



1. A partir do arquivo **register.php**, que está no campus:
 - a. Adicionar um campo de "Confirmar senha"
 - b. Adicionar funcionalidade para que, caso um parâmetro com o nome "**versaoCurta**" seja recebido por GET, não seja exibido o campo de confirmação de senha.
 - c. Escrever em PHP um array com no mínimo 10 países. Com isso, incluir uma tag <select> para que o usuário possa escolher uma nacionalidade. As opções devem ser exibidas de forma dinâmica de acordo com o conteúdo do array.
2. Usando **GET** como método, criar um arquivo chamado **inscricao_usuarios.php**, contendo um pequeno formulário de inscrição, e que, depois da inscrição de um novo usuário,
 - a. envie os dados a uma nova página chamada "**confirmacao.php**".
 - b. Dentro dessa página, mostrar na tela todos os dados inseridos, respeitando uma determinada aparência, por exemplo:
Agradecemos a sua inscrição, NOME COMPLETO. Você disse que tem X anos (mostrar idade). Registramos o seu e-mail: (mostrar EMAIL). Obrigado!
 - c. Para elementos como checkboxes, levar em conta que o elemento pode não existir ou pode ter vários valores (que podem ser resolvidos com o uso de **IF** ou **FOR**)
3. Modificar **inscricao_usuarios.php** e **confirmacao.php** para que enviem e processem os dados pelo método POST para evitar problemas.
4. Para testar, fazer com que o formulário envie os dados ao mesmo arquivo (**inscricao_usuarios.php**). Depois disso, fazer com que o formulário já apareça carregado com os campos que o usuário preencheu (exceto a senha), caso o formulário já tenha sido enviado.
5. Modificar **confirmacao.php** de forma que, caso não haja dados enviados por post, redirecione para **inscricao_usuarios.php**.





6. Utilizando `strpos()`, encontrar a posição da primeira aparição de "php" dentro da cadeia: "Adoro php, Eu também adoro php!"
7. Através de PHP, confirmar **register.php** a partir do servidor. A validação deve ser feita no mesmo arquivo. Se ela for bem-sucedida, redirecionar para **parabens.html**, que mostrará uma mensagem de parabéns. Em caso de erro, o arquivo deve indicar quais são os erros e deve preencher o formulário com os campos que já foram enviados.

Exercícios complementares

1. Criar um arquivo **faq.php**, com uma série de perguntas e respostas. Fazer com que elas estejam definidas como conteúdo de um array. Em seguida, imprimir o conteúdo HTML a partir de um **FOR** que percorra as perguntas e respostas.

Exemplo de array:

`<?php`

```
$faq = [  
    ["pergunta" => "Quanto é 1 + 1?", "resposta" => "2"],  
    ["pergunta" => "Quanto é 2 + 2?", "resposta" => "4"]  
];
```

`?>`

2. A partir de um arquivo com a linha `$cor = array('Branco', 'Verde', 'Vermelho')`, escrever um script PHP que mostre as cores desta forma:
Branco, Verde, Vermelho:
 - Branco
 - Verde
 - Vermelho
3. Modificar o exercício anterior para que cada elemento do array tenha o nome e o valor hexadecimal (css) da cor. Assim, seria possível desenhar cada nome com a cor indicada. Para isso, recomendamos utilizar o atributo



style e a **propriedade color** para imprimir o valor hexadecimal necessário.

4. Criar um array em que seja especificado o nome de um artigo e o respectivo preço. Escrever um script PHP para mostrar a sequência (os valores) dentro de uma tabela, com o nome na coluna esquerda e o preço na coluna direita.

Output:

Preço Item A	\$23421
Preço Item B	\$423
Preço Item C	\$8289