



| UNIFACS

Sistemas Distribuídos e Mobile.

Anderson Santos Neris Silva - 1272219303
André Neves dos Santos Braga - 12722126112
Pedro Martins Caires - 12722124034
Rafael Rodrigues Figueiredo - 12722130532
Sérgio Filho de Paula Fidelis 1272215886

2023

1- Descrição dos requerimentos de softwares necessários para execução da aplicação (linguagem, bibliotecas, etc).

obs: abaixo informaremos como instalar cada uma delas

Softwares:

- **Visual Studio Code** = O Visual Studio Code é um editor de código-fonte
- **MySQL** = é um sistema de gerenciamento de banco de dados relacional

Ferramenta:

- **Node.JS** = é a ferramenta que vai nos entregar a capacidade de interpretar código JavaScript, de maneira bem similar ao navegador

Linguagem:

- **JavaScript** = uma linguagem de programação que permite a você implementar itens complexos em páginas web — toda vez que uma página da web faz mais do que simplesmente mostrar a você informação estática — mostrando conteúdo que se atualiza em um intervalo de tempo, mapas interativos ou gráficos 2D/3D animados, etc.

2 - Justificativa da escolha da tecnologia

- Detalhamento da arquitetura, estratégia e algoritmos utilizados

A escolha estratégica de adotar uma API para os relatórios de produtos mais vendidos e produtos por cliente, por exemplo, é fundamentada em sua eficiência na comunicação de dados entre sistemas, garantindo uma integração ágil e segura. Além disso, a flexibilidade e escalabilidade da API se destacam, adaptando-se facilmente às mudanças no escopo do projeto. A centralização da lógica de negócios simplifica o desenvolvimento e a manutenção, enquanto as camadas de segurança asseguram o controle de acesso aos dados sensíveis. A capacidade de integração com ferramentas externas solidifica essa escolha, proporcionando uma base robusta para impulsionar a eficiência operacional e preparar a plataforma para um crescimento contínuo.

Ademais, na implementação da nossa API, a escolha pelo **JavaScript** se alinha perfeitamente com a natureza simplificada da aplicação, onde operações de baixa complexidade são executadas de forma eficaz. A agilidade do JavaScript, combinada com sua familiaridade, otimiza o desenvolvimento, permitindo que nossa equipe concentre esforços em aprimorar a experiência do usuário e a funcionalidade central dos relatórios que serão mostrados. Dessa maneira, conseguimos exibir as informações pertinentes sobre a compra dos produtos, como a quantidade do produto, a lista do usuário e compras, além dos produtos que estão com estoque baixo.

Para fins avaliativos da A3, o projeto foca na eficiência operacional. No contexto específico deste trabalho e aplicação, as operações são diretas e não demandam um tratamento de segurança intensivo, permitindo uma abordagem mais ágil no desenvolvimento da API. Essa simplificação, por consequência, contribui para uma experiência de implementação mais fluída, adequando-se às exigências do projeto de forma eficaz e direta.

Frameworks, Ferramentas e Database:

Express:

Express é um framework web para Node.js. Ele simplifica o desenvolvimento de aplicativos web e APIs ao fornecer uma estrutura mínima, mas poderosa, para lidar com rotas, middleware, solicitações e respostas HTTP.

Principais Características:

Rotas: Permite definir rotas para manipular solicitações HTTP.

Middleware: Facilita a adição de funções de middleware para processar solicitações antes de chegarem às rotas.

Simplicidade: É projetado para ser simples e flexível, facilitando a criação de aplicativos web rápidos e eficientes.

Nodemon:

Nodemon é uma ferramenta de desenvolvimento que monitora arquivos no projeto Node.js e automaticamente reinicia o servidor quando detecta alterações.

Principais Características:

Auto reinício: Elimina a necessidade de reiniciar manualmente o servidor após cada alteração no código, acelerando o desenvolvimento.

Fácil configuração: Pode ser instalado globalmente ou como uma dependência de desenvolvimento no projeto.

MySQL:

MySQL é um sistema de gerenciamento de banco de dados relacional amplamente utilizado. A biblioteca mysql para Node.js permite a interação com um banco de dados MySQL a partir de aplicativos Node.js.

Principais Características:

Conexão ao Banco de Dados: Permite estabelecer uma conexão com um servidor MySQL.

Execução de Consultas: Facilita a execução de consultas SQL no banco de dados.

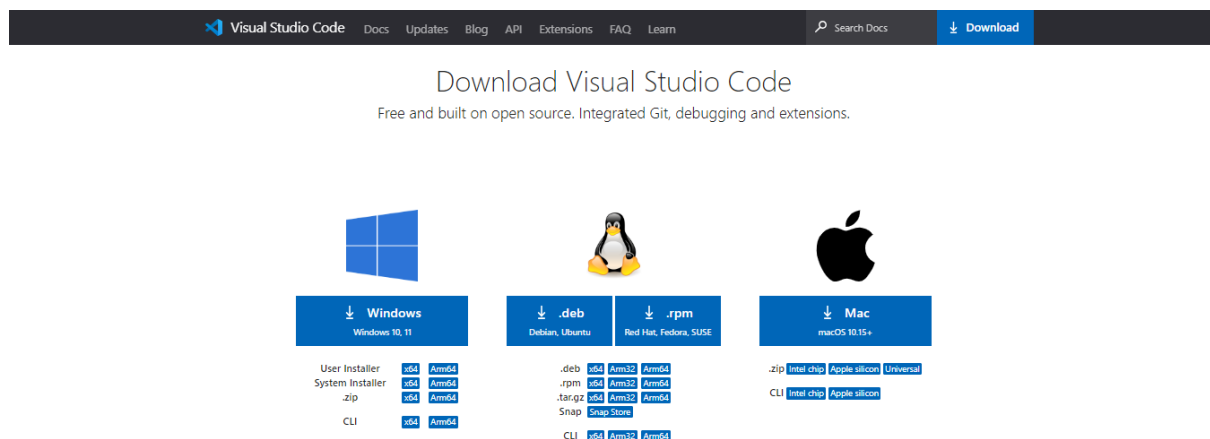
Manuseio de Resultados: Permite acessar e manipular os resultados das consultas.

Instruções de instalação dos softwares

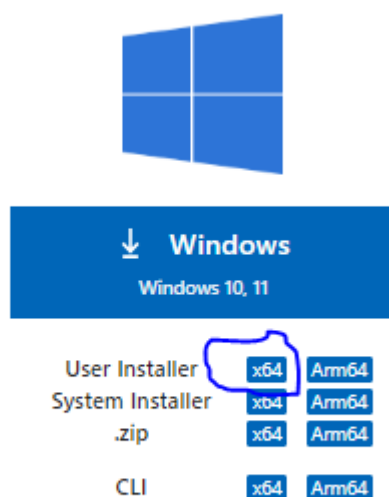
Instalação do Visual Studio Code

- [Clique aqui](https://code.visualstudio.com/Download) para ir a página de **Download** (ou copie em seu navegador: <https://code.visualstudio.com/Download>).

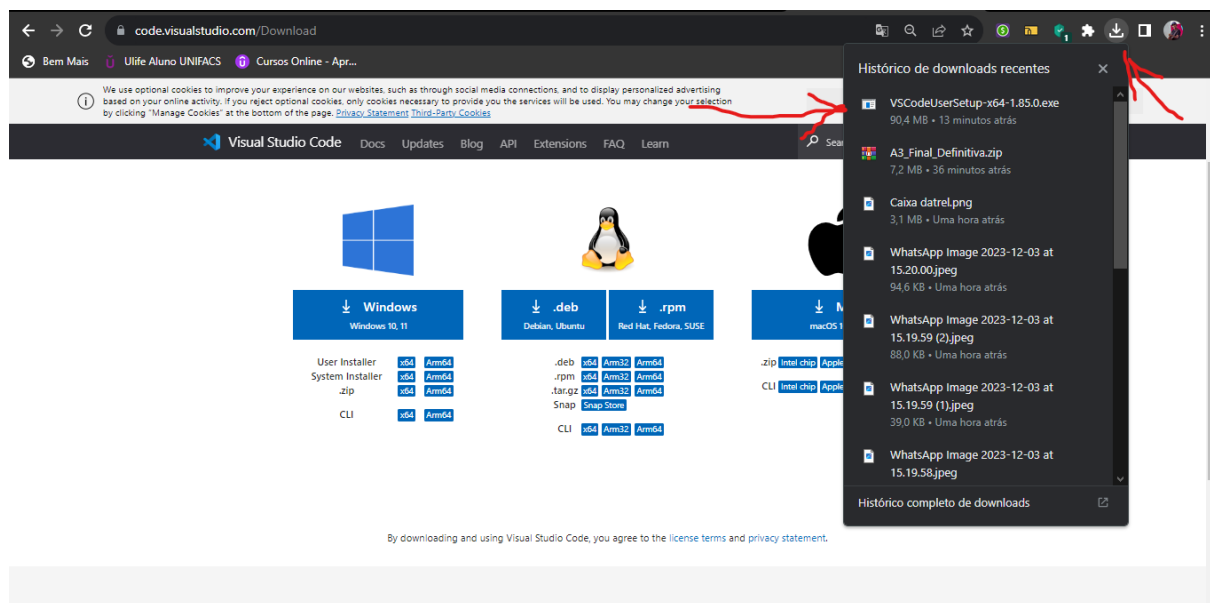
1- Após clicar no link acima, você será direcionado para esta tela, neste momento você irá selecionar o sistema operacional e qual modelo baixar.



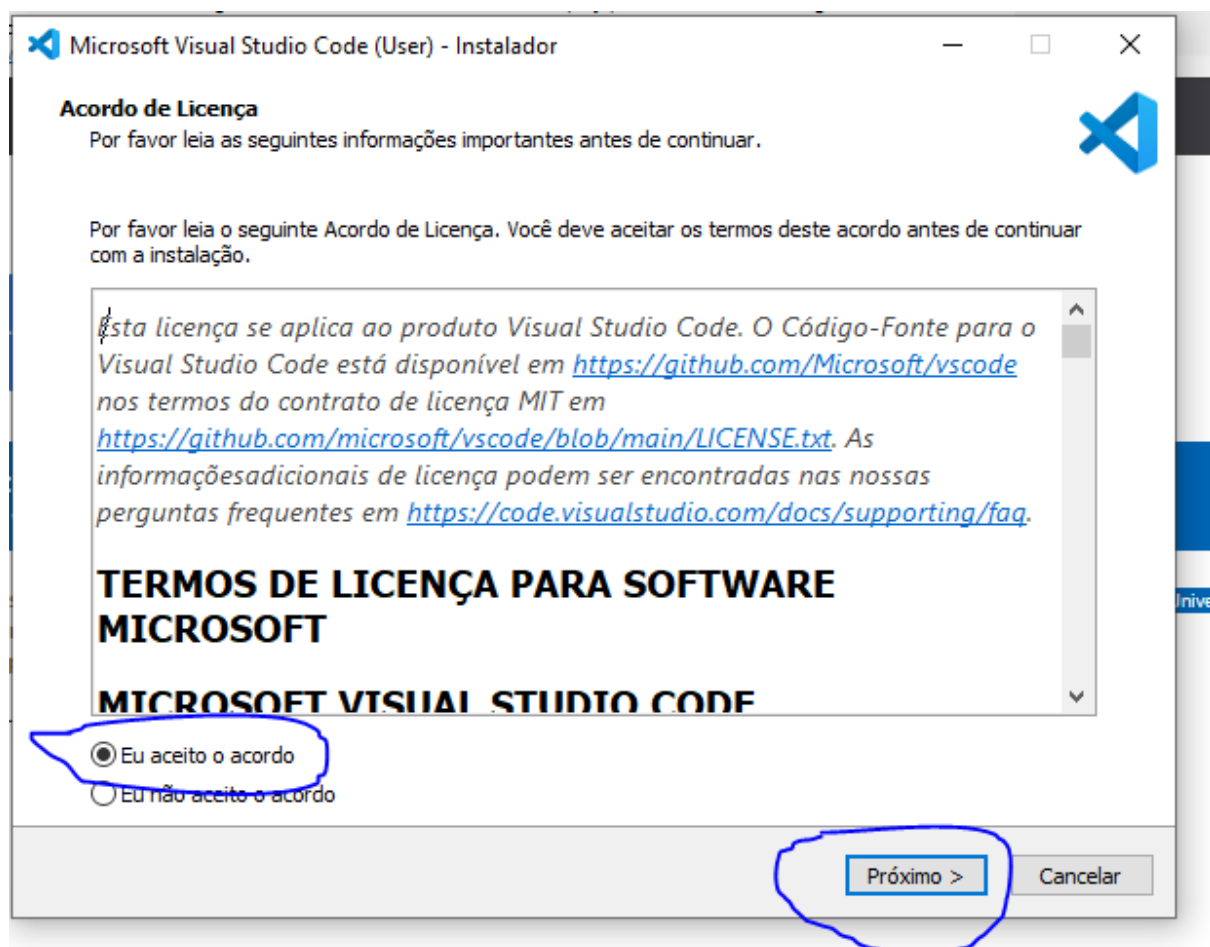
2- Se for **Windows**, clicar na opção abaixo e o download será feito automaticamente.



