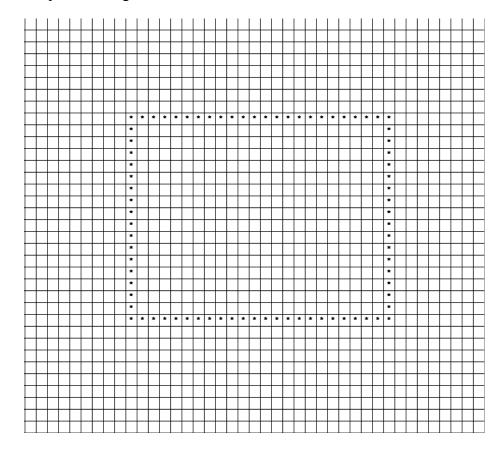
Gabarito da AP1 de Programação II

- 1. (2 Pontos) Diga se são verdadeiras ou falsas as seguintes afirmações:
 - i. (V) Em PHP, uma variável é criada quando a ela é atribuído um valor qualquer.
 - ii. **(V)** Em PHP, para concatenar strings é usado o operador ".", por exemplo,\$string1.\$string2.
 - iii. (F) Para eliminar um banco de dados em MySQL é usado o comando DELETE DATABASE.
 - iv. (F) Tabelas de dados Em MySQL devem ter uma chave primária PRIMARY KEY.
- 2. (4 Pontos) Escreva em PHP uma classe chamada desenho que define uma matriz de caracteres na qual se deseja fazer desenhos usando caracteres ASCII. O construtor dessa classe recebe dois números representando a largura e a altura da matriz. A classe deve suportar o método (função) retangulo (\$x1, \$y1, \$x2, \$y2) que desenha com asteriscos um retângulo com coordenadas mínimas (\$x1, \$y1) e máximas (\$x2, \$y2). A classe deve também suportar o método imprime() que escreve o resultado. Por exemplo, a seqüência de comandos

```
$d = new desenho(41,35);
$d->retangulo(10,10,27,33);
echo "\n"; $d->imprime(); echo "";
```

produz o seguinte resultado:



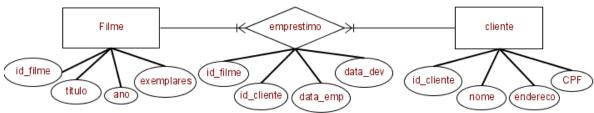
```
Resp.:
class desenho{
   var $width;
   var $height;
   var $screen = array();
   function desenho($w, $h){
      this->width = w+1;
      this->height = this
      for (\$i = 0; \$i < \$h; \$i + +) {
          $this->screen[] = array();
          for (\$j = 0; \$j < \$w; \$j + +) {
             $this->screen[$i][] = " ";
      }
   }
   function retangulo($x1, $y1, $x2, $y2){
      if ((\$x1 >= 0 \&\& \$x1 <= \$this->width) \&\&
           ($x2 >= 0 \&\& $x2 <= $this->width) \&\&
           (\$y1 >= 0 \&\& \$y1 <= \$this->height) \&\&
           (\$y2 >= 0 \&\& \$y2 <= \$this->height)){
          for (\$i = \$x1; \$i < = \$x2; \$i + +) {
             $this->screen[$i][$y1] = "*";
             $this->screen[$i][$y2] = "*";
          for (\$j = \$y1; \$j <= \$y2; \$j++) {
             $this->screen[$x1][$j] = "*";
             $this->screen[$x2][$j] = "*";
      }
   }
   function imprime(){
      for (\$i = 0; \$i<\$this->height; \$i++) {
          for (\$j = 0; \$j < \$this -> width; \$j ++) {
             echo $this->screen[$i][$j];
          echo "\n";
      }
   }
};
```

- 3. (4 Pontos) Suponha que lhe foi pedido a modelagem de um banco de dados para uma locadora de filmes. A modelagem deve suportar:
 - i. Cadastro de DVDs. Cada filme possui atributos como título e ano de produção e a locadora pode ter adquirido um certo número de exemplares de cada um.
 - ii. Cadastro de clientes. Cada cliente possui atributos como nome, endereço, telefone e CPF.
 - iii. Deve ser possível saber para cada filme, quantos DVDs estão na loja e quantos estão emprestados.
 - iv. Deve ser possível saber quais clientes tomaram emprestado um determinado filme.
 - v. Deve ser possível saber quais filmes um determinado cliente tomou emprestado.

Pede-se

- a. Crie uma modelagem entidade-relacionamento para o banco de dados.
- b. Crie um modelo físico para o banco de dados.
- c. Escreva um comando sql para inserir um novo filme com título "A Múmia", produzido em 1943 com 4 exemplares.
- d. Escreva um comando sql para registrar que um novo exemplar foi comprado para o filme com título "A Múmia".

Resp.:



```
CREATE TABLE filme (
id filme int(11) not null auto increment,
titulo varchar(10) not null,
ano int(4),
exemplares int(3),
PRIMARY KEY (id filme)
CREATE TABLE cliente (
id cliente int(11) not null auto increment,
nome varchar(80) not null,
endereco varchar(200),
CPF varchar(11),
PRIMARY KEY (id cliente)
);
CREATE TABLE emprestimos (
id filme int(11) not null,
id cliente int(11) not null,
data emp datetime,
data dev datetime,
PRIMARY KEY (id filme, id cliente)
);
```

```
INSERT INTO filme(titulo, ano, exemplares) VALUES ("A Múmia", 1943, 4);
UPDATE filme SET exemplares = exemplares + 1 WHERE titulo = "A Múmia";
```