**Prática 3**

1. Configurar o VSCode para a realização da prática.

a) Abra o aplicativo VSCode. Certifique-se de estar na pasta onde deseja clonar o repositório (ex:. Documents).

b) Clone o seu repositório backend do GitHub, utilizando o terminal do VSCode. Caso o repositório já exista no seu computador, atualize-o com o repositório remoto.

c) Abra a pasta do repositório clonado no VSCode.

d) Crie dentro da pasta praticas uma pasta chamada pratica03.

e) Vá para a barra lateral esquerda do VSCode e clique no ícone de extensões (quatro quadrados).

f) Pesquise por "Thunder Client" e clique em "Install" para instalar a extensão.

2. Criar um teste para uma API REST.

a) Vá para a barra lateral esquerda do VSCode e clique no ícone do Thunder Client.

b) Clique no botão New Request para criar uma requisição para uma API REST.

c) Mantenha o método GET e informe a URL https://dummyjson.com/products.

d) Clique no botão Send. Verifique se o status da resposta foi 200 OK e se o conteúdo foi um JSON contendo um array products.

e) Mantenha o método GET e informe a URL https://dummyjson.com/products/1.

f) Clique no botão Send. Verifique se o status da resposta foi 200 OK e se o conteúdo foi um JSON contendo o produto com ID igual a 1.

g) Mantenha o método GET e informe a URL https://dummyjson.com/products/101.

h) Clique no botão Send. Verifique se o status da resposta foi 404 Not Found e se o conteúdo foi um JSON contendo uma mensagem.

i) Troque para o método POST e informe a URL https://dummyjson.com/products/add.

j) Clique na aba Body e crie um JSON contendo as propriedades title, description e price para um novo produto.

k) Clique no botão Send. Verifique se o status da resposta foi 200 OK e se o conteúdo foi um JSON contendo o produto com ID igual a 101.

l) Troque para o método PUT e informe a URL https://dummyjson.com/products/1.

m) Clique na aba Body e crie um JSON contendo as propriedades title, description e price para atualizar um produto.

n) Clique no botão Send. Verifique se o status da resposta foi 200 OK e se o conteúdo foi um JSON contendo o produto com ID igual a 1.

o) Mantenha o método PUT e informe a URL https://dummyjson.com/products/101.

p) Clique no botão Send. Verifique se o status da resposta foi 404 Not Found e se o conteúdo foi um JSON contendo uma mensagem.

q) Troque para o método DELETE e informe a URL https://dummyjson.com/products/1.

r) Clique na aba Body e apague o conteúdo do JSON.

s) Clique no botão Send. Verifique se o status da resposta foi 200 OK e se o conteúdo foi um JSON contendo o produto com ID igual a 1.

t) Mantenha o método DELETE e informe a URL https://dummyjson.com/products/101.

u) Clique no botão Send. Verifique se o status da resposta foi 404 Not Found e se o conteúdo foi um JSON contendo uma mensagem.

v) Clique na aba Collections do Thunder Client e acione o menu New Collection no ícone ☰.

w) Informe o nome para a coleção (Ex.: Testes da API REST) e pressione ENTER.

x) Clique na aba Activity do Thunder Client e acione o menu Save to Collection no ícone … de uma requisição.

y) Selecione em Collection a coleção criada anteriormente e clique em Submit. Repita os passos x e y para as demais requisições.

z) Clique na aba Collections do Thunder Client e acione o menu Export no no ícone … da coleção. Informe a pasta pratica03 como destino.

3. Subir as modificações no GitHub.

a) Adicione todas as modificações realizadas no projeto no stage do git.

b) Faça um commit com as modificações adicionadas.

c) Caso não esteja configurado os parâmetros user.name e user.email, configure-os através do git.

d) Envie o commit para o branch principal do seu repositório no GitHub.