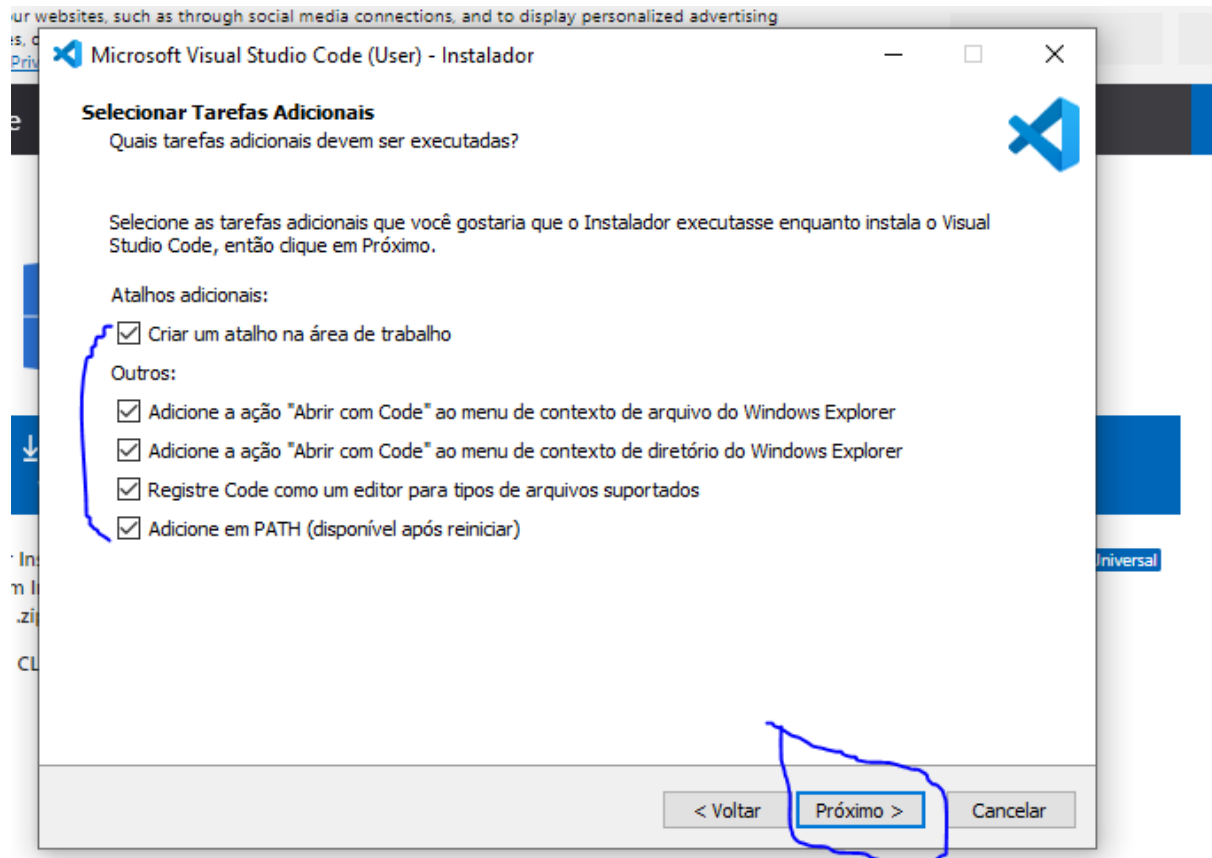
3-Depois você poderá clicar na aba de **downloads** do seu navegador e irá observar que o instalador do software estará lá, pode clicar nele



4-Depois clicar no instalador, ele irá abrir a seguinte tela, e você irá clicar em “Eu aceito o acordo”, depois poderá clicar no botão “Próximo”.

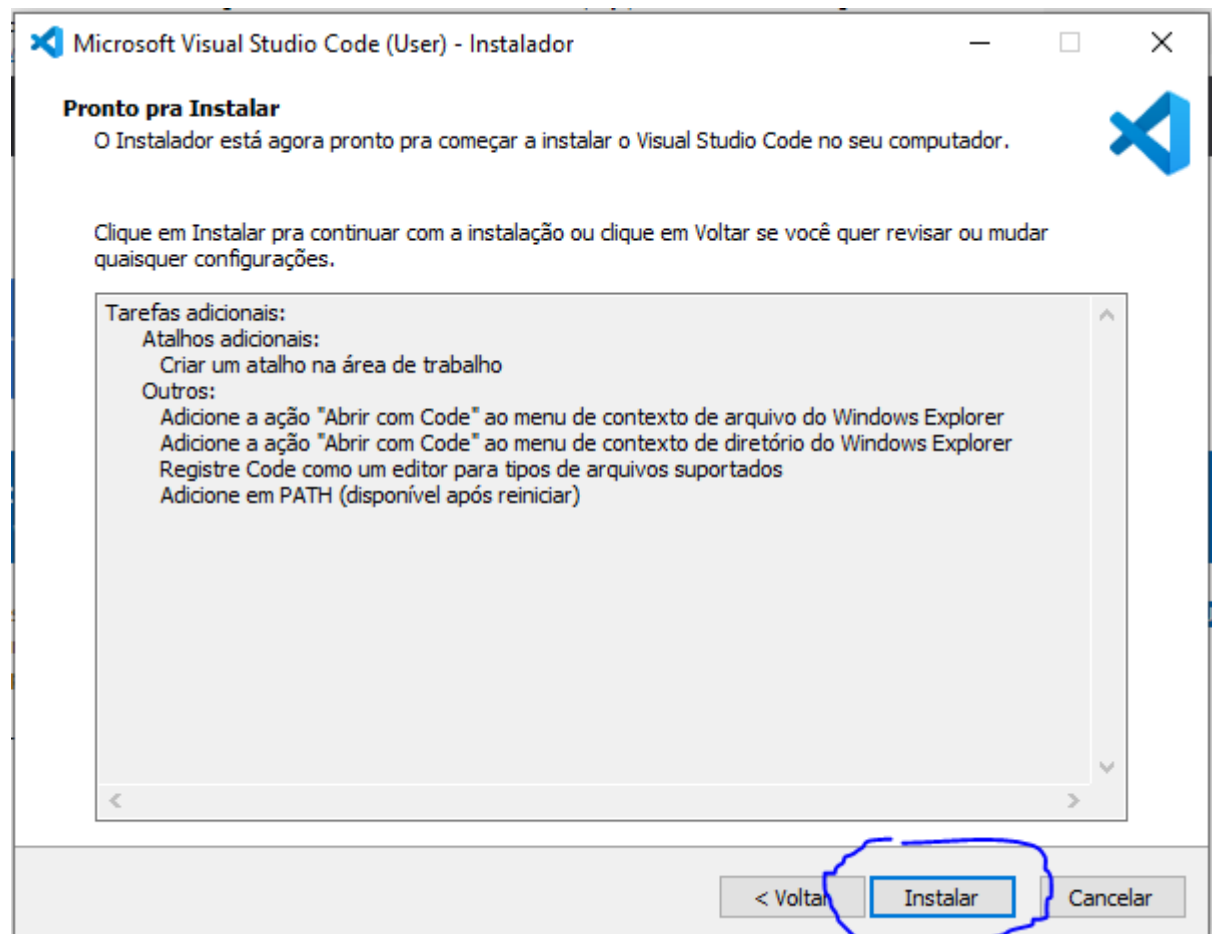


5-Marca todas as opções, e clica em “Próximo”.



By downloading and using Visual Studio Code, you agree to the [license terms](#) and [privacy statement](#).

6-Clicar em “Instalar” e aguardar a instalação.



Instalação do Node.JS

1- Baixar o Node.js:

- Acesse o site oficial do Node.js: <https://nodejs.org/>.
- Na página inicial, existem duas versões: "LTS" e "Current" (versão menos estável). Para a maioria dos usuários, é recomendado baixar a versão LTS.
- Clique no botão de download correspondente à versão LTS.

Node.js® is an open-source, cross-platform JavaScript runtime environment.

Download Node.js®

20.10.0 LTS

Recommended For Most Users

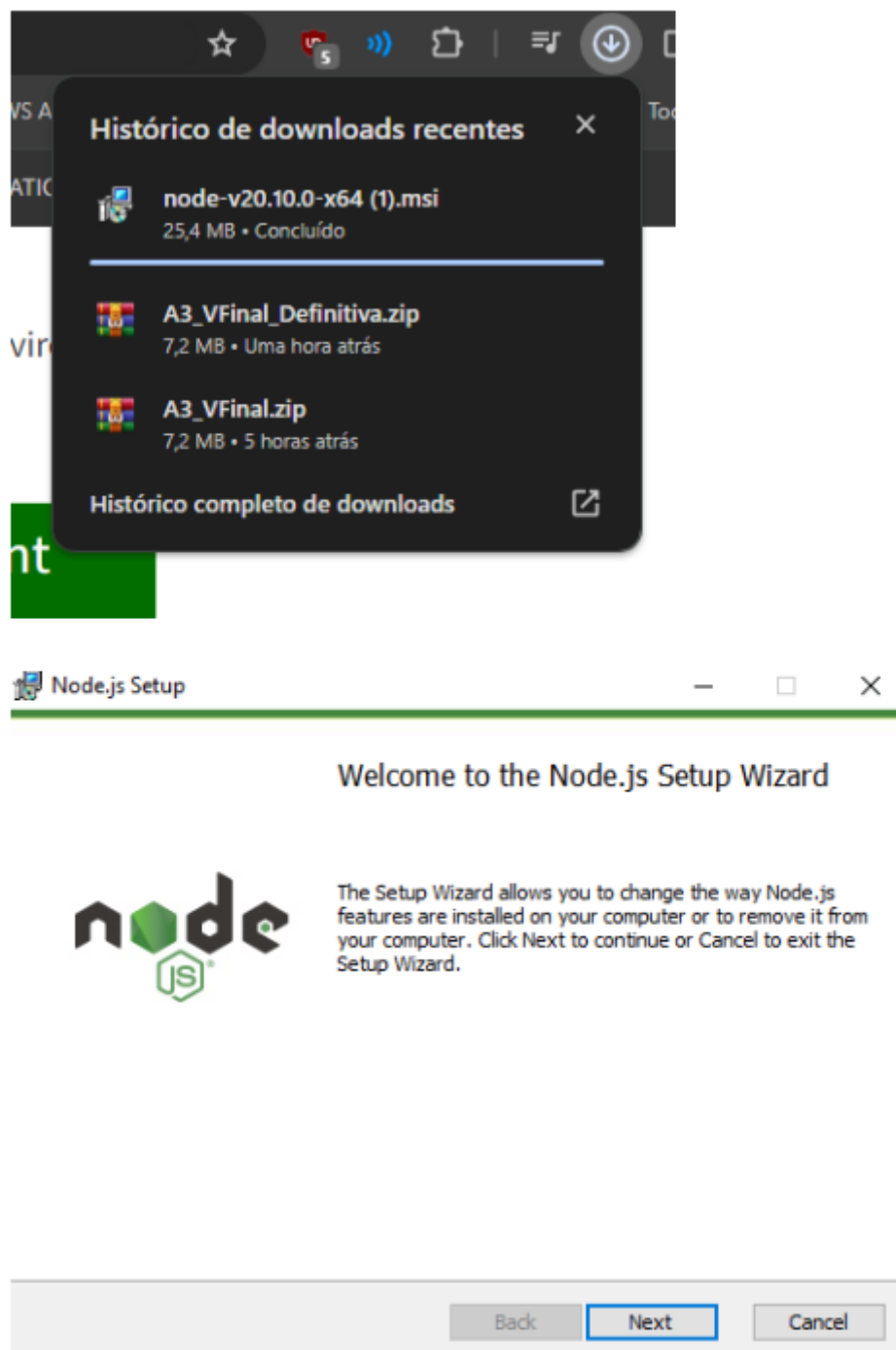
21.4.0 Current

Latest Features

[Other Downloads](#) | [Changelog](#) | [API Docs](#) [Other Downloads](#) | [Changelog](#) | [API Docs](#)

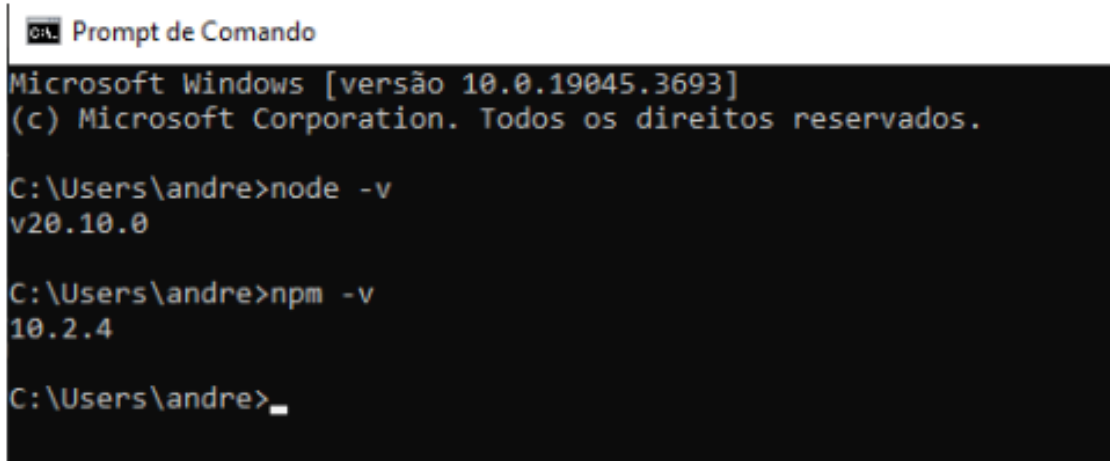
For information about supported releases, see the [release schedule](#).

2- Executar o instalador: Após o download, execute o instalador clicando nele na aba de download.



