```
//UTILIZANDO EXPRESS PARA CRIAR UM SERVIDOR SIMPLES:


//Todo serviço WEB precisa de ter uma porta de comunicação entre o lado cliente e servidor - Só relembrando, utilizamos portas para identificar á qual serviço web estamos referenciando para que haja uma diferenciação entre um determinado serviço web e outro serviço, cada um deverá ter a sua própria porta.
const porta = 3003

//Importamos o framework express...
const express = require('express')

//Abaixo nós criamos uma instância para o framework express por através do módulo "express()", assim tornamos possível a utilização dos módulos do framework por através da variável que receberá essa instância, que no caso é "app"...
const app = express()

//Abaixo estamos usando o método "get()" de express, esse método é responsável por gerar uma resposta utilizando o método get invés do post, onde conseguiremos capturar os valores contidos no caminho "/produtosGet", porém note que não atribuímos valor nenhum para ser enviado quando o método get for chamado, em vez disso, nós ativamos uma função next() que por padrão irá realizar um middleware, chamando a próxima função...
app.get('/produtosGet', (request, response, next) => { //get só pode responder a uma pesquisa get, nunca poderá responder a uma pesquisa post...
    next()
})

//Podemos também utilizar o método "use()", esse método não irá procurar por um caminho específico get ou post, em vez disso ele vai executar o comando que estiver dentro de use()...
app.use((request, response, next) => {
    response.send({nome: 'Celular', preco: 123.46}) //O método send() irá enviar uma resposta a requisão http desse arquivo...
})

//Agora temos um exemplo usando o método post...
app.post('/produtosPost', (request, response, next) => { //agora ele responderá a chamadas post, mas não responderá a chamadas get...
    response.send({nome: 'Notebook', preco: 123.46}) //Detalhe importante sobre send(), ele automáticamente transforma um objeto em JSON...
})
```

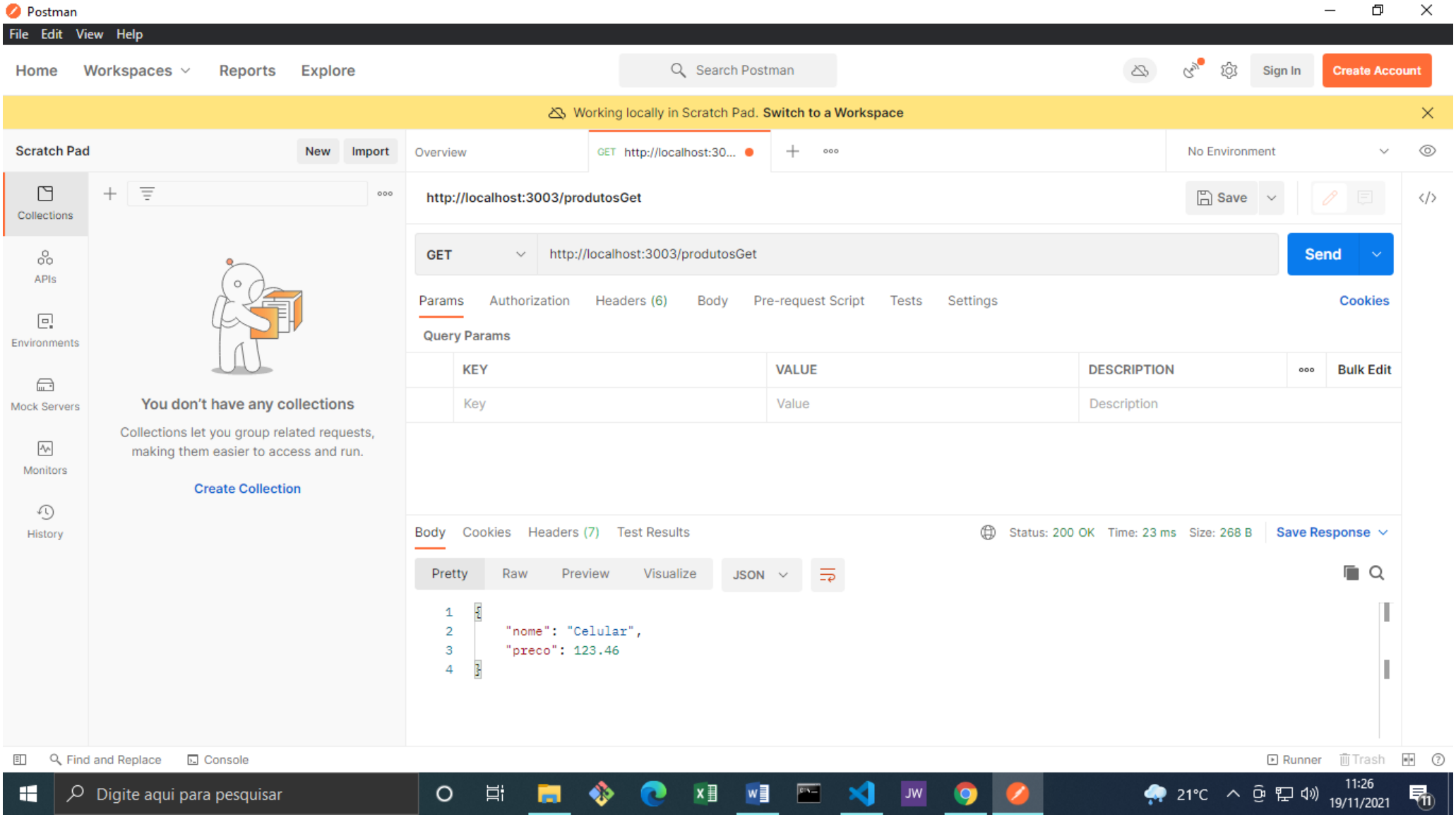
```
//Abaixo estamos usando o método "listen()", esse método é responsável por ativar o servidor sobre uma determinada porta, veja que
ele também recebe uma callback, que pode ser executada em conjunto com a abertura do servidor, nesse caso, nós imprimimos no console
e informação de que o servidor está ativo, e qual é a porta em que ele está ativo...
app.listen(porta, () => {
    console.log(`O servidor está ativo na porta ${porta}...`)
})

//Faça o exercício de executar essa porta, e depois a procure, seja no POSTMAN ou no browser por através do caminho:
http://localhost:3003/produtos, você vai ver que receberá como resposta o objeto de produtos, mas no formato JSON...
```

