

Trabalho 1 de JavaScript

Nome:

Em todas questões abaixo:

- coloque os códigos em uma página HTML padrão com bootstrap ou materialize e mostre os elementos pedidos conforme o visual destes frameworks.
- as páginas devem passar pelo validador da w3c sem erros (<https://validator.w3.org/>).
- as páginas não devem apresentar erros pegos pelo console do JavaScript(JS) dos navegadores.
- Todas alterações na página feitas pelo JS devem gerar uma nova página que não apresenta erros captados pelo validador da w3c ou pelo console. (As páginas serão corrigidas por um do debug JavaScript completo)
- Utilizar let em todas declarações de variável. Todas variáveis devem ser declaradas.

Questões referentes as aulas 1 e 2

1(1pt)- Em uma página HTML construa as seguintes funções e comandos JS

Construir uma função chamada declarationMed (function declaration) que recebe 3 números como parâmetro e retorna a média destes números

Chame a função passando os 3 números, guarde o retorno em uma variável e depois imprima a variável no console através do console.log(var) Para visualizar o console utilize a ferramenta do desenvolvedor do google chrome (aba console)

Construir uma função similar chamada anonimaMed só que na sintaxe anonima

Chame a função passando 3 números, guarde o retorno em uma variável e depois imprima a variável no console através do console.log(var)

Construir uma função similar chamada arrowMed só que na sintaxe arrow

Chame a função passando 3 números, guarde o retorno em uma variável e depois imprima a variável no console através do console.log(var).

2 (1,5pt)- Fazer um formulário com os campos abaixo. Ao clicar no botão submeter uma função JS deve ser chamada. Esta função deverá fazer as seguintes operações nos campos abaixo:

Nome: verificar se foi preenchido.

CPF: verificar se foi preenchido e se foram digitados 11 caracteres.(utilizar a propriedade .length das strings)

Idade: Verificar se é um numero entre 0 e 100

No final a função escreve todos os erros na página logo abaixo do formulário. Além disso, a função deve pintar as bordas dos campos errados de vermelho (garanta que os campos errados fiquem com a borda vermelha quando estiverem errados e sem borda vermelha quando estiverem corretos, penso nos casos em que se envia e se corrige os campos várias vezes ao corrigir e enviar os dados várias vezes). Por fim, ao corrigir o preenchimento dos dados a função deve deixar o foco em algum campo com erro.

3 (1pt)- Em um campo texto perguntar quantos alunos quer cadastrar. Ao clicar um botão, o JS deve criar e mostrar na tela a quantidade de campos texto informada (Utilize document.createElement(".."), document.createTextNode(".."), appendChild(...), etc.) e um novo botão. Ao clicar no novo botão o JS deve imprimir na tela o nome dos alunos inscritos (os nomes dos alunos que foram colocados nos campos texto da etapa anterior)

Questão referente a aula 3

4(2,5pt)- Dado o código de um estoque do arquivo estoque.html disponibilizado na aula 3, faça as seguintes alterações.

- a- Altere o HTML, acrescente CSS e bootstrap para melhorar o visual da página.
- b- Crie um segundo objeto chamado carro que irá armazenar os produtos que serão comprados. Para cada produto ele irá armazenar o nome e o respectivo preço.
- c- Abaixo da seção de estoque coloque uma seção “Carrinho de compra”. Esta seção será montada na função mostra(). Ou seja, cada vez que a função mostra () for chamada ela irá mostrar os atuais produtos no objeto estoque e os atuais produtos no objeto carro.
- d- Atualmente o botão excluir de cada item do estoque apenas exclui um item do estoque. Neste trabalho ele deve excluir um item do objeto estoque e depois coloca-lo no objeto carro. O nome mais apropriado do botão seria “Adicionar ao carrinho de compras”.
- e – O carrinho de compras será uma seção semelhante ao estoque. O botão que fica ao lado de cada produto do carrinho deve se chamar “Retirar do carrinho”. Este botão tira o produto do objeto carro e o coloca no objeto estoque.