3- Verificar a instalação:

- Abra o prompt de comando (cmd).
- Execute os seguintes comandos para verificar se o Node.js e o npm (gerenciador de pacotes do Node.js) foram instalados corretamente: “node -v” e “npm -v”



```
Cit. Prompt de Comando
Microsoft Windows [versão 10.0.19045.3693]
(c) Microsoft Corporation. Todos os direitos reservados.

C:\Users\andre>node -v
v20.10.0

C:\Users\andre>npm -v
10.2.4

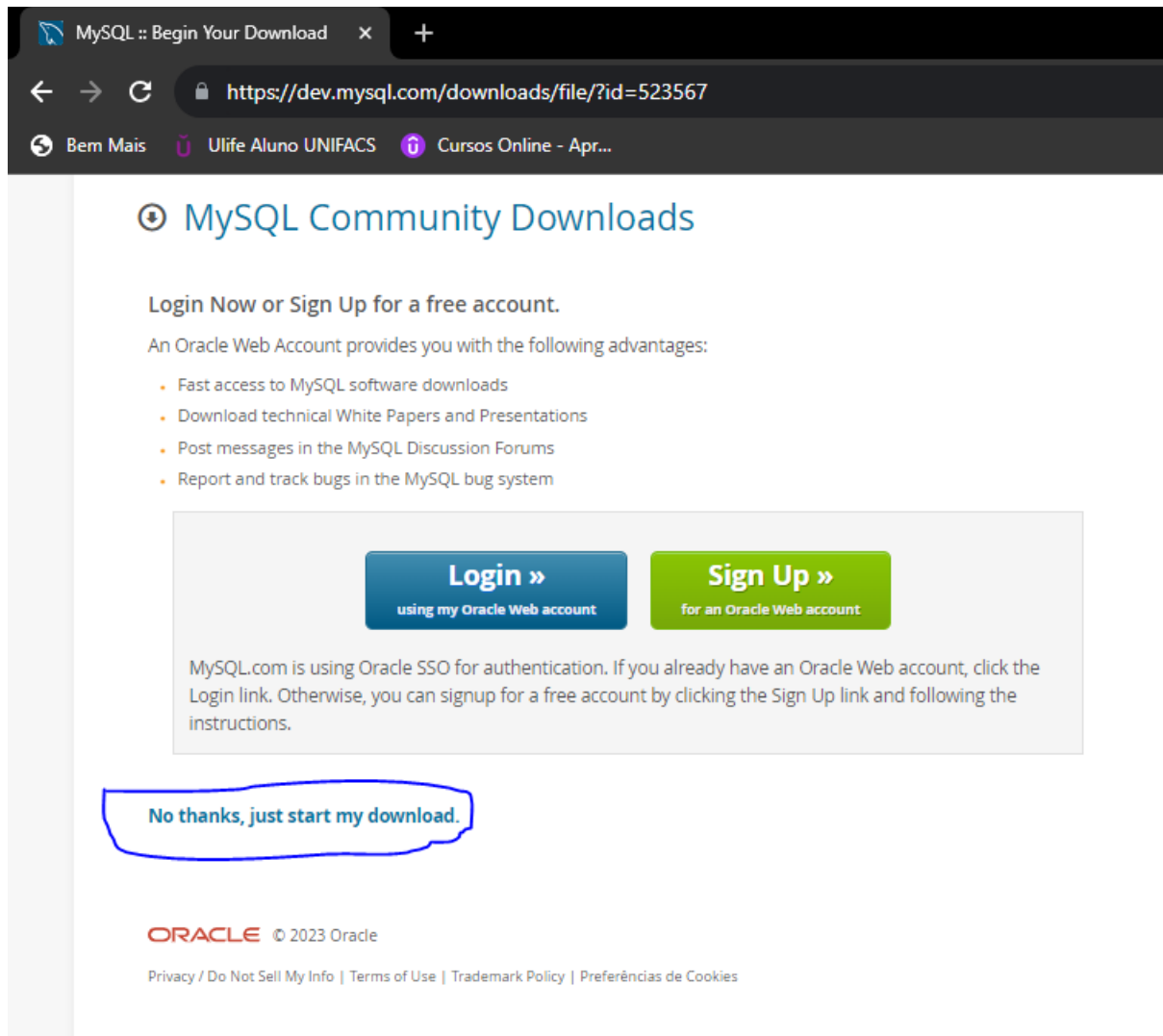
C:\Users\andre>
```

Após isso, a aplicação utilizando o node.js pode ser realizada

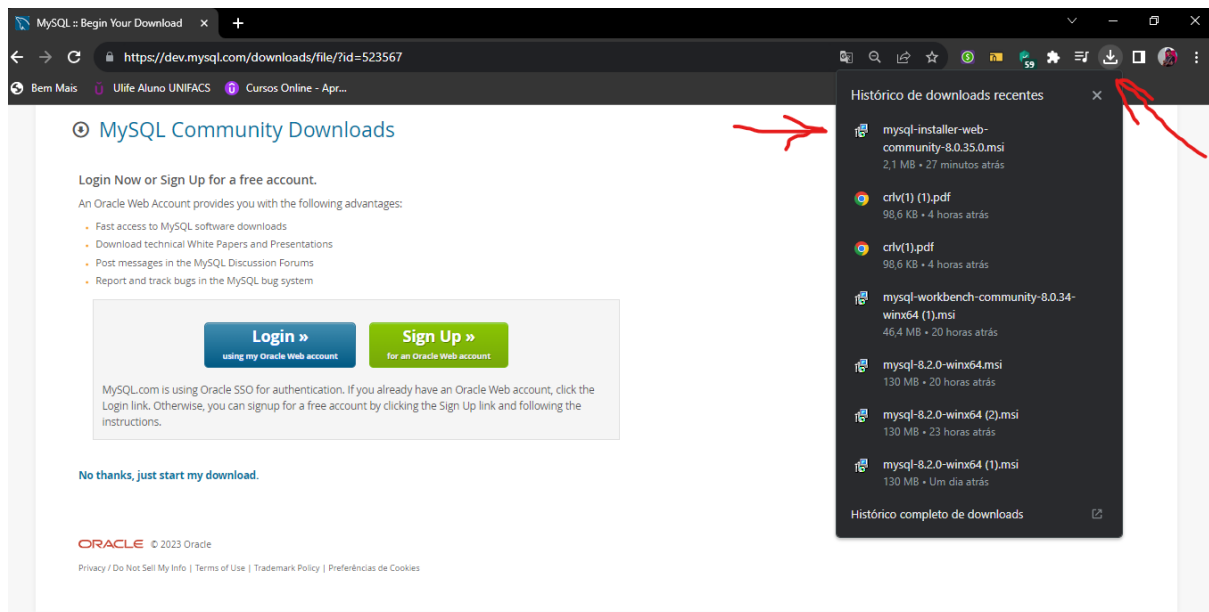
Instalação do MySQL - no Windows

-[Clique aqui](https://dev.mysql.com/downloads/file/?id=523567), para ir a página de **Download** (ou copie em seu navegador: <https://dev.mysql.com/downloads/file/?id=523567>)

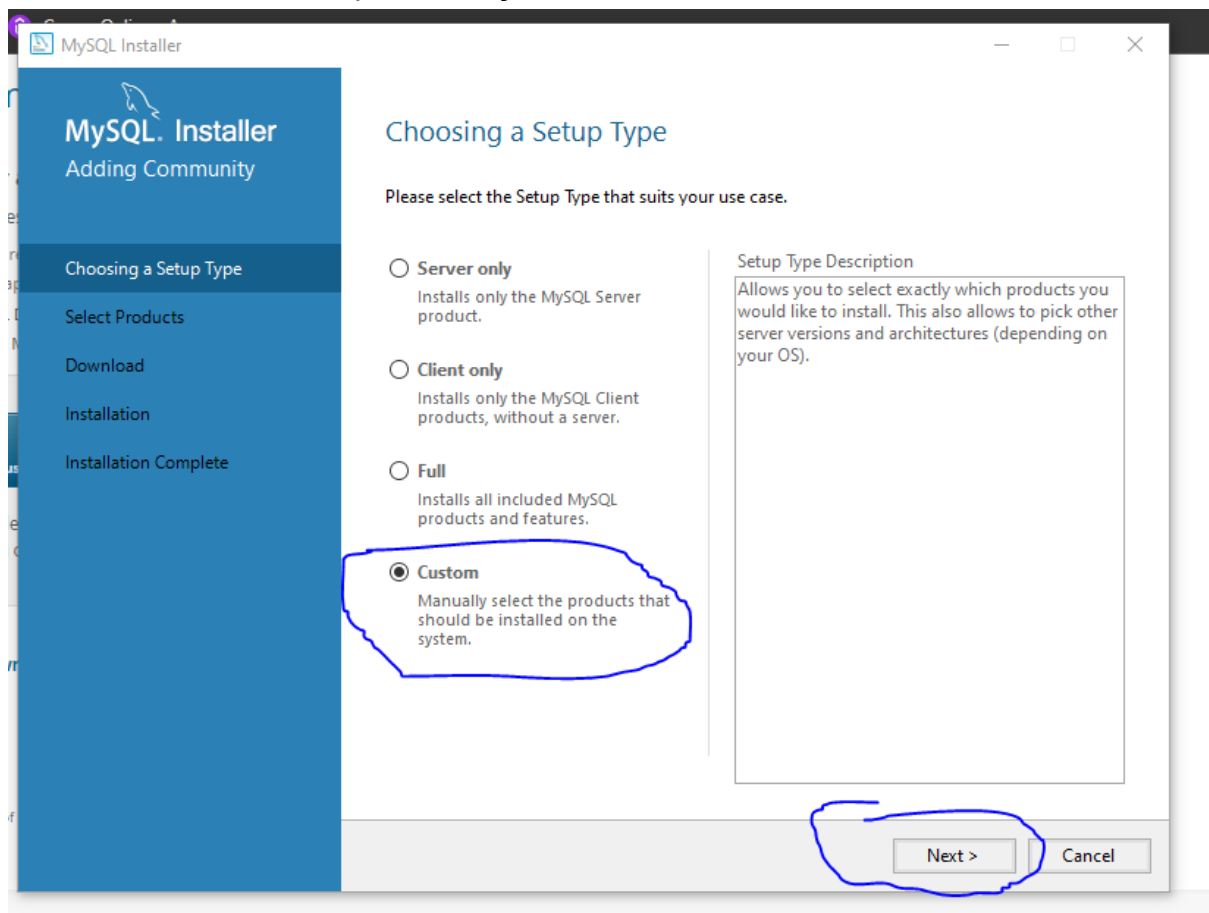
1. Após clicar no link acima, você será direcionado para esta tela, neste momento você irá clicar na área selecionada, e o download será feito automaticamente,



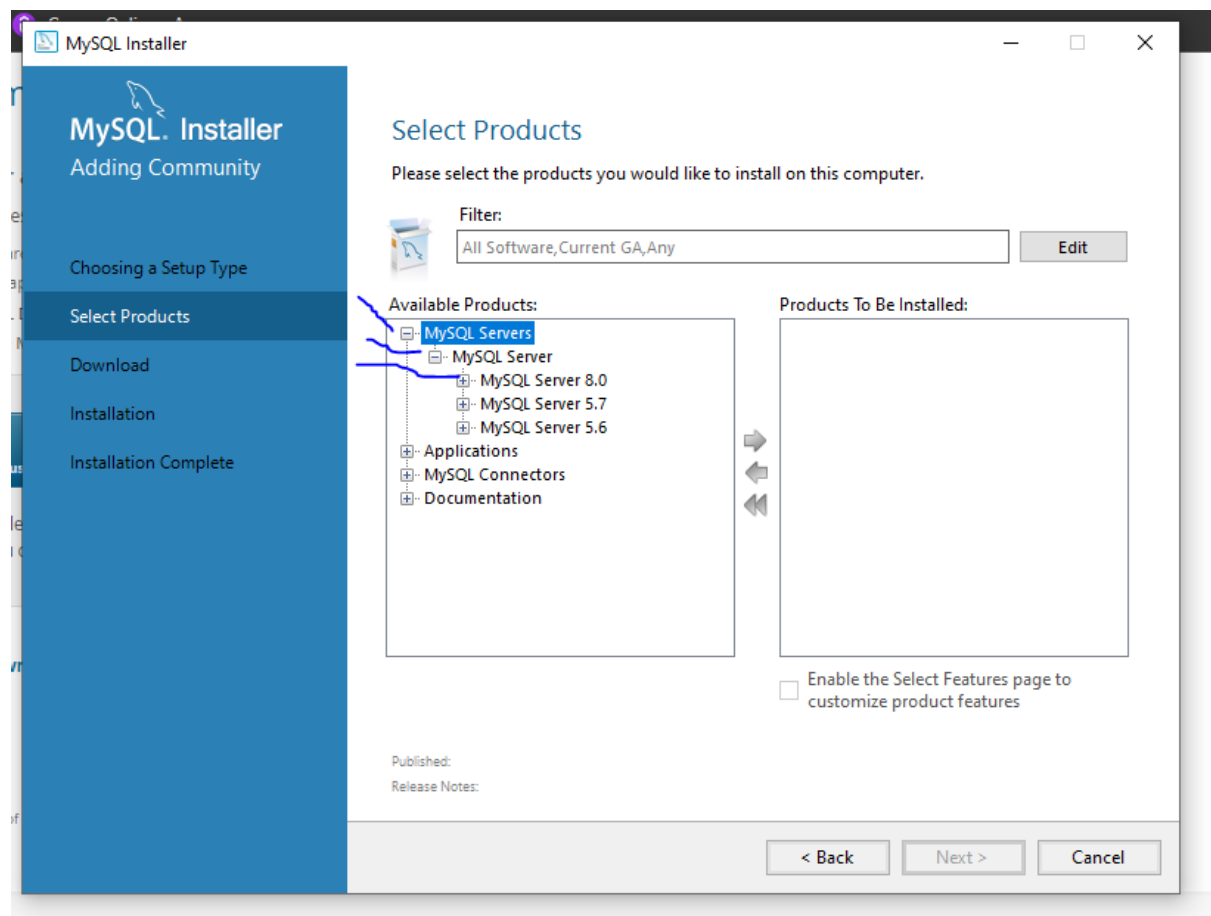
- Depois você poderá clicar na aba de **downloads** do seu navegador e irá observar que o instalador do software estará lá, pode clicar nele



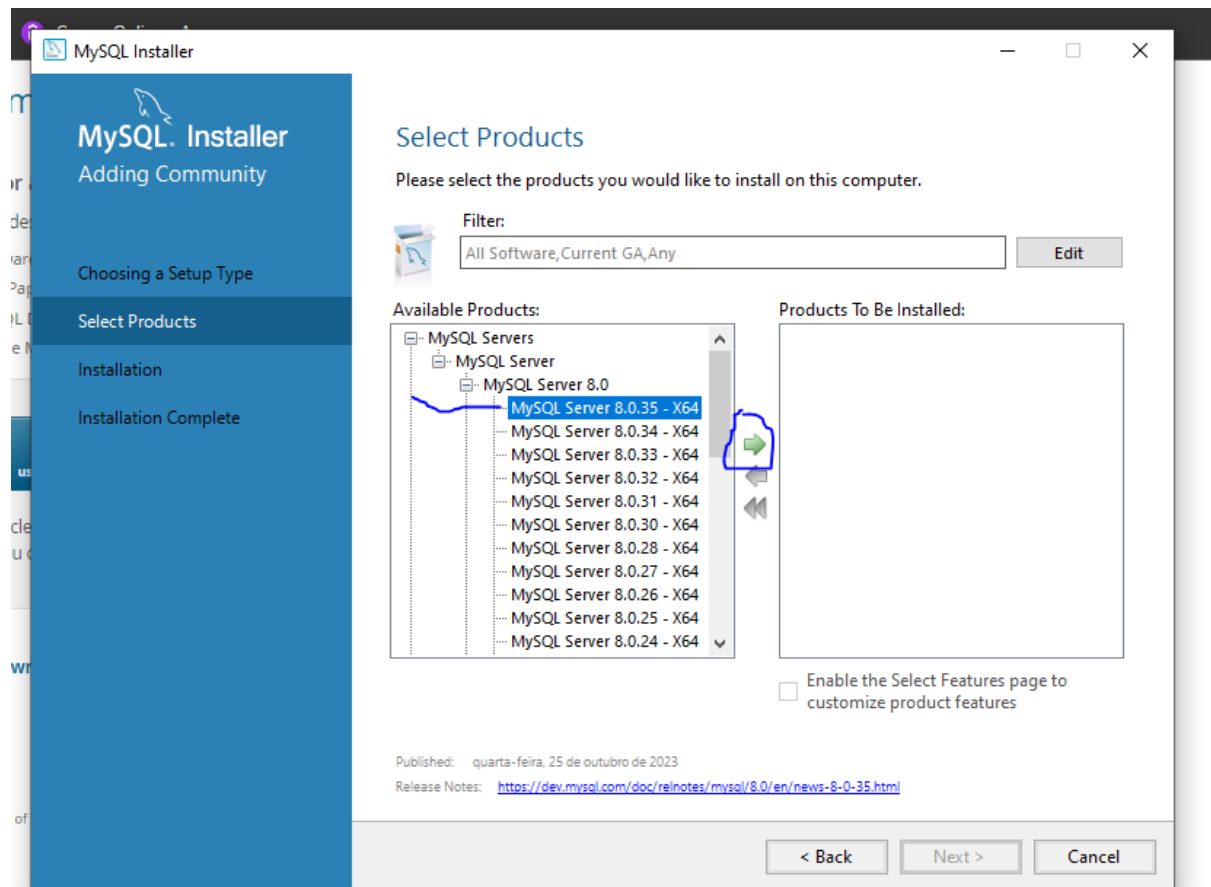
- Após clicar no instalador, o Windows apresentará uma tela com a seguinte mensagem “**deseja que este aplicativo faça alterações no seu dispositivo**” por 2 vezes, e você irá clicar em “**SIM**” para confirmar.
- Depois será aberto a seguinte tela, você marcará a opção “**custom**”, e depois clicará em “**Next**” para avançar.