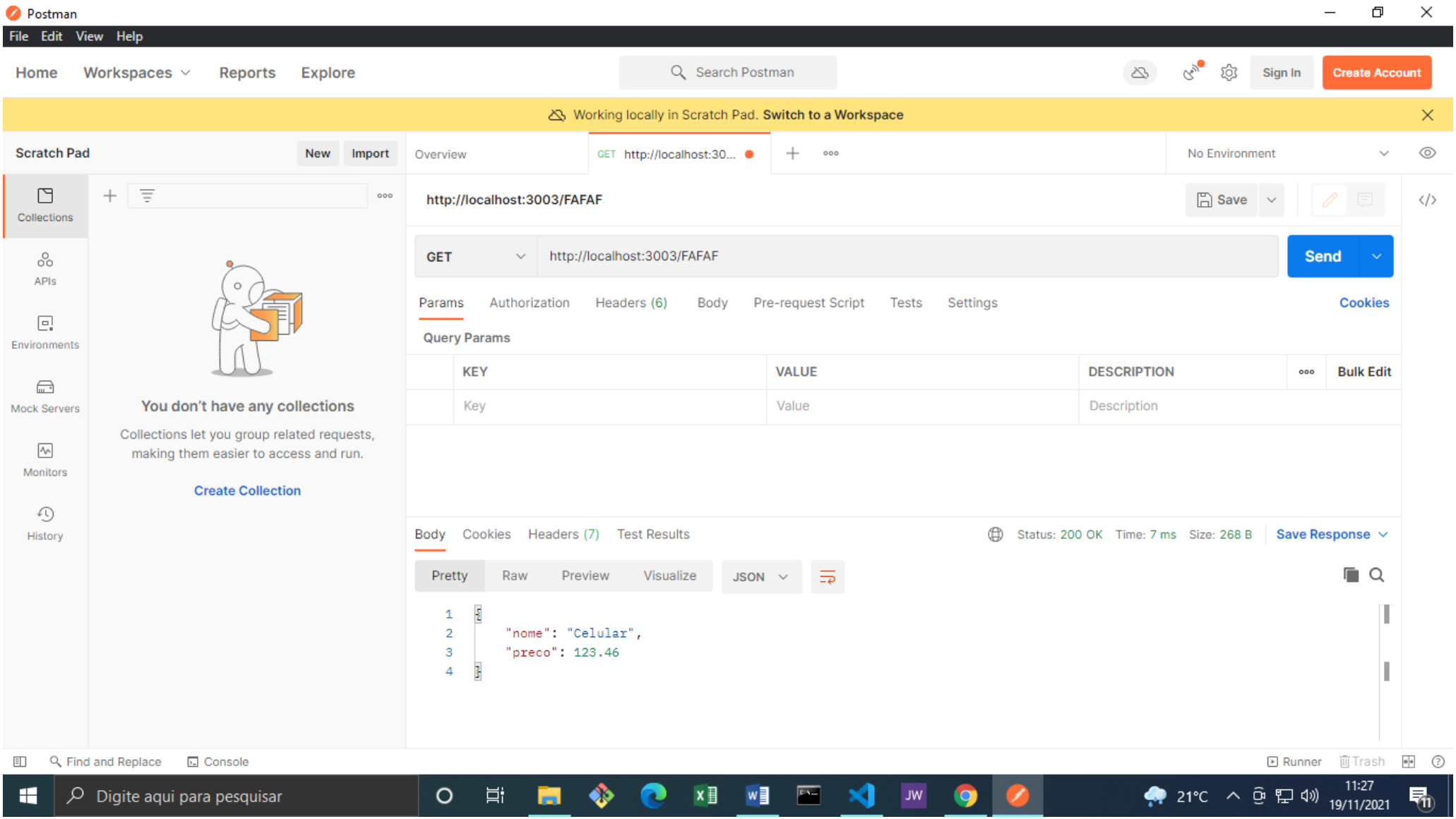
RESULTADO NO CONSOLE...

```
[Running] node
"c:\Users\Almoxarifado\Documents\javascript\arquivos_das_aulas\Projetos_Back_End\Projeto_API_Express\src\Criando_Servidor_com_Express
.js"
O servidor está ativo na porta 3003...
```

RESULTADO NO POSTMAN PARA O MÉTODO GET...



RESULTADO NO POSTMAN PARA O MÉTODO USE...



RESULTADO NO POSTMAN PARA O MÉTODO POST...