  public $pesoEmQuilos = 0.00;  
  public $alturaEmMetros = 0.00;  
  public function __construct(  
    $id = 0, $nome = '', $pesoEmKilos = 0.00, $alturaEmMetros = 0.00 ) {  
    $this->id = $id;  
    $this->nome = $nome;  
    $this->pesoEmKilos = $pesoEmKilos;  
    $this->alturaEmMetros = $alturaEmMetros;  
  }  
}
```



- 0,95 a) [0,5] Crie um arquivo "repositorio-lutador.php" com um *namespace* "Mma" que contenha uma interface que represente um repositório de lutadores (abstração responsável por recuperar e manipular lutadores em algum meio de persistência – tal como um arquivo ou um banco de dados), que contenha métodos para (i) adicionar um lutador (objeto da classe Lutador) no meio de persistência; (ii) remover um lutador, a partir de um *id* fornecido, do meio de persistência.
- 2,05 b) [2,5] Crie um arquivo "repositorio-lutador-em-bdr.php", com um *namespace* "Mma", que implemente a interface anteriormente indicada, para o banco de dados relacional (BDR) indicado na questão anterior. Na implementação, não utilize quaisquer funções de impressão (como *echo* ou *die*).
- 1,2 c) [1,5] Crie um arquivo de script "remove.php" que NÃO esteja no *namespace* "Mma" e que: (i) solicite que o usuário informe o *id* de dois lutadores; (ii) Utilizando controle de transação e o repositório criado anteriormente, remova os lutadores informados.
- 1,0 4) [1,0] Faça uma função para formatar um número de telefone (*string*) recebido como argumento que, somente se o número contiver 10 ou 11 caracteres e esses forem numéricos, a função deve retornar o telefone formatado. Do contrário, deve retornar uma *string* vazia. O formato do telefone para quando esse tiver 10 caracteres (ex. "2225271727") é como neste exemplo: "(22) 2527-1727". O formato para quando tiver 11 caracteres (ex. "22988776655") é como neste exemplo: "(22) 98877-6655".

Boa prova!

#### Funções úteis (sintaxe simplificada; explicação não será fornecida):

```
checkdate(int $mes, int $dia, int $ano): bool;
isset(mixed $var [, mixed $...]): bool;
unset(mixed $var): void;
count(array $array): int;
in_array(mixed $valor, array $array [, bool $checarTipos]): bool;
array_search(mixed $valor, array $array, bool $strict = false): int|bool;
array_push(array &$array, mixed $value1 [, mixed $value2, ...]): int;
empty(string|array $valor): bool;
mb_strlen(string $texto): int;
mb_strpos(string $texto, string $busca [, int $inicio]): int;
mb_substr(string $texto, int $inicio [, int $tamanho]): string;
str_replace(mixed $busca, mixed $substituto, mixed $texto [, int &$modificacoesRealizadas]): mixed;
trim(string $texto [, string $caracteresARetirar]): string;
explode(string $delimitador, string $texto [, int $limite]): array;
implode(string $delimitador, array $a): string;
is_numeric(mixed $var): bool;
readline(string|null $prompt = null): string|false;
file_get_contents( string $arquivo ): string|false;
preg_match(string $expressao, string $valor [, array &$casamentos [, int $flags=0 [, int $offset= 0]]]): int;
```

#### Contantes úteis (explicação não será fornecida):

```
PHP_EOL
PDO::ATTR_ERRMODE
PDO::ERRMODE_EXCEPTION
PDO::FETCH_ASSOC
```