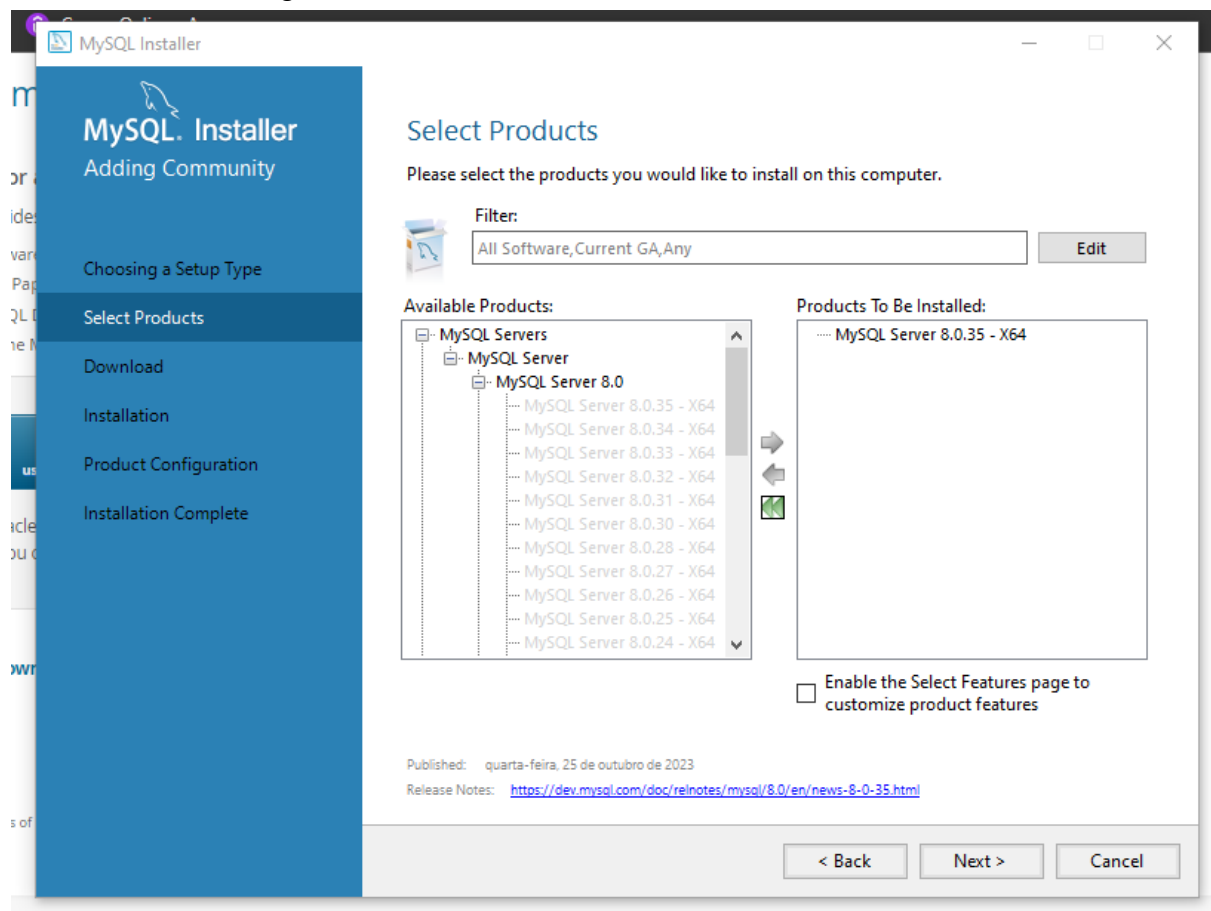
5. a próxima tela, é aberto a aba “**Select Products**”, Em “**MySQL Servers**” você irá clicar nas partes marcadas, até abrir a opção de “**MySQL Server 8.0**” .



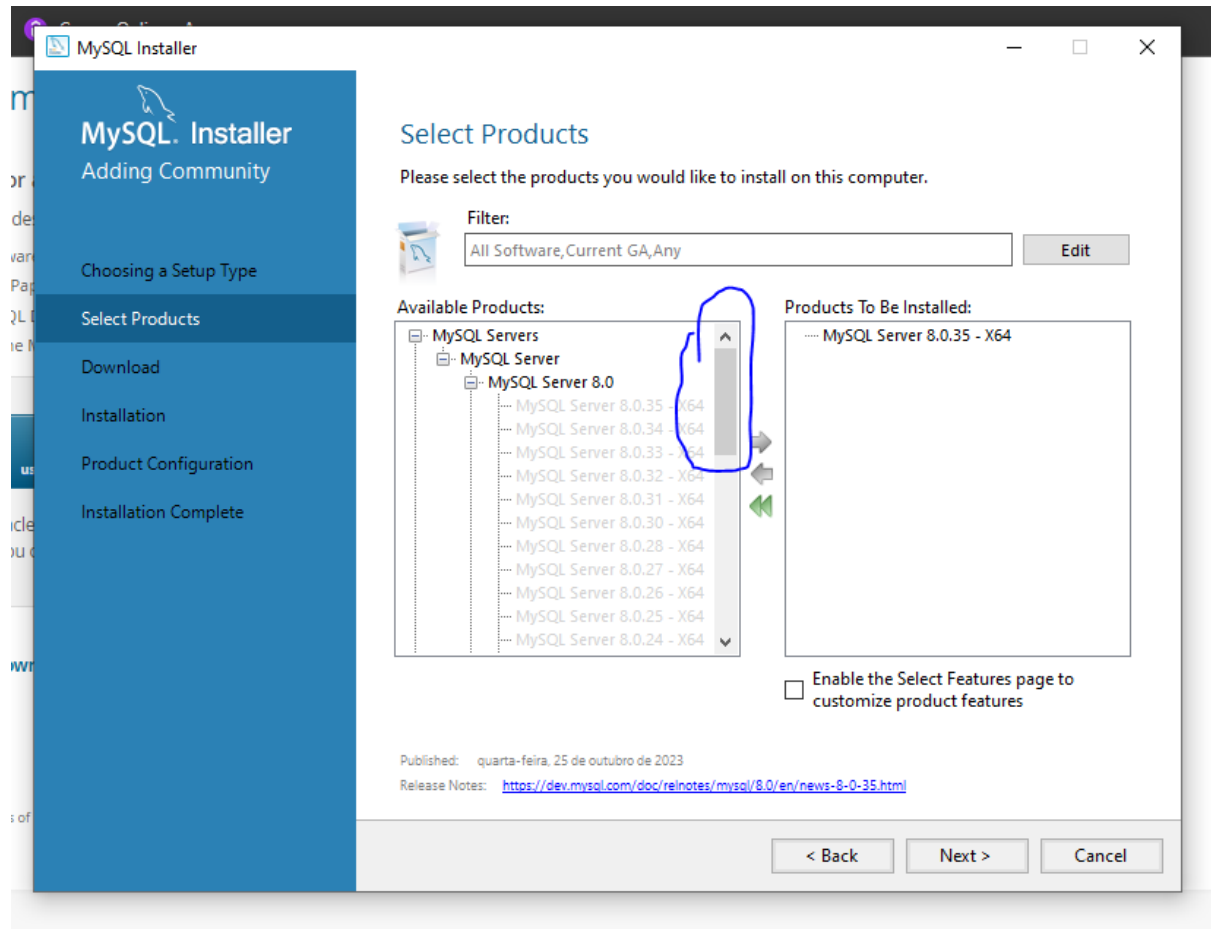
6. Você irá selecionar a primeira opção que é a versão mais atualizada, e vai clicar na seta **verde** para escolher o produto.



