**Questão 1**

A- Qual método http devo usar em um serviço para atualizar o nome de um

cliente em um cadastro?

**Método PUT**

B- Qual "status code" deve retornar para esta atualização?

**Status Code 200**

**Questão 2**

Faça uma análise do código abaixo e:

**Não possuo experiência com a linguagem Ruby. Mas conforme analisei, esta linguagem está utilizando o framework rspec e o HTTParty para realizar testes automatizados. Minha experiência com automação é com java e c#.**

A- Explique resumidamente cada bloco

B- Faça pelo menos uma sugestão de melhoria do código

#bloco 1: **Aqui acredito que está realizando o setup e configurações com o framework.**

require 'httparty'

RSpec.configure do |config|

config.expect\_with :rspec do |expectations|

expectations.include\_chain\_clauses\_in\_custom\_matcher\_descriptions = true

end

#bloco 2: **Criando a configuração do HTTParty**

config.mock\_with :rspec do |mocks|

mocks.verify\_partial\_doubles = true

end

config.shared\_context\_metadata\_behavior = :apply\_to\_host\_groups

config.color\_mode = true

config.before(:all) do

class HttParty

include HTTParty

end

end

#bloco 3: **Iniciando a automação API e validando se serviço está rodando com sucesso**

RSpec.describe 'Bloco 3' do

it 'Deve retornar 200 para a request valida' do

response = HttParty.get(' http://fakerestapi.azurewebsites.net/api/books')

expect(response.code).to eql(200)

end

it 'Deve retornar o content-type application/json; charset=utf-8' do

response = HttParty.get(' http://fakerestapi.azurewebsites.net/api/books')

expect(response.headers['content-type']).to eql('application/json;

charset=utf-8')

end

it 'O retorno não deve ser vazio' do

response = HttParty.get(' http://fakerestapi.azurewebsites.net/api/books')

expect(response.parsed\_response).not\_to be\_empty

end

#bloco 4: **Validando cada item da API**

it 'Bloco 4' do

response = HttParty.get(' http://fakerestapi.azurewebsites.net/api/books')

expect(response.parsed\_response).to be\_a\_kind\_of(Array)

response.parsed\_response.each do |item|

expect(item).to be\_a\_kind\_of(Object)

expect(item["ID"]).to be\_a\_kind\_of(Integer)

expect(item["Title"]).to be\_a\_kind\_of(String)

expect(item["Description"]).to be\_a\_kind\_of(String)

expect(item["PageCount"]).to be\_a\_kind\_of(Integer)

expect(item["Excerpt"]).to be\_a\_kind\_of(String)

expect(item["PublishDate"]).to be\_a\_kind\_of(String)

end

end

#bloco 5

it 'Bloco 5' do

response = HttParty.get(' http://fakerestapi.azurewebsites.net/api/books/1')

expect(response.parsed\_response).to be\_a\_kind\_of(Object)

expect(response["ID"]).to be\_a\_kind\_of(Integer)

expect(response["Title"]).to be\_a\_kind\_of(String)

expect(response["PageCount"]).to be\_a\_kind\_of(Integer)

expect(response["Excerpt"]).to be\_a\_kind\_of(String)

expect(response["PublishDate"]).to be\_a\_kind\_of(String)

end

#bloco 6:

it 'Bloco 6' do

body = {

'ID': 150,

'Title': "Titulo Livro teste api",

'Description': "Descrição teste api",

'PageCount': 255,

'Excerpt': "Resumo do livro",

'PublishDate': "2019-02-06T20:35:21.207Z"

}.to\_json

response = HttParty.post(' http://fakerestapi.azurewebsites.net/api/books',

body: body,

headers: {'Content-Type' => 'application/json'})

expect(response.code).to eql(200)

expect(response.parsed\_response).to eql(JSON.parse(body))

end

end

**Questão 3**

Considere a seguinte criação das tabelas Cliente e Pedido:

CREATE TABLE Cliente (

id INTEGER PRIMARY KEY AUTOINCREMENT,

nome varchar(255),

sobrenome varchar(255)

);

CREATE TABLE Pedidos (

id INTEGER PRIMARY KEY AUTOINCREMENT,

cliente\_id int(5),

produto varchar(255),

preco decimal(5,2)

);

Baseado na criação das tabelas acima, responda com a declaração SQL correta

para:

A - Criar uma Coluna de email na tabela Cliente

**Alter table Cliente**

**Add column email varchar (255)**

B - Inserir, 3 clientes na tabela Cliente com os seguintes dados:

-José Silva, email: jose@cb.com.br

-João Pedro, email: joao@pf.com.br

-Pedro Silva, email : pedro@ex.com.br

**insert into Cliente values ('José', 'Silva', 'jose@cb.com.br');**

**insert into Cliente values ('João', 'Pedro', 'joao@pf.com.br');**

**insert into Cliente values ('Pedro', 'Silva', 'pedro@ex.com.br');**

C – Inserir um pedido para cada cliente com os seguintes dados:

-Para o cliente 1: Produto: Geladeira Brastemp, Preço: 1800,00

-Para o cliente 2: Produto: Fogão Consul, Preço 850,90

-Para o cliente 3: Produto: Celular Iphone XR, Preço 3399,00

**insert into Pedidos values (1, 'Geladeira Brastemp', 1800);**

**insert into Pedidos values (2, 'Fogão Consul', 850.90);**

**insert into Pedidos values (3, 'Celular Iphone XR', 3399);**

D - Selecionar todos os pedidos de clientes com sobrenome Silva

**select \* from Pedidos P**

**left join Cliente C on C.id = P.cliente\_id**

**where c.sobrenome = 'Silva';**

E - Apagar a tabela Pedidos

**Drop table Pedidos;**

**Questão 4**

Considere a história de usuário:

Como cliente das Casas Bahia

Quero adicionar um produto no carrinho

Para reservar meu produto

Escreva em BDD (Behavior Driven Development) os 2 cenários abaixo:

A- Adicionar um Iphone no carrinho

B- Adicionar um ventilador com a opção 110v no carrinho

**Funcionalidade: Adicionar produtos no carrinho das Casas Bahia**

**Como um cliente**

**Quero adicionar produtos no carrinho**

**Para reservar um produto**

**Cenário: Adicionar um Iphone no carrinho**

**Dado** que eu acesse a página Casas Bahia

**Quando** eu clicar em comprar um Iphone

**Então** deve ser adicionado o produto no carrinho

**E** reservar meu produto

**Cenário: Adicionar um ventilador com a opção 110v no carrinho**

**Dado** que eu acesse a página Casas Bahia

**Quando** eu clicar em comprar um ventilador

**E** a voltagem de 110v

**Então** deve ser adicionado o produto no carrinho

**E** reservar meu produto

**Questão 5 – Em anexo.**

**Escolha a linguagem de programação de sua preferência para automatizar o fluxo**

**abaixo:**

**Funcionalidade:** Pesquisa com o QA

**Como** um recrutador

**Quero** colher dados da pesquisa

**Para** fazer análises mais detalhadas do QA

**Link:** https://bit.ly/3jOMrR9

**Cenário:** Preencher pesquisa com dados obrigatórios válidos

**Dado** que eu acesse a página da VV Test

**E** acesse o menu "Pesquisa - QA"

**Quando** eu preencher todos os campos obrigatórios

**Então** deve ser direcionado para uma página de sucesso