

JS Moderno (ES6+)

1. Dentro de um arquivo chamado *funcoes.js* faça o seguinte:
 - a. Crie uma função convencional, nomeada, chamada `parOuImpar1(numero)`, que retorne a *string* 'par' se `numero` for par e 'ímpar' caso `numero` seja ímpar. Para isso utilize o operador módulo (%) a fim de testar se o resto da divisão de `numero` por 2 é 0 (neste caso o número é par, caso contrário é ímpar). Faça o teste utilizando `console.log()`.
 - b. Crie uma *arrow function* chamada `parOuImpar2` que use `return` e tenha a mesma funcionalidade da questão anterior e faça o teste com o `console.log()`.
 - c. Faça uma terceira versão chamada `parOuImpar3` que utilize o operador de decisão ternário `?:` e retorno implícito, ou seja, sem escrever a palavra `return`. Teste com `console.log()`.
2. Crie um arquivo *form.html* que:
 - a. Crie um formulário em HTML com os seguintes campos de `<input>`: `nome`, `endereço`, `email`, `nascimento`, `salário`, `senha` e `conf_senha`.
 - b. Utilize os tipos corretos de campos como `text`, `email`, `date`, `number` e `password`.
 - c. Deve haver um botão `Submeter`. Quando ele for clicado uma função em JavaScript deverá ser executada. Esta função deverá verificar se os campos `senha` e `conf_senha` são iguais. Caso não sejam deve mostrar um `alert()` avisando o usuário sobre isso e então sair da função. Se forem iguais, proceder na letra (d).
 - d. A função deve criar um objeto literal chamado `pessoa` contendo os atributos `nome`, `endereço`, `email`, `nascimento`, `salário` e `senha`. Atribua os valores dos campos do formulário às propriedades do objeto.
 - e. Utilize o método `console.table()` para listar o objeto `pessoa` já populado no console.
3. Crie uma página HTML chamada *aneis.html* com o seguinte
 - a. Um script JS com um objeto literal chamado `sa` (abreviatura de Senhor dos Anéis), contendo, por sua:
 - Pares chave-valor descritos abaixo:
 - `nome`: 'Senhor dos Anéis'
 - `escritor`: 'J. R. R. Tolkien'
 - `diretor`: 'Peter Jackson'
 - `hobbits`: Lista contendo os nomes 'Merry', 'Pippin', 'Frodo', 'Sam' e 'Bilbo'
 - `magos`: Lista contendo os nomes 'Gandalf', 'Radagast' e 'Saruman'
 - Métodos:
 - `listarHobbits`: Deverá usar o método `forEach()` para percorrer a lista de hobbits e imprimir no console: 'o filme ' + `this.nome` + ' tem um Hobbit chamado ' + `<nome do hobbit>`. Dica: utilize uma *arrow function* dentro do `forEach()` para evitar problemas com o `this`.
 - `listarMagos`: Deverá usar a instrução `for .. of` para percorrer a lista de magos e imprimir no console: 'o filme ' + `this.nome` + ' tem um mago chamado ' + `<nome do mago>`.
 - `criarResumo`: Deverá **retornar** uma *template string* que monta um texto com as propriedades do objeto (`nome`, `escritor` e `diretor`) assim como as listas de hobbits e magos. Dica: Para gerar uma *string* a partir de um *array* pode-se usar o método `join()`.
 - b. Uma `div` chamada `resumo`.
 - c. Um botão que, ao ser clicado, irá mostrar um texto na `div`. O texto a ser mostrado na `div` será um resumo do objeto literal acima. Esse texto de resumo será obtido a partir do método `criarResumo` criado anteriormente. Use o DOM (`querySelector` ou `getDocumentById`) para recuperar a referência da `div` em uma variável chamada `resumo` e então o comando `resumo.innerHTML = sa.criarResumo()` para atribuir o texto do resumo à `div`.
4. Crie uma função em JS chamada `multiplicar()` com 2 parâmetros: `x` e `y`. Utilize o recurso de parâmetros *default* para atribuir o valor 1 aos parâmetros caso não seja passado um valor a eles na chamada da função. Chame a função passando os 2 valores, não passando valor algum, passando só o valor de `x` e passando só o valor de `y` para ver se ela está funcionando corretamente.