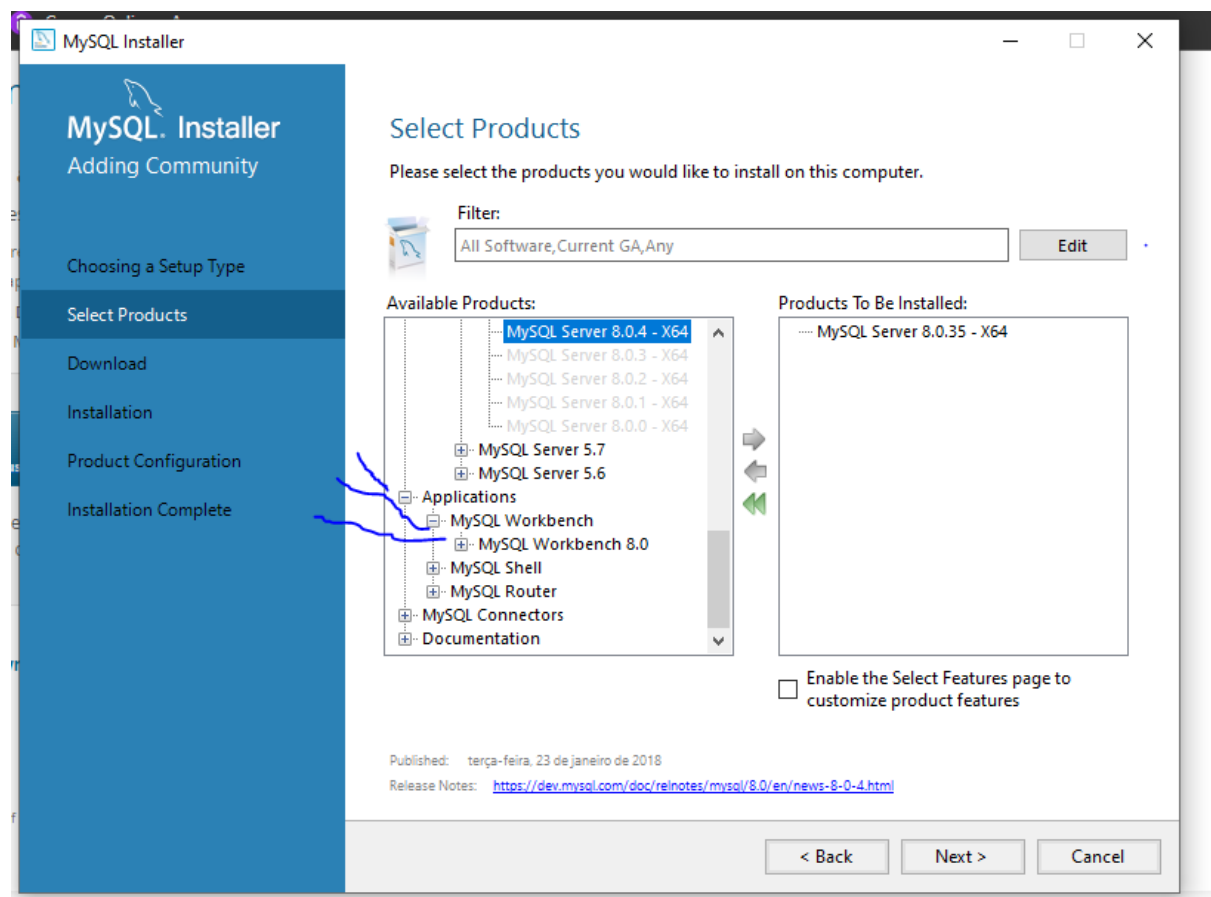
7. Ficará da seguinte forma



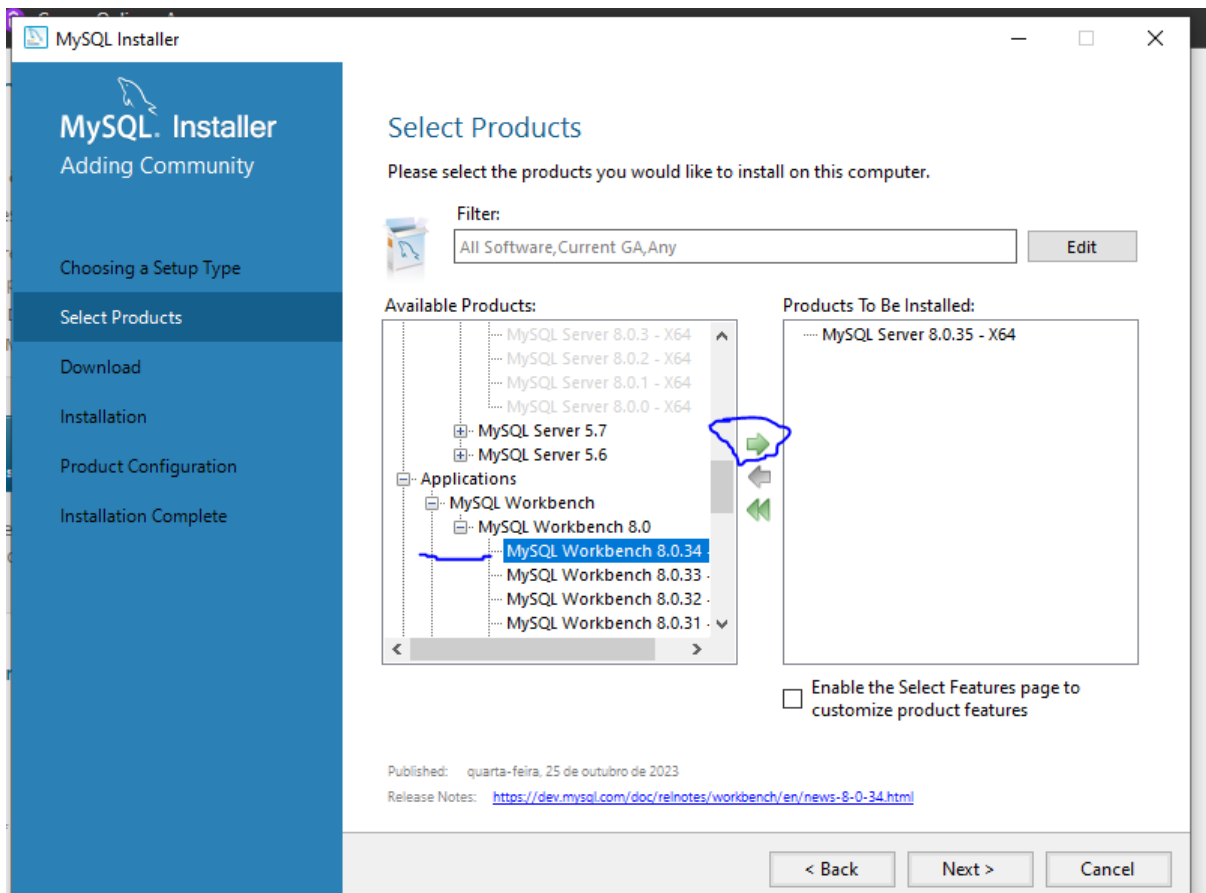
8. Em seguida você puxará a barra de rolagem toda para baixo



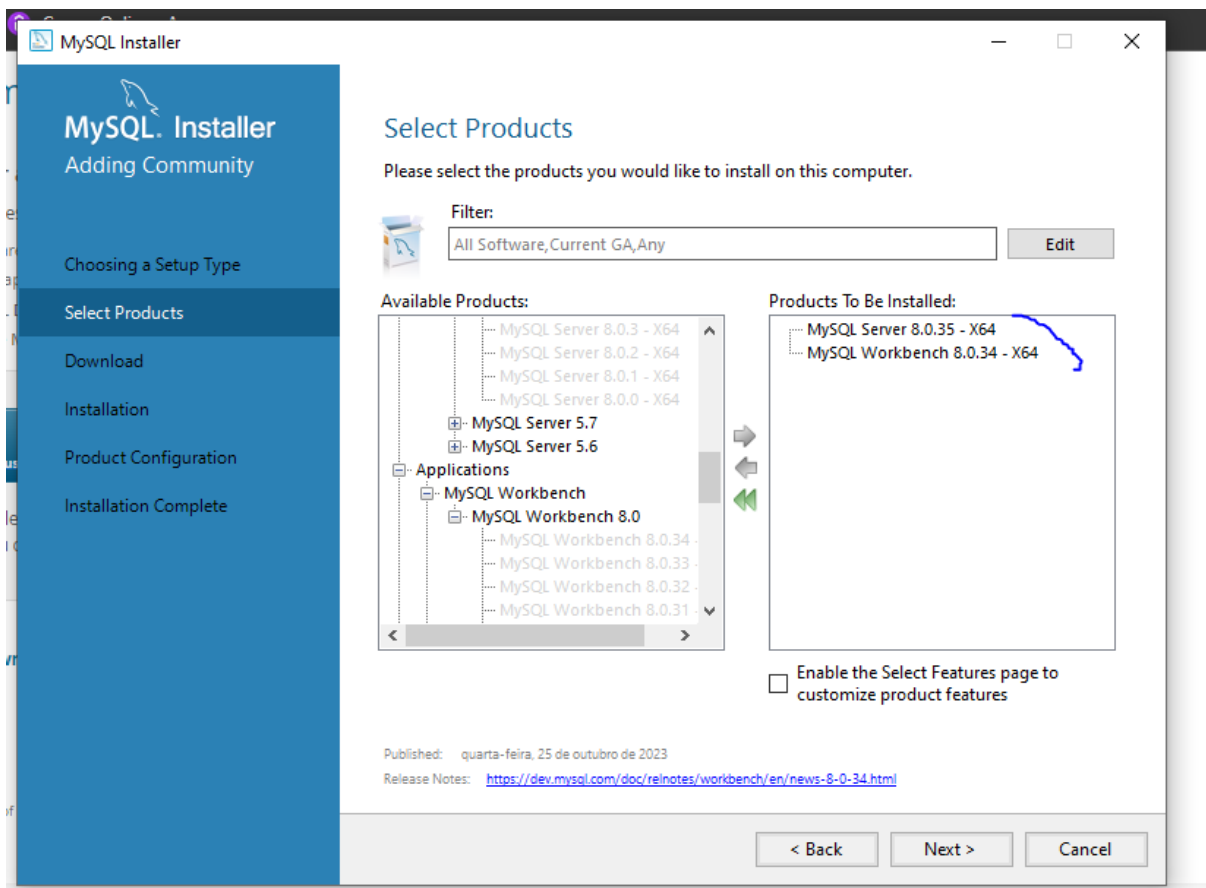
9. Em seguida você irá clicar em “Applications”, “MySQL Workbench”, “MySQL Workbench 8.0”



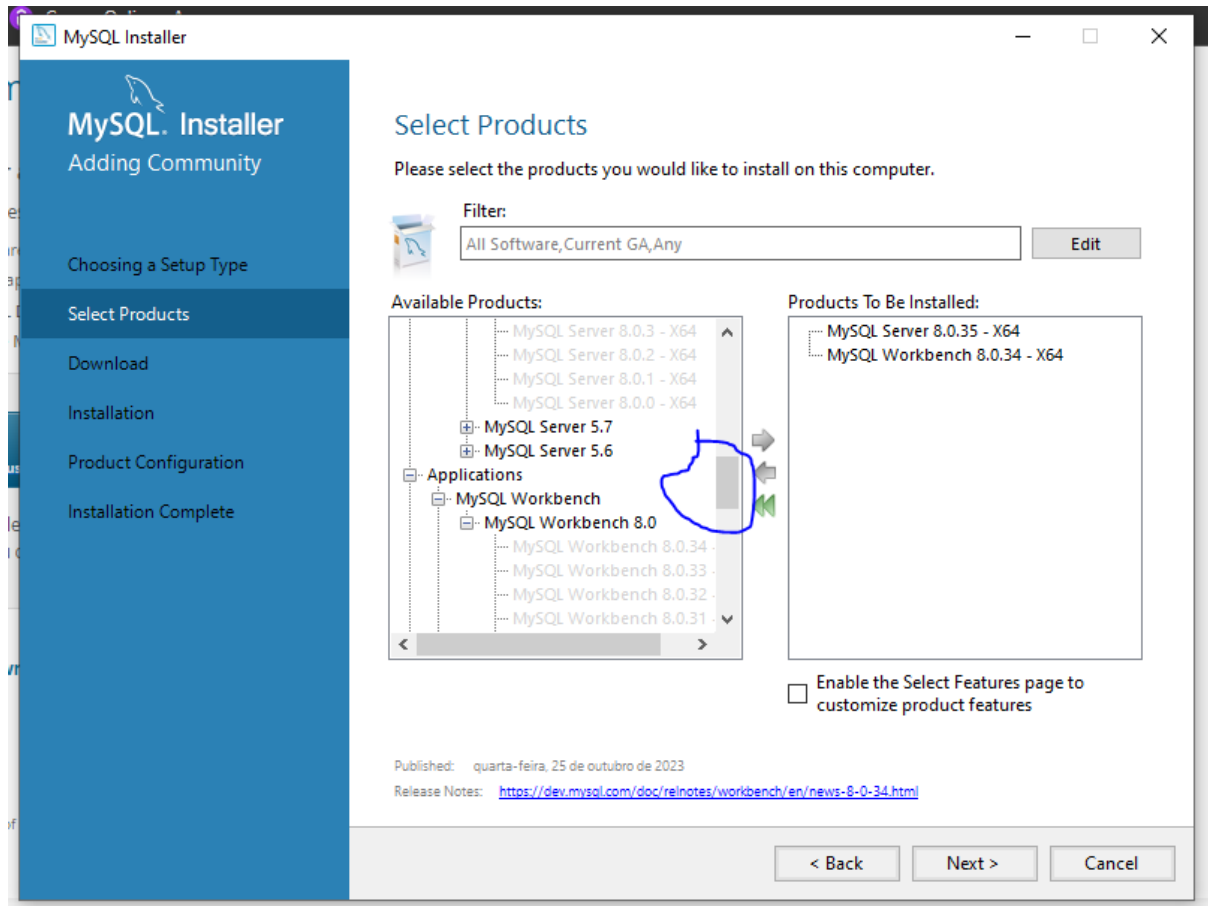
10. Você irá selecionar a primeira opção que é a versão mais atualizada, e vai clicar na seta **verde** para escolher o produto.



