

Módulo 2 - PHP | Exercícios consolidação

Temas abordados

- Estruturas de decisão e repetição
- Sessões
- Arrays

Parte 1

Pasta “parte01”

No ficheiro “index.php” escreva o código PHP necessário para escrever a mensagem “Olá mundo!” nos estilos decrescentes de *heading 1* a *heading 5*. De seguida, escrever a mesma mensagem com os mesmos estilos, mas na ordem crescente de *heading 5* a *heading 1*.

Escreva uma solução devidamente otimizada para evitar repetir código.

http://labmm.clients.ua.pt/deca_21L4/deca_21L4_p/PHP_sistematizacao/parte01/

Parte 2

Pasta: “parte02”

Observe com atenção os conteúdos do ficheiro “index.php”.

Garanta que a página destino dos links fornecidos apresenta os seguintes comportamentos para o parâmetro “x”:

- Valor “2243”: escreve parágrafo com “Que produto tão bonito”
- Valor “193”: escreve parágrafo com “Que produto fantástico”
- Valor “4267”: escreve parágrafo com “Que produto tão foleiro”
- Para outros valores: escreve parágrafo com “Produto desconhecido”

Caso não seja passado um parâmetro “x” então deve escrever um parágrafo com “Alguma coisa correu mal...”.

Inclua na página destino um link para a página “index.php”. Quando o valor do parâmetro “x” for válido, na página “index.php” deve mostrar um parágrafo com o formato “O valor escolhido anteriormente foi [valor anterior]”. Inclua esse parágrafo no final da página.

http://labmm.clients.ua.pt/deca_21L4/deca_21L4_p/PHP_sistematizacao/parte02/

Parte 3

Pasta “parte03”

No ficheiro “index.php” escreva o código PHP necessário para implementar a seguinte lógica, dependendo do número introduzido no formulário existente na página:

- Se o número for positivo deve mostrar a mensagem “número positivo” e aplicar a classe “alert-info”;

- Se o número for negativo deve mostrar a mensagem “número negativo” e aplicar a classe “alert-warning”;
- Se o número for zero deve mostrar a mensagem “número inválido” e aplicar a classe “alert-danger”.

Ignore os erros gerados caso o valor introduzido no formulário não seja um número.

http://labmm.clients.ua.pt/deca_21L4/deca_21L4_p/PHP_sistematizacao/parte03/

Parte 4

Pasta: “parte04”

Observe com atenção os conteúdos do ficheiro “index.php”.

Na página “maquina.php” deve validar a existência dos parâmetros relativos ao valor inicial e ao valor final. Se não existirem, imprima a mensagem “Avaria na máquina :(”, no formato de *heading 1*.

Caso existam, imprima os números desde o valor inicial ao final, utilizando um parágrafo para cada número. Implemente uma solução funcional prevendo a possibilidade do número final ser inferior ao número inicial. Por exemplo, se o valor inicial for 5 e o valor final for 3, o resultado a imprimir deve ser:

5

4

3

Se os números inicial e final forem iguais deve imprimir a mensagem “brincalhão :P”, no formato *heading 2*.

Inclua na página destino um link para a página “index.php”.

http://labmm.clients.ua.pt/deca_21L4/deca_21L4_p/PHP_sistematizacao/parte04/

Parte 5

Pasta: “parte05”

Na página “index.php” implemente uma lógica que, ao visitar múltiplas vezes a página a partir do mesmo *browser*, o valor impresso na página é “BINGO”, apenas de 5 em 5 tentativas. Nas restantes tentativas o valor a imprimir é “NOK”.

Escreva esses textos no formato de *heading 1*.

O mecanismo a implementar não deve utilizar a query string.

Inclua um link para a própria página com o texto “Voltar a tentar”.

http://labmm.clients.ua.pt/deca_21L4/deca_21L4_p/PHP_sistematizacao/parte05/

Parte 6

Pasta: “parte06”

Gere um número aleatório entre 1 e 10 utilizando a seguinte função do PHP:

```
rand(número_mínimo, número_máximo);
```

Crie um array com os seguintes números primos: 2, 3, 5, 7, 11, 13, 17, 19

Imprima na página os seguintes resultados:

- Cabeçalho em formato *heading 1* com “Tabuada do número x”, sendo “x” o número aleatório que foi gerado;
- Tabuada no formato apresentado na imagem, para o número aleatório gerado;
- Soma final dos resultados da tabuada anterior.

| Tabuada do número 3 | |
|---------------------|----|
| 3 * 2 = | 6 |
| 3 * 3 = | 9 |
| 3 * 5 = | 15 |
| 3 * 7 = | 21 |
| 3 * 11 = | 33 |
| 3 * 13 = | 39 |
| 3 * 17 = | 51 |
| 3 * 19 = | 57 |
| 231 | |

http://labmm.clients.ua.pt/deca_21L4/deca_21L4_p/PHP_sistematizacao/parte06/

Parte 7

Pasta: “parte07”

Observe os conteúdos da pasta “imgs”.

Na página “index.php” crie um array e armazene o nome de todas as imagens disponibilizadas, sem alterar o nome original das imagens.

Insira o código PHP necessário para mostrar todas as imagens na página, a partir da informação introduzida no array anterior. Junto a cada imagem deve escrever o respetivo nome do ficheiro. Separe as imagens através de um elemento <hr>. Escreva uma solução devidamente otimizada para evitar repetir código.

http://labmm.clients.ua.pt/deca_21L4/deca_21L4_p/PHP_sistematizacao/parte07/