# Curso de Tecnologia em Sistemas de Computação Disciplina: Programação de Aplicações Web - AP1X - 1° semestre de 2020.

# Glauber de Souza Faria

# 17213050160

# Questão 1

<?php

function y($x){

    #Expressao Y

    return $y = pow($x,2);

}

function check\_par($func\_name){

    #valor ficticio para X

    $x=2;

    $par = $func\_name($x) == $func\_name (-$x);

    $impar = $func\_name($x) == -$func\_name (-$x);

    if ($par == true){

        echo "Par";

    }

    elseif($impar == true){

        echo "Impar";

    }

    else{

        echo "ERRO";

    }

}

check\_par('y');

?>

# Questão 2

#PROGRAMA APRESENTADO

Function add\_extra($string) {

    $string .= "DEF";

   }

   $str = "ABC";

add\_extra($str);

echo $str;

Nesse contexto, teremos o programa principal e uma função, note que em momento algum o algoritmo retorna o novo valor da variável executada na função para o programa principal.

Exemplo:

**#criação da funçao add\_extra que recebe como parametro uma variavel chamada string.**

Function add\_extra($string) {

**#Varivavel string recebe seu valor passa por parâmetro + “DEF”.**

**#A variavel fica apenas dentro da funçao não retorna programa principal.**

    $string .= "DEF";

   }

**#Uma Variavel chamada str inicia com valor “ABC”.**

   $str = "ABC";

**#Invoca a função add\_extra passando como parametro a variavel str.**

add\_extra($str);

**#imprima na tela o conteudo da variavel str.**

echo $str;

#PROGRAMA FUNCIONANDO

**#criação da funçao add\_extra que recebe como parametro uma variavel chamada string.**

Function add\_extra($string) {

**#retorna “DEF” para o programa principal.**

 return "DEF";

}

**#Uma Variavel chamada str inicia com valor “ABC”.**

$str = "ABC";

**#Invoca a função add\_extra passando como parametro a variavel str e concatena seu retorno com conteúdo anterior.**

$str.=add\_extra($str);

**#Imprime na tela o conteudo da variavel str.**

echo $str;

# Questão 3

<?php

Function listarImagens($saidaDoComando) {

    $linhas = (explode("\n",$saidaDoComando));

    $conteudo = separaLinhas($linhas);

    echo("<table border = 3>");

        $tableHeader = tableHeader($conteudo[0]);

        $tableHeader = tableBody($conteudo);

    echo("</table>");

}

function separaLinhas($linhas){

    for ($i=0;$i<=count($linhas)-1; $i++){

        $conteudo[$i] = (explode(";",$linhas[$i]));

    }

    return $conteudo;

}

function tableHeader($conteudo){

    echo("<tr>");

    for($i=0; $i <= count($conteudo)-1; $i++){

        echo("<th>".$conteudo[$i]."</th>");

    }

    echo("</tr>");

}

function tableBody($conteudo){

    for($i=1; $i <= count($conteudo)-1; $i++){

        echo("<tr>");

        for($j=0; $j <= count($conteudo[$i])-1; $j++){

        echo("<td>".$conteudo[$i][$j]."</td>");

        }

        echo("</tr>");

    }

}

 $quadro = ("REPOSITORY;TAG IMAGE ID;CREATED;SIZE

 mysql 5.7;b598110d0fff;6 weeks ago;435MB

 wordpress 5.3.2-php7.2-apache;6669a7a94d1b;2 months ago;539MB");

 ListarImagens($quadro);

 ?>

# Questão 4

<?php

class Pessoa{

    private $nome;

    function \_\_construct($nome)

    {

        $this->nome = $nome;

    }

    function getNome(){

        return $this->nome;

    }

}

class Paciente extends Pessoa{ }

class Medico extends Pessoa{ }

class ReceitaMedica{

    private $medico;

    private $paciente;

    private $receita = Array();

    function \_\_construct($medico,$paciente)

    {

        $this->medico = $medico;

        $this->paciente = $paciente;

    }

    function \_\_destruct()

    {

        echo("Médico: ".$this->medico->getNome() . "</br>");

        echo("Paciente: ".$this->paciente->getNome(). "</br>");

        echo ("<br/>RECEITA <br/>");

        for($i=0;$i<=count($this->receita)-1;$i+=2){

            print\_r($this->receita[$i]->getNomeMedicamento()." ");

            print\_r($this->receita[$i+1]->getQuantidade());

            print\_r($this->receita[$i+1]->getUnidadeMedida()."<br/>");

        }

    }

    function adicionar($medicamento,$dosagem){

        array\_push($this->receita,$medicamento,$dosagem);

    }

}

class Medicamento{

    private $nome;

    function \_\_construct($medicacao)

    {

       $this->nome = $medicacao;

    }

    function getNomeMedicamento(){

        return $this->nome;

    }

}

class Dosagem{

    private $quantidade;

    private $unidadeMedida;

    function \_\_construct($quantidade,$unidadeMedida)

    {

        $this->quantidade = $quantidade;

        $this->unidadeMedida = $unidadeMedida;

    }

    function getQuantidade(){

        return $this->quantidade;

    }

    function getUnidadeMedida(){

        return $this->unidadeMedida;

    }

}

$receitaMedica = new ReceitaMedica(new Medico("Flavio"),new Paciente("Silva"));

$receitaMedica-> adicionar(new Medicamento("Aspirina"),new Dosagem(20, "mg"));

?>