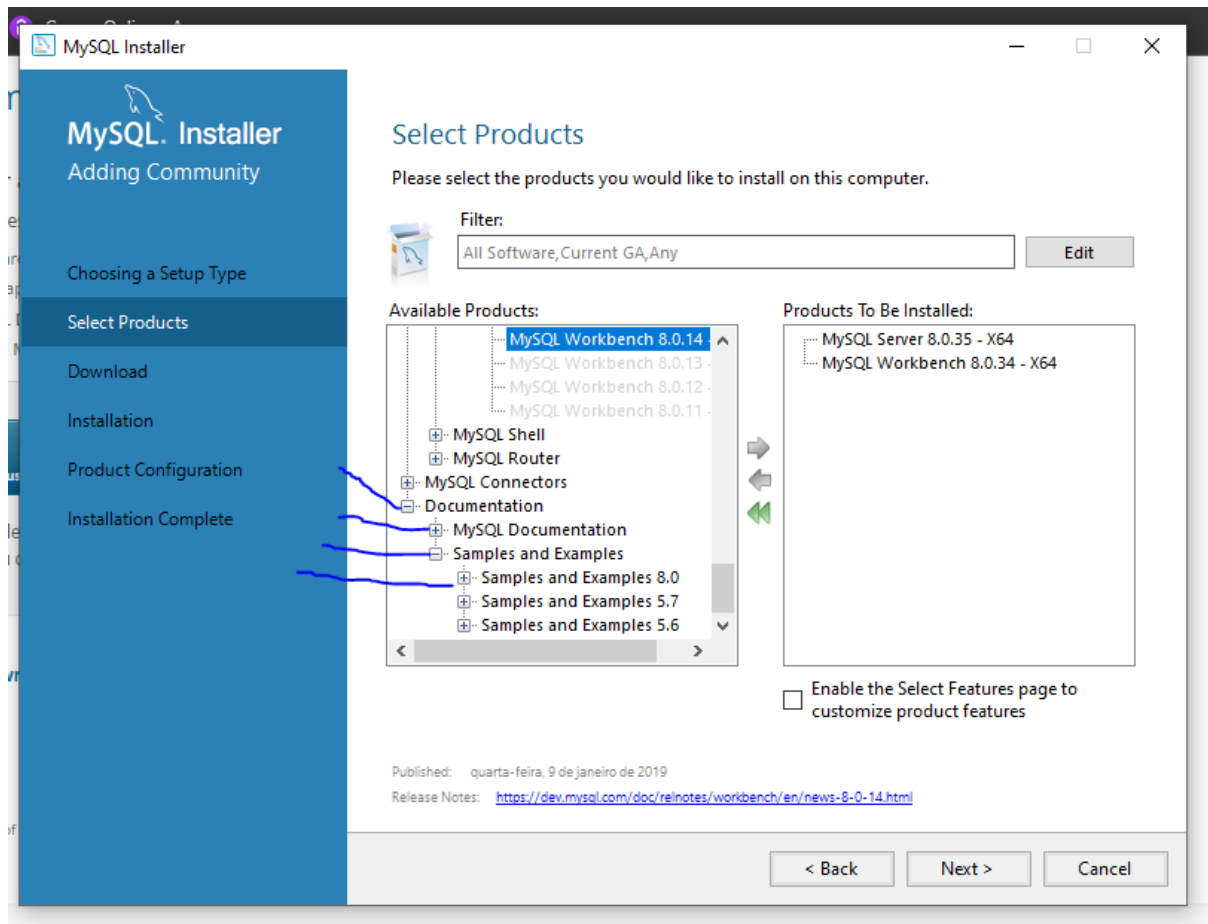
11. Ficará da seguinte forma



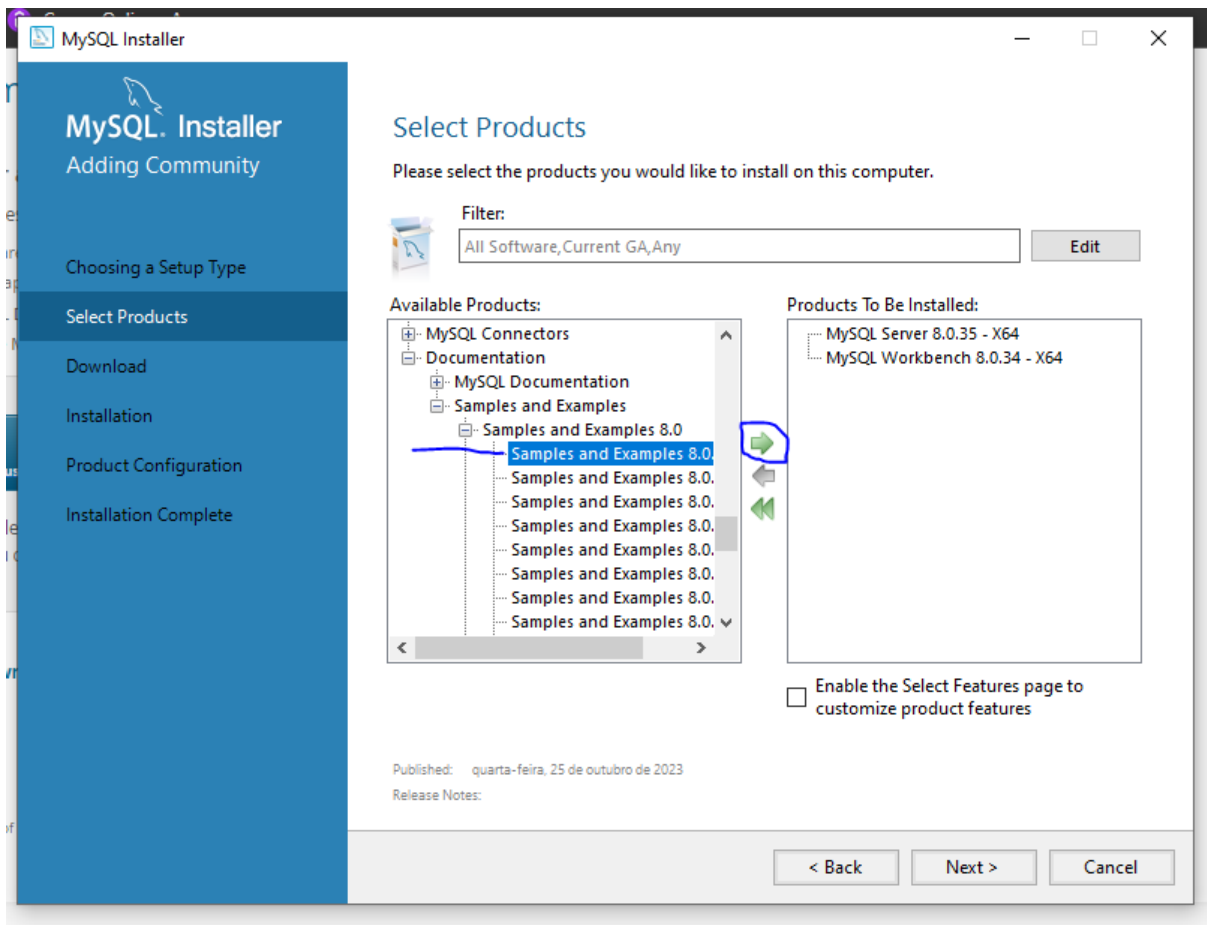
12. Em seguida você puxará a barra de rolagem toda para baixo.



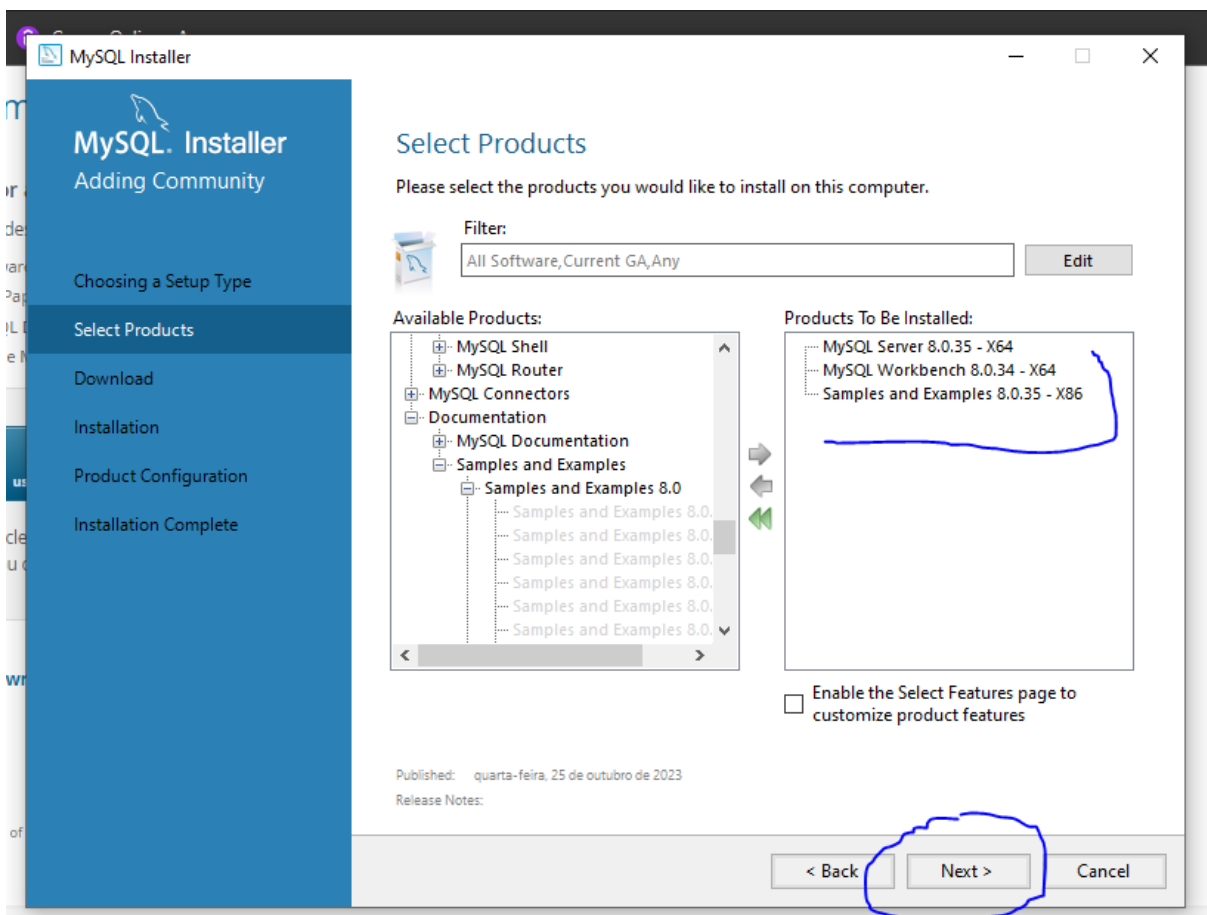
13. Em seguida você irá clicar em “Documentation”, “MySQL Documentation”, “Sample and Examples” e “Samples and Examples 8.0”



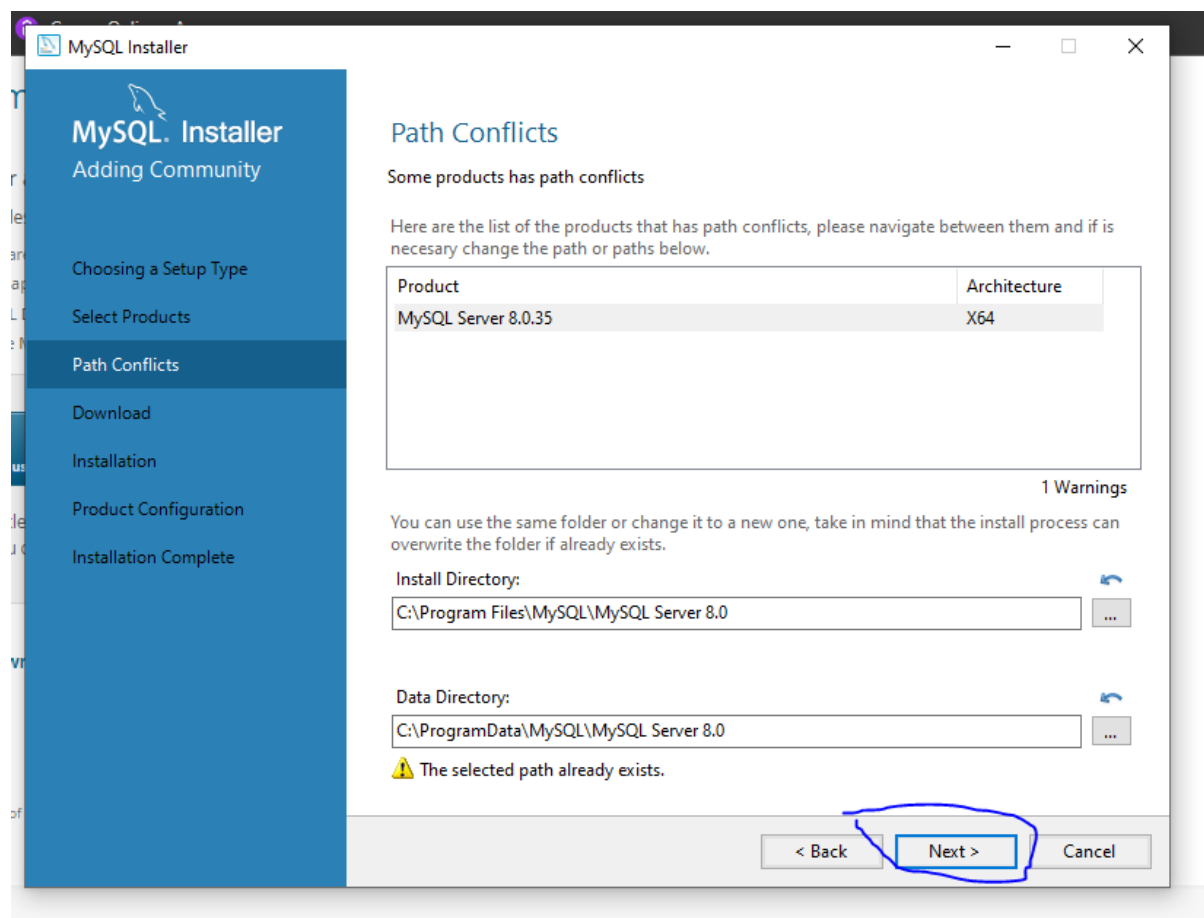
14. Você irá selecionar a primeira opção que é a versão mais atualizada, e vai clicar na seta **verde** para escolher o produto.



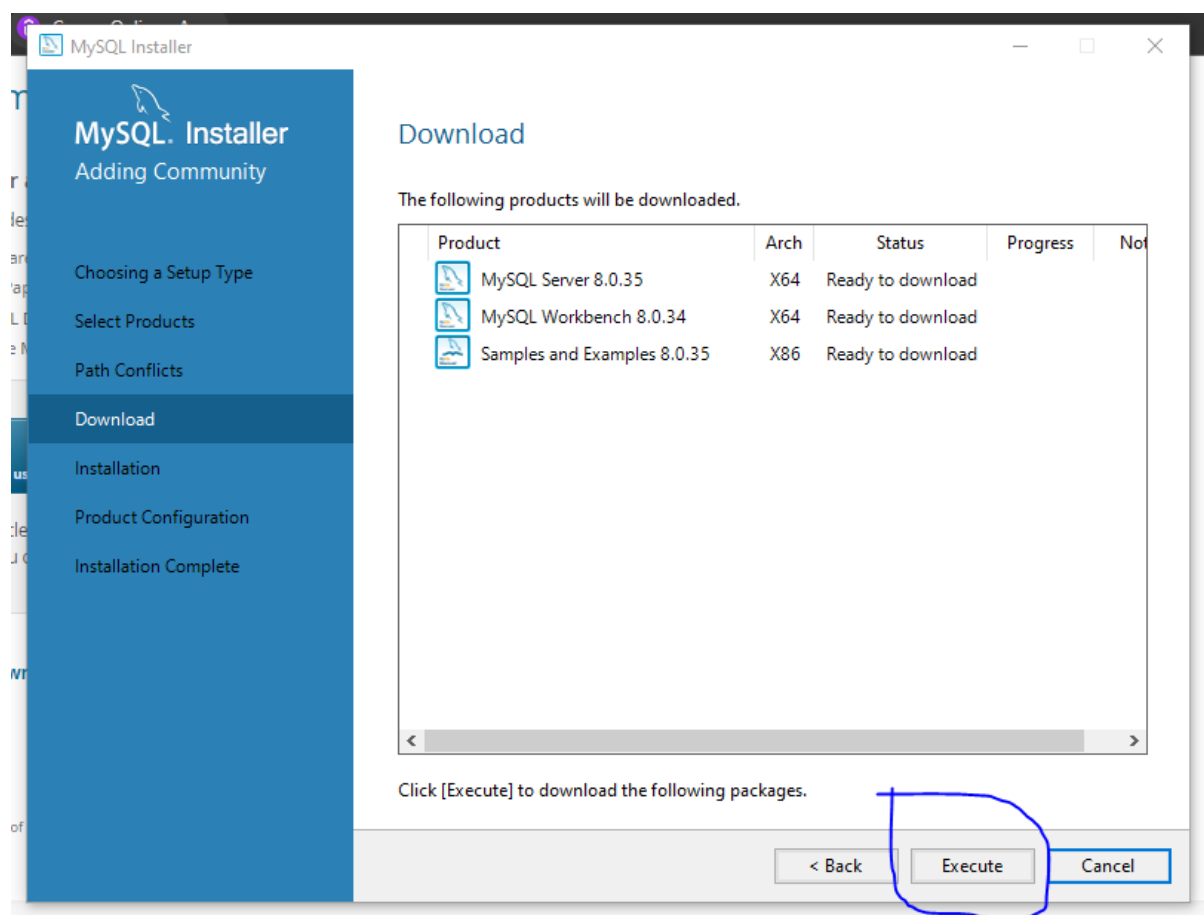
15. Ficará da seguinte forma, e você poderá clicar em “**Next**”



