

Programação 2

Gabarito AP2

1. (2 pontos) Responda se é verdadeiro ou falso
 - a. (V) A Linguagem de Definição de Dados, componente do padrão SQL-92, é usada para definir bancos de dados.
 - b. (F) A expressão `ereg("[A-Za-z0-9_]*", "Este_teste")` retorna `false`.
 - c. (F) Em PHP, para eliminar o elemento de índice `pos` do array `arr` pode-se usar a construção `remove(arr,pos)`.
 - d. (F) A consulta SQL `"SELECT id, nome FROM usuarios ORDER BY nome WHERE id < 100"` está sintaticamente correta.
2. (2 pontos) Aponte os erros e corrija o seguinte script PHP de forma que o resultado seja
O Morango é vermelho.
A banana é amarelo.

```
<?php
$str = array("morango" => "vermelho", "banana" => "amarelo");

echo "O Morango é $str["morango"].<br>";

echo 'A banana é $str["banana"].<br>';

?>
```

Resp.

```
<?php
$str = array("morango" => "vermelho", "banana" => "amarelo");

echo "O Morango é ".$str["morango"]."<br>";

echo 'A banana é '.$str["banana"]."<br>';

?>
```

3. (2 pontos) Em PHP implemente uma função chamada `losango($n)` que imprime um padrão de algarismos na forma de um losango cuja altura e largura depende de `$n` (um número inteiro entre 1 e 9). Em particular, a borda exterior deve ser desenhada usando o algarismo 1, o centro deve ser desenhado com o algarismo `$n` e as demais posições devem ser preenchidas com algarismos de valor crescente desde o centro até a borda. Por exemplo, `losango(5)` exibe a seguinte figura:

```

  1
 121
12321
1234321
123454321
1234321
12321
 121
  1
```

Resp.

```
<?php
```

```
function line($k){
    $res = "$k";
    for($i = $k-1; $i > 0; $i--){
        $res = $i.$res.$i;
    }
    return $res;
}
function losango($k){
    $res = line($k)."\n";
    for($i = $k-1; $i > 0; $i--){
        $line_i = line($i);
        $str = "";
        for ($j = 0; $j < $k - $i; $j++)
            $str .= " ";
        $res = $str.$line_i."\n".$res.$str.$line_i."\n";
    }
    return $res;
}

echo "<pre>";
echo losango(5);
echo "</pre>";

?>
```

4. (4 pontos) Considere o diagrama ER para uma aplicação para locadora de DVDs conforme mostrado na figura abaixo (exercício proposto na avaliação anterior). Uma possível modelagem física para este banco de dados encontra-se a seguir.

```
CREATE TABLE cliente (
    id_cliente int(11) NOT NULL auto_increment,
    nome varchar(80) NOT NULL,
    endereco varchar(200) default NULL,
    CPF varchar(11) default NULL,
    PRIMARY KEY (id_cliente)
);
```

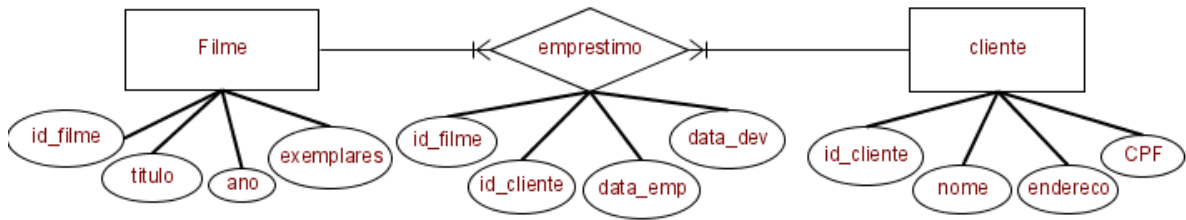
```
CREATE TABLE emprestimos (
    id_filme int(11) NOT NULL,
    id_cliente int(11) NOT NULL,
    data_emp datetime default NULL,
    data_dev datetime default NULL,
    PRIMARY KEY (id_filme,id_cliente)
);
```

```
CREATE TABLE filme (
    id_filme int(11) NOT NULL auto_increment,
    titulo varchar(80) NOT NULL,
    ano int(4) default NULL,
    exemplares int(3) default NULL,
```

```

PRIMARY KEY (id_filme)
) ;

```



Usando esta modelagem, pede-se:

- Escreva uma consulta SQL para mostrar os filmes mais emprestados no último mês.
- Escreva uma consulta SQL que permita obter os filmes que estão emprestados atualmente e os nomes dos clientes que os tomaram emprestados, ordenados a partir da data atual. Assuma que, quando um DVD emprestado é devolvido, o registro correspondente é removido da tabela emprestimo.

Resp.

a.

```

SELECT titulo, COUNT(*) AS nro_emprestimos FROM filme AS f,
emprestimos AS e
WHERE f.id_filme = e.id_filme AND (e.data_emp >= "2008-11-01"
AND e.data_emp <= "2008-11-30")
GROUP BY (titulo)

```

b.

```

SELECT titulo, c.nome, e.data_emp FROM filme AS f, emprestimos
AS e, cliente AS c
WHERE f.id_filme = e.id_filme AND
c.id_cliente = e.id_cliente
ORDER BY e.data_emp desc

```