**TESTE C#**

**- Usuário**

+ id,

+ nome

+ data de nascimento

+ email

+ senha

+ foto de perfil

**- Tarefas**

+ id

+ descricao

+ data/hora

**Com base nas informações acima, faça:**

**- Deve conter as rotas as seguintes ações:**

**- Gerenciamento de usuários (Cadastro, atualização, busca por vários, buscar por id e logar no sistema) (deve conter um usuário que tenha permissão de visualizar todos, caso contrário, o usuário pode visualizar somente suas informações)**

**- Gerenciamento de tarefas (Cadastro, atualização, busca por vários, buscar por id e remoção)**

**- Deve conter tratamento de dados, retornando o "http code" correto para os erros ou não erros;**

**- Deve conter pelo menos um teste (unitário, api ou outros).**

**- O usuário deve "confirmar" o seu cadastro com um código enviado por e-mail.**

**- Crie uma rota para deletar um usuário, essa rota deve deletar também todas as tarefas associadas a esse usuário.**

**Banco**

**- Anexar em uma pasta no projeto os scripts das criações das tabelas**

**- Criar uma trigger que será disparada ao cadastrar um novo usuário no banco, deverá enviar um e-mail para "contato@sistemas.com.br".**

**- Criar uma procedure que envie um e-mail informando a quantidade de usuários cadastrados nos últimos 92 dias;**

**- Criar um diagrama de banco de dados**

**- Criar uma procedure que envie um e-mail ao aniversariante do dia.  
  
Utilize Javascript**

**1** **Crie uma função que receba um objeto JSON e imprima no console o nome de todas as propriedades do objeto**

**2 Crie uma função que receba um array de objetos JSON e verifique caso o objeto possua a propriedade cidade e o valor seja Belo Horizonte acrescente a propriedade estado com valor MG e retorne o array**

**3 Crie uma função que receba um array de objetos JSON e verifique caso o objeto possua a propriedade cidade e o valor seja São Paulo imprima o objeto no console, remova o objeto e retorne o array**

**4 Crie uma variável que possa ser usada somente no escopo em que foi criada e uma que possa ser usada fora do escopo que foi criada**

**5 Crie uma que receba um array de objetos e retorne ordenado e ordem alfabética a partir da propriedade name do objeto**

**6 crie uma função que receba um array e retorne um novo array somente com números pares contidos nele**

**7 crie uma função que receba um array e retorne um novo array somente com números primos contidos nele**

**8 crie uma função que remova o quinto elementos do array caso ele seja um número primo e imprima o total da soma do array no console**, **se o quinto elemento não for um número primo gere um novo array randômico de 20 posições e com números primos e imprima no console o array e assoma dos números dele.**

**9 Crie uma função que receba uma classe de um input html, capture o valor do input e formate em moeda brasileira R$ e imprima no console caso não seja um valor numérico imprima no console o valor é valido.**

**10 Crie uma função que receba um array de números e converta para todos para números inteiros arredondado para baixo e imprima no console.**

**11 – Utilizando o seguinte objeto**

const user = {

  name: "Roberto",

  idade: "33",

  formado: "Sim",

  escolaridade: {

    instituicao1: "UNIVERSIDADE MG, UNIVERSIDADE SP, UNIVERSIDADE PR, UNIVERSADE RS",

    instituicao2: "Preencher essa informação",

  },

}

**11.1 Crie uma função que receba o objeto e altere propriedade instituição 1 para um array de strings utilizando o separador virgula utilizando o valor da propriedade e adicione 2 Adicione 3 novas propriedade no objeto user endereço, nacionalidade e situação do curso.**   
  
**Utilizando jquery - Utilizando o exemplo de html abaixo:**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta http-equiv="X-UA-Compatible" content="IE=edge">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

    <body>

        <form action="" method="post">

            <div>

                <label for="nome">Nome:</label>

                <input type="text" id="Nome" />

            </div>

            <div>

                <label for="email">E-mail:</label>

                <input type="email" id="email" />

            </div>

            <div>

                <label for="msg">Mensagem:</label>

                <textarea id="msg"></textarea>

            </div>

            <div>

                <label for="msg">alertar:</label>

                <input type="checkbox" name="alertar" checked id="alert">

            </div>

            <div>

                <label for="msg">Enviar:</label>

                <input type="button" name="alertar" value="Enviar" id="send">

            </div>

        </form>

    </body>

</html>

**1 – Crie uma função que ao clicar no botão enviar imprima no console os valores dos seguintes inputs: Nome, E-mail e enviar**

**2 – Crie uma função que verifique o checkbox: alertar caso esteja checado e imprima no console “o usuário deve ser alertado”, caso não, deve imprimir “o usuário não deseja ser alertado”**

**3 – Crie uma função que que ao clicar no botão enviar valide os inputs caso eles estejam vazios a cor borda do input deve ficar vermelha**

**4 – Codifique um script que caso o campo salário não seja preenchido o botão enviar deve ser desabilitado**

**5 – Crie uma função que crie uma máscara monetária(R$) para salário.   
  
  
O projeto deve estar no seu GitHub pessoal como (Privado), e nos compartilhe o repositório apenas com os e-mail a baixo para avaliação.**[**cesar.mascedo@sysmap.com.br**](mailto:cesar.mascedo@sysmap.com.br) **– cesarmacedosysmap**[**flavio.carmo@sysmap.com.br**](mailto:flavio.carmo@sysmap.com.br) **– flaviodaniel**[**governanca.sistemas@sysmap.com.br**](mailto:governanca.sistemas@sysmap.com.br) **-**