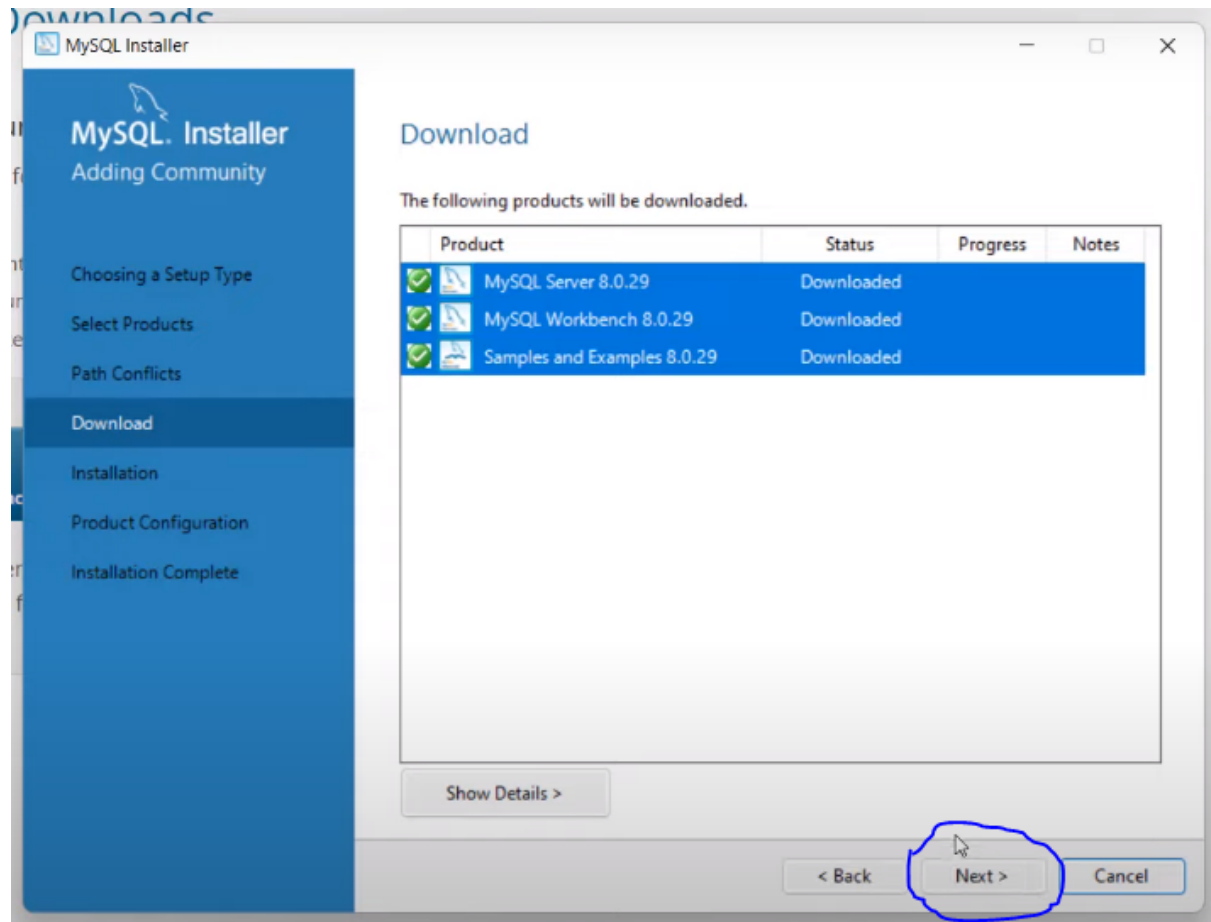
16. Aparecerá a seguinte tela, e você irá clicar em “**Next**” para avançar.



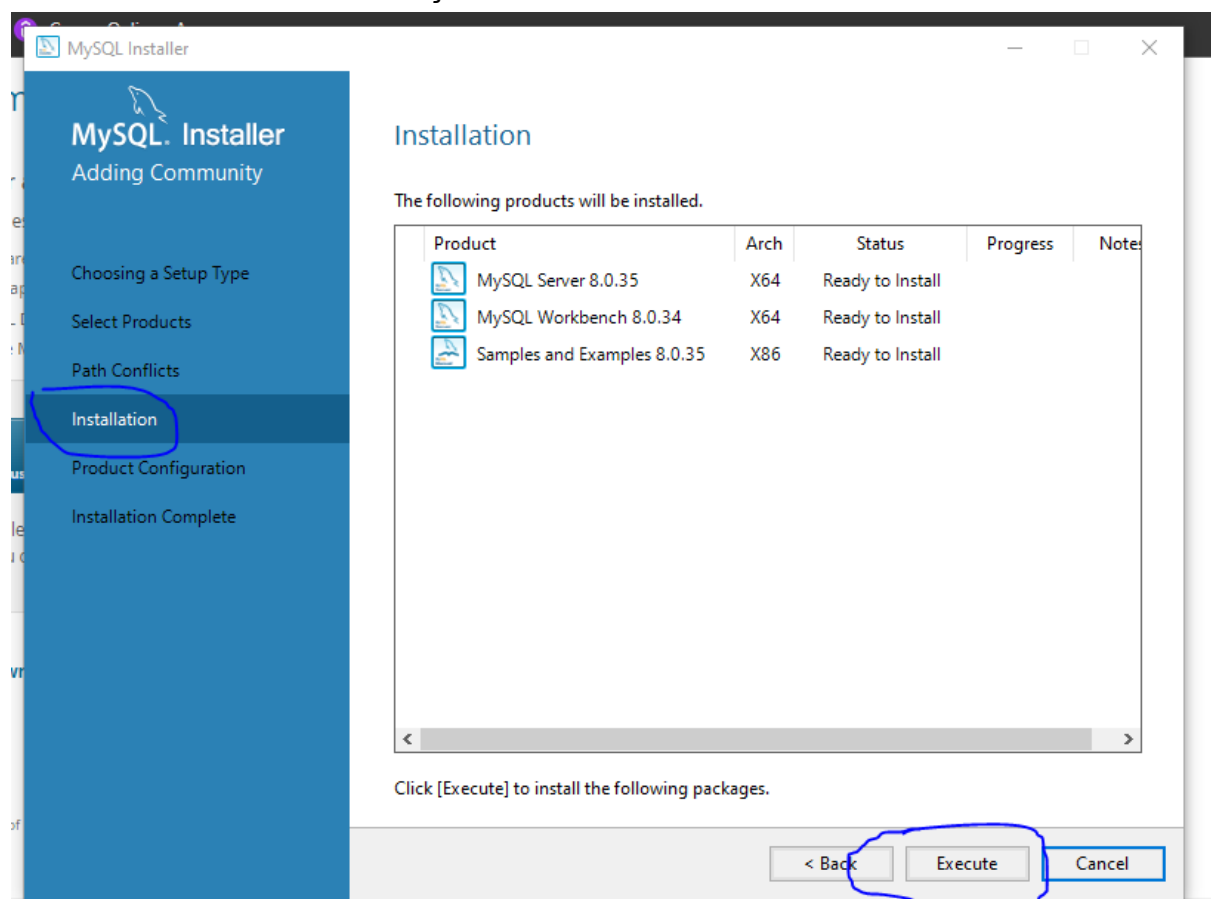
17. Nesta tela você irá clicar em “**EXECUTE**”, e o MySQL irá baixar os produtos escolhidos



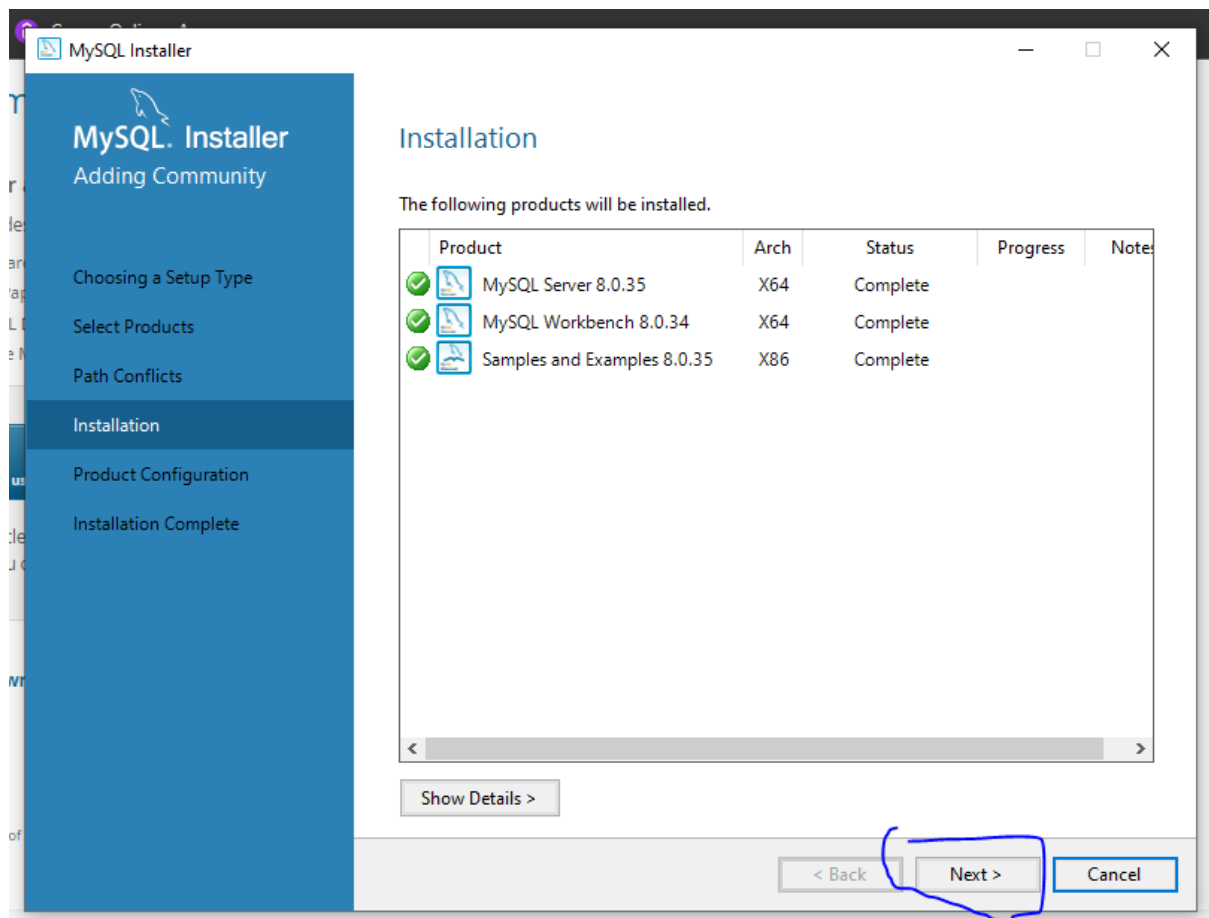
18. As opções ficaram verdes, depois clique em “**Next**”.



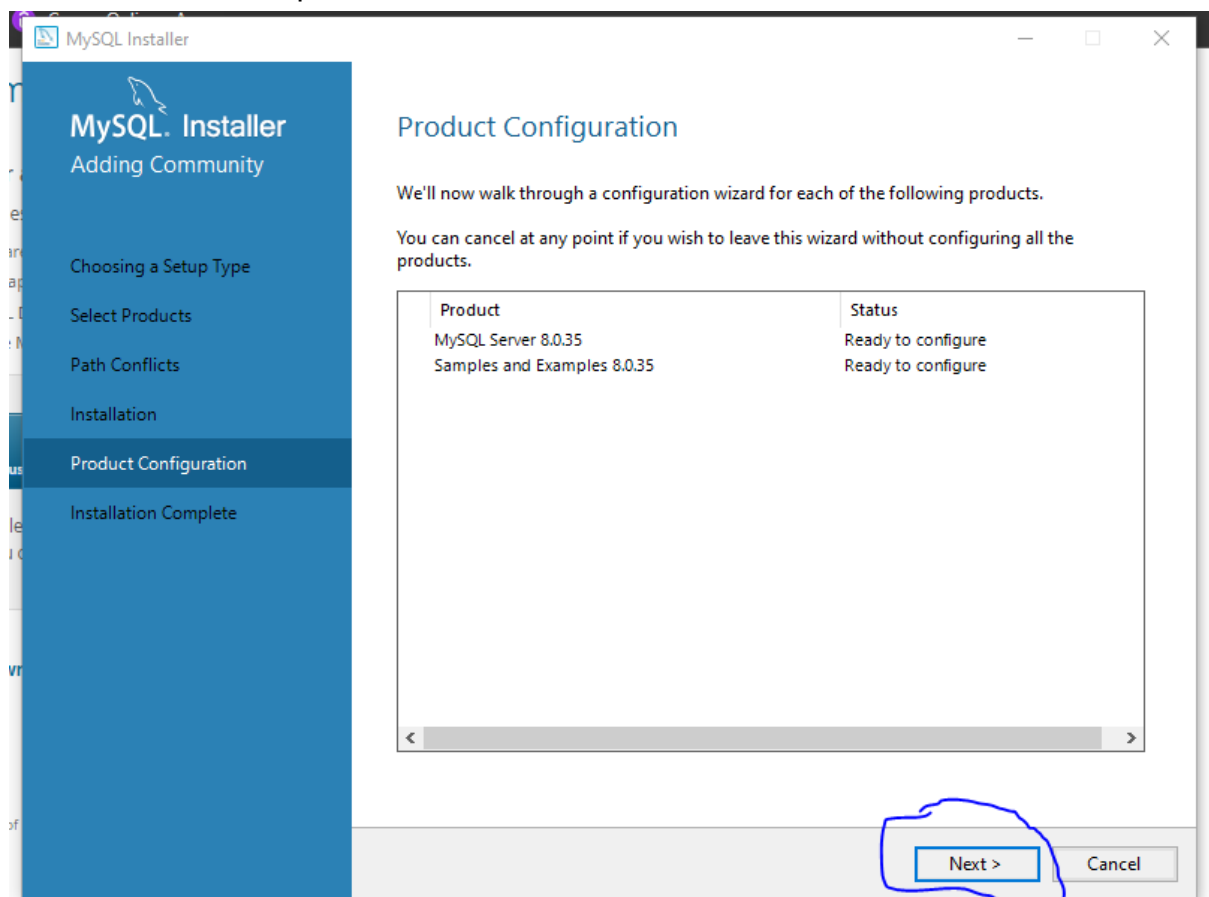
19. Esta é a aba de instalação e é só clicar em “**Execute**”



