



Lista de Exercícios

// Prática Individual

OBS: para execução dessa atividade é necessário ter: Pilha LAMP, WAMP ou XAMP, ou fazer a instalação do PHP em versões posteriores a 7.0 e também um servidor web seja ele apache ou nginx.

- Com base na lista anterior, dentro do repositório criado anteriormente **aceleracao-dev** . Salve seus arquivos na pasta criada anteriormente com o nome “PHP basico”. Dentro dela salve os arquivos de cada exercício.

****Atenção para cada exercício crie um arquivo com a nomenclatura: exercicionumerodoexercicio.php.**

46 - Dado o array abaixo, escreva um programa que mostre o nome, idade e o sexo (F ou H) de cada um dos individuos presente no arra. Em seguida , crie um array com os individuos do sexo masculino e outro com os do sexo feminino. Oobservação: separe a letra após a “,” onde consta (m ou f) que indica respectivamente qual o sexo do indivíduo, use esse valor para definir uma condicional e seprar os arrays, lembre-se de usar a função de separação);

```
$clientes = array("Joaquim" => "28,m", "Maria" => "41,f", "Pedro" => "18,m", "Joana Silva" => "58,f", "João Lopes" => "21,m", "Joana" => "15,f", "Fernanda" => "25,f");
```

47 - Crie um script capaz de gerar um arquivo chamado cartao.txt, dentro dele grave as informações consumidas da url usando o comando `get_file_contents` (https://random-data-api.com/api/business_credit_card/random_card), grave tudo com o formato de array dentro do arquivo. Em seguida leia o arquivo e transforme novamente a informação em JSON. Depois com o json convertido exiba na tela todas as informações usando `var_export`.

Programa de Aceleração para Programadores em PHP



48 - Dado o conjunto de urls (https://random-data-api.com/api/stripe/random_stripe, https://random-data-api.com/api/stripe/random_stripe, https://random-data-api.com/api/app/random_app, https://random-data-api.com/api/beer/random_beer, https://random-data-api.com/api/commerce/random_commerce, https://random-data-api.com/api/device/random_device, https://random-data-api.com/api/subscription/random_subscription). Crie um array com todas as urls do conjunto mencionado anteriormente, em seguida faça uma requisição para cada uma das urls usando o `file_get_contents`. Crie um arquivo chamado `api.txt`, dentro dele adicione o conteúdo de cada uma das requisições feitas com os dados convertidos em array, lembre-se de abrir e fechar o arquivo e escrever no final dele a cada nova requisição, preservando o conteúdo da requisição anterior. Use “\r\n” para realizar a organização das informações. Em seguida leia o arquivo `api.txt` e exiba o seus respectivos valores convertidos para json.

49 - Usando um laço de repetição, crie diretórios (Pastas) com a numeração de 1 a 150 e exiba uma mensagem cada vez que um diretório for criado. Em seguida, crie dentro dos diretórios ímpares um arquivo com a nomenclatura `log-{data-e-hora}.txt` e exiba uma mensagem contendo um feedback ao usuário de sucesso na criação do arquivo, em seguida exiba as informações do arquivo criado, tais como tamanho do arquivo em bytes e data de modificação, e também todas as infos disponíveis no `pathinfo` do arquivo.

50 - Usando `scandir`, leia todos os diretórios criados de 1 a 150 no exercício anterior. Escreva uma mensagem sempre que um diretório estiver vazio, caso ele esteja vazio exclua ele e mostre uma mensagem ao usuário. Para os diretórios que estiverem com arquivos dentro, crie um array e adicione o filename de cada um dos arquivos dentro do array usando `array_push`, por fim exiba as informações do array.

51 - Crie 4 cookies. Dentro de cada 1 deles defina um hash gerado usando SHA-256, como referência utilize o site (<http://andti.com.br/tool/hash>) . Defina 2 cookies com 1 hora de validade e os outros dois com 2 horas. Em seguida, recupere as informações dos cookies gerados e armazene-os em um arquivo chamado `cookies.txt`. Lembre-se de converter essas informações para um formato json antes de gravar no arquivo.

Programa de Aceleração para Programadores em PHP



52 - Crie 2 arrays 20 posições cada, dentro de cada um deles atribua números inteiros aleatórios. Em seguida, pegando o valor correspondente a posição de ambos os arrays, crie uma função que receba 2 valores inteiros como argumentos e retorne a sua soma. Se o valor da soma for negativo o método deverá retornar o valor 0. Em seguida crie um arquivo chamado arrays.txt e armazene os dois números inteiros passados na operação, separados por virgula juntamente com o valor retornado pela função.

53 - Crie 2 arrays **`$n1 = array()`** e **`$n2 = array()`** com 16 posições cada, onde a chave de cada um dos arrays será o nome de um aluno e o valor será sua nota de n1 ou n2, lembre-se de atribuir o mesmo nome nos dois arrays para corresponder ao mesmo aluno. Em seguida, crie uma função que receba 2 notas (N1 e N2) de um aluno e retorne um booleano indicando se o aluno passou. Para passar, a soma das notas deve ser igual ou superior a 19 e ambas devem ser superiores a 7. Escreva em um arquivo de texto chamado aluno-aprovados.txt somente o nome dos alunos aprovados.