Questões referentes a aula 4

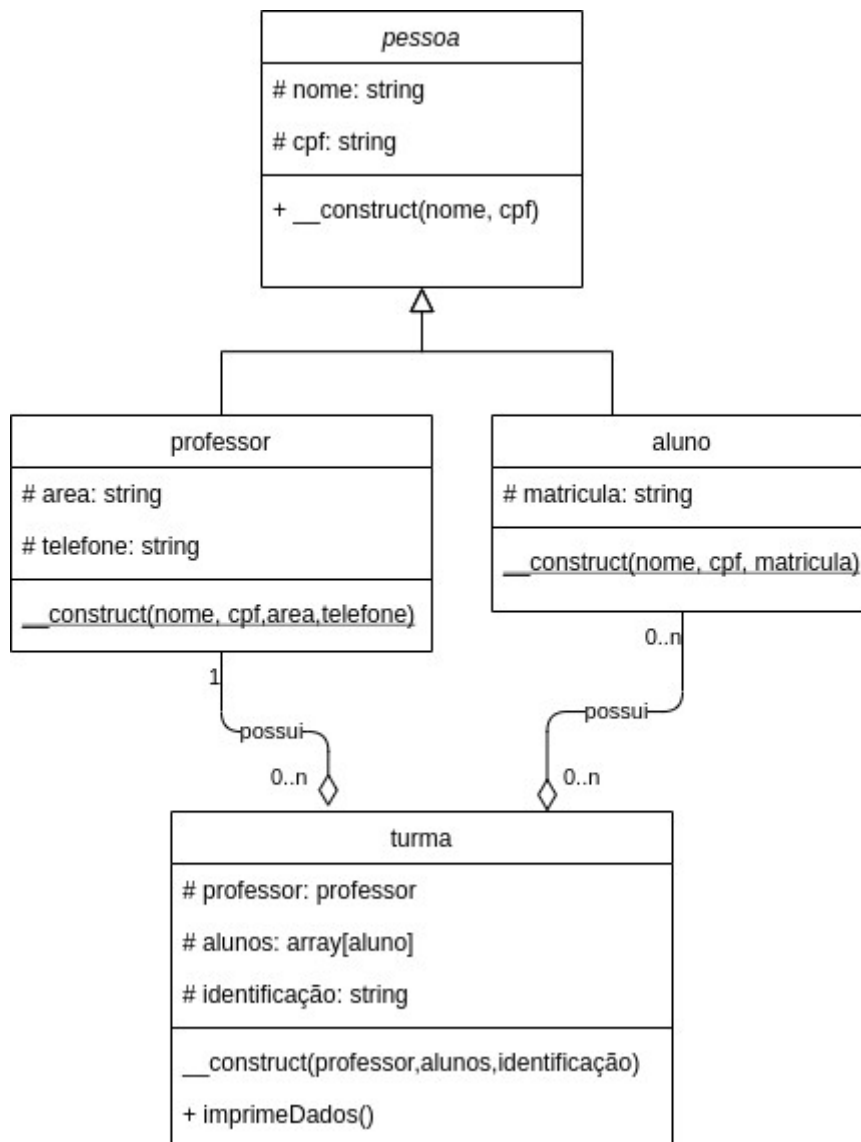
5(2pt)- Em uma página HTML com JS:

- Criar a classe pessoa(cpf,nome), herdada pelas classes professor(area, telefone) e aluno(matricula)
- criar a classe turma(professor, alunos,identificação)
- criar métodos constructs para todas classes
- criar o método imprimeDados na classe turma. Este método imprime todos os dados da turma incluindo os dados do professor e de todos os alunos da turma.

Depois, diretamente no código:

- criar 2 objetos professores
- criar 6 objetos alunos
- criar 2 objetos turmas de 4 alunos
- chamar o método imprime dados das duas turmas. A impressão deve ser feita em uma página HTML5 padrão com bootstrap. Os dados da turma, professor e alunos devem estar em tabelas

A figura abaixo mostra o diagrama de classe deste exercício



6(2pt)- Faça agora uma versão do exercício 5 aonde:

- Seja mantida a mesma estrutura de classes;
- Em um formulário é solicitado o nome, cpf e matricula de um aluno. Com estes dados um objeto aluno deve ser criado e armazenado em um array de alunos. Deve ser possível acrescentar quantos alunos o usuário quiser. Ou seja cada vez que clicar em um botão “Adicionar aluno” um objeto aluno é adicionado ao array de alunos.
- Em outro formulário é solicitado o nome, cpf, area e telefone de um professor. Com estes dados um objeto professor deve ser criado.
- Em outro formulário é solicitada a identificação de 1 turma. Ao clicar em um botão “Criar turma” é criado um objeto turma. Ao criar este objeto deve ser passado ao seu construtor a identificação da turma, o objeto professor armazenado anteriormente e o array de alunos armazenado anteriormente. Após criar o objeto turma, o método `imprimeDados` da turma deve ser chamado. Este método irá mostrar na página todos dados da turma, de seu professor e de seus alunos.
- Todos estes formulários e dados devem ser mostrados na mesma página