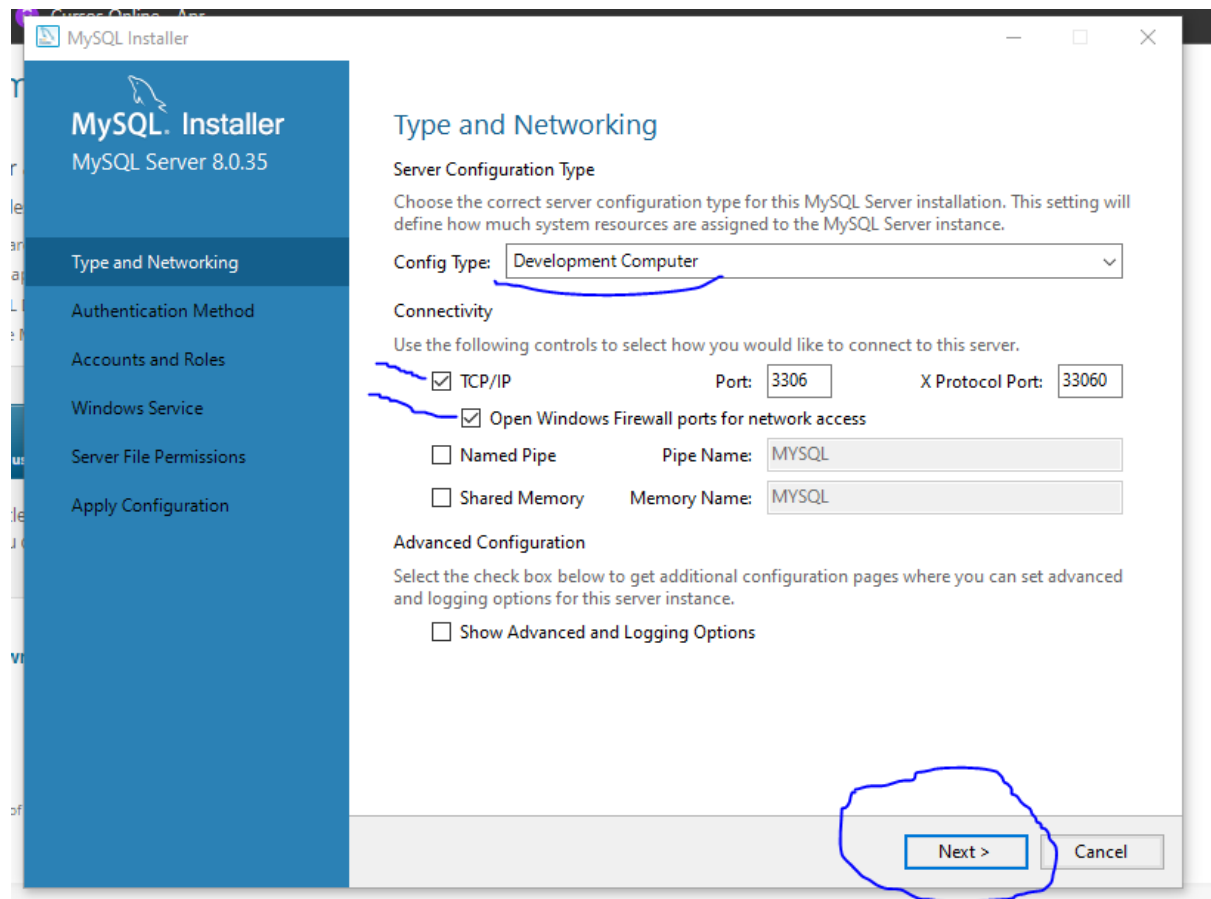
20. Depois de tudo instalado, pode clicar em “Next”



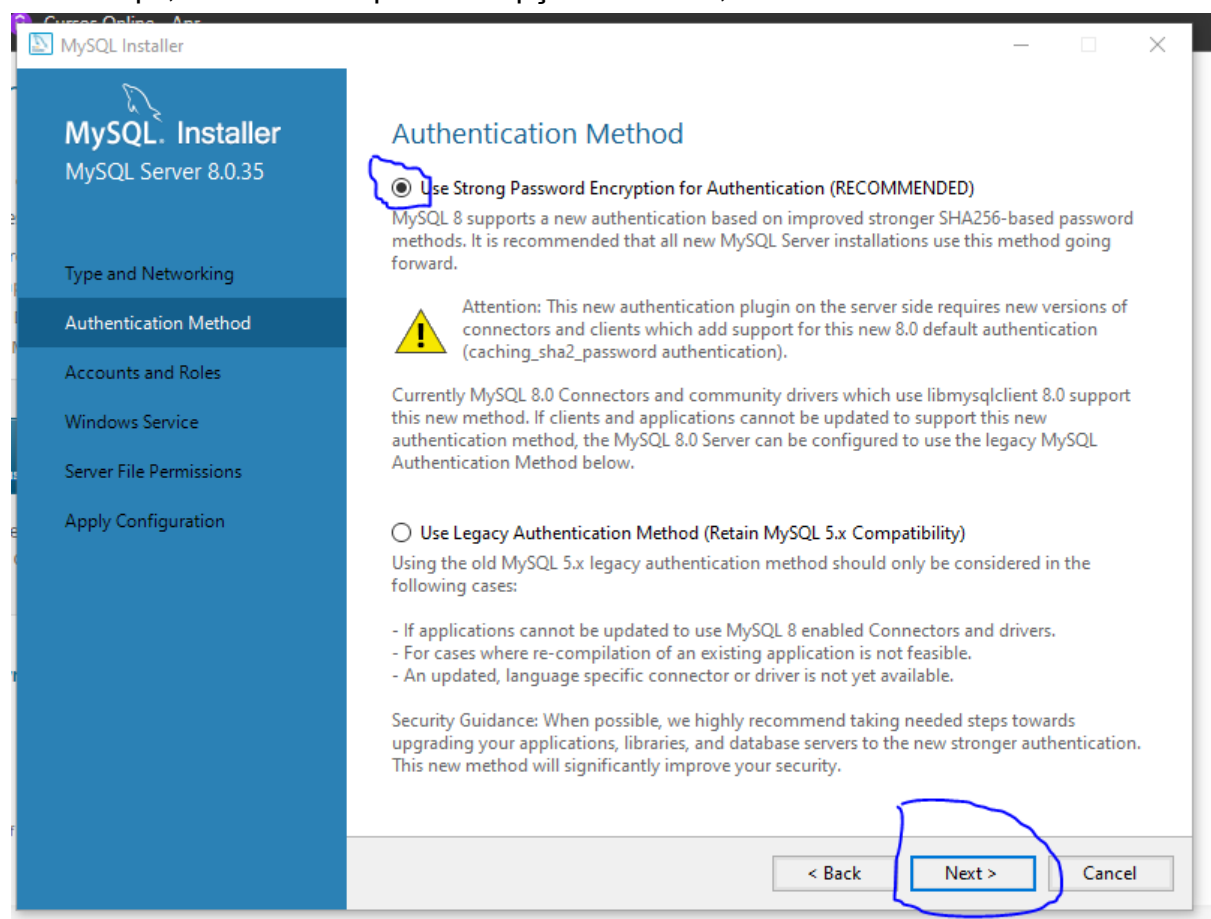
21. Nesta aba clique em “Next”



22. Deixa nas configurações padrões, clica em “Next”



23. Aqui, você deixa a primeira opção marcada, e clica em “Next”.



24. Nesta parte você irá preencher com os dois formulários com a mesma senha, depois clique em “**Next**”. Para essa aplicação, é preferível a senha “**password**”, caso escolha outra senha, lembre-se dela, pois irá precisar fazer alterações no código, que será explicado mais para frente neste documento.

MySQL Installer
MySQL Server 8.0.35

Type and Networking
Authentication Method
Accounts and Roles
Windows Service
Server File Permissions
Apply Configuration

Accounts and Roles

Root Account Password
Enter the password for the root account. Please remember to store this password in a secure place.

MySQL Root Password:

Repeat Password:

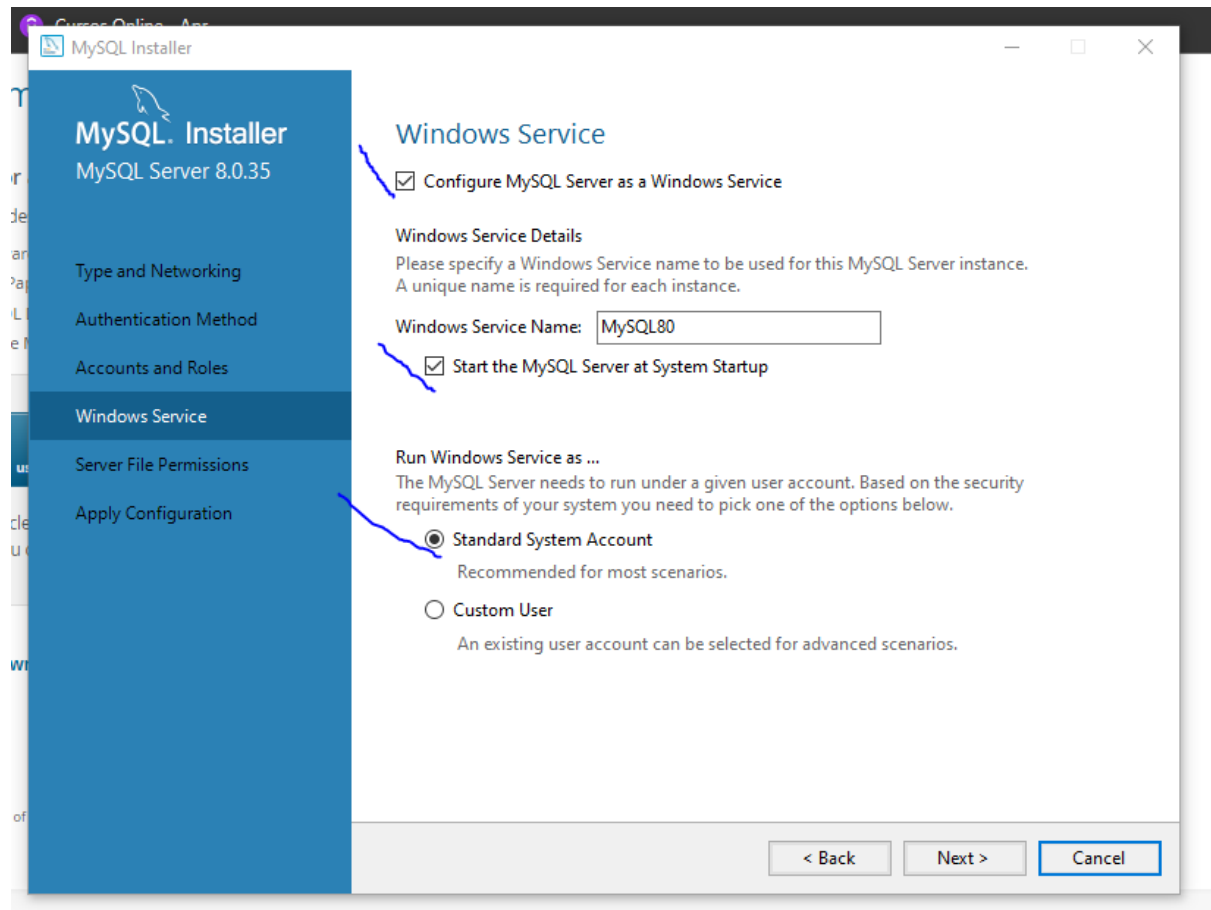
Password strength: **Weak**

MySQL User Accounts
Create MySQL user accounts for your users and applications. Assign a role to the user that consists of a set of privileges.

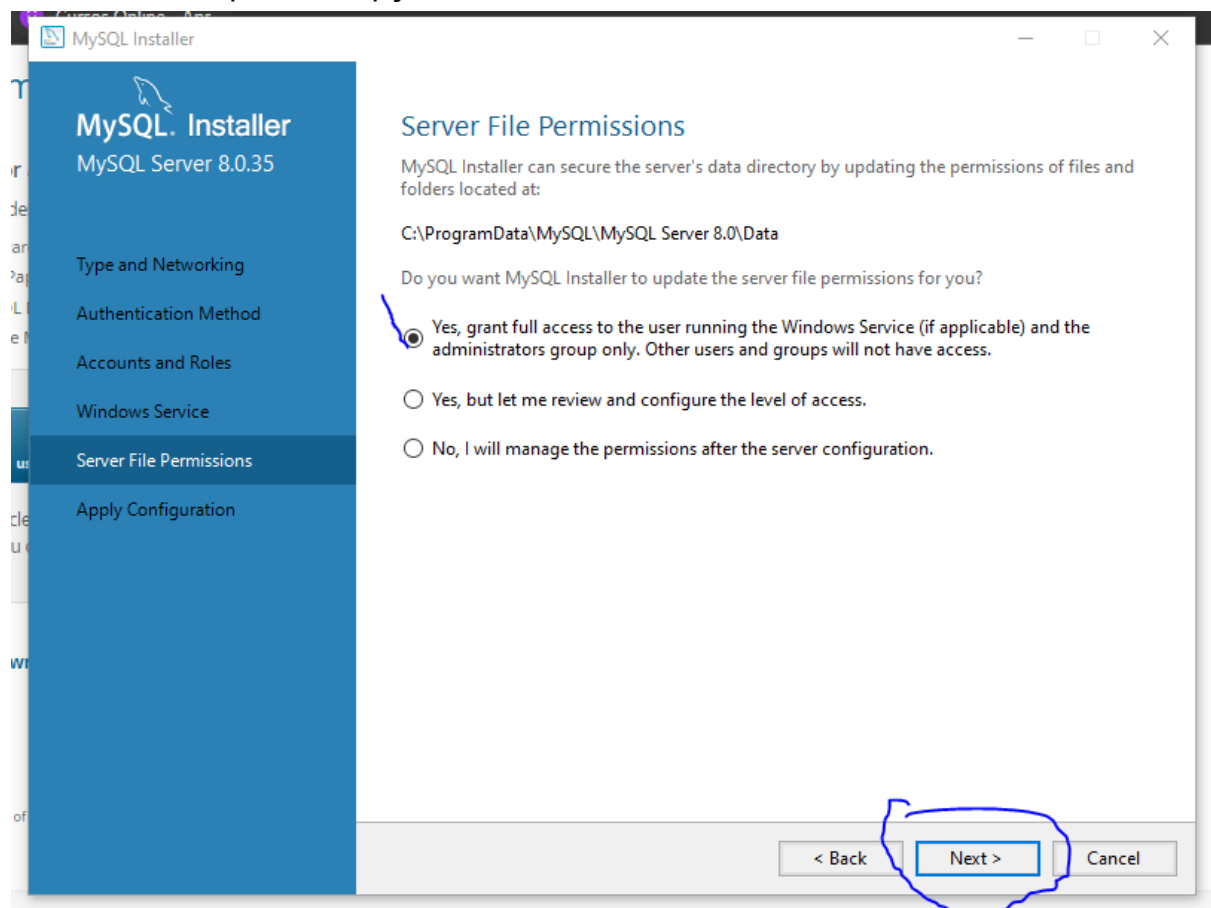
MySQL User Name	Host	User Role
-----------------	------	-----------

< Back **Next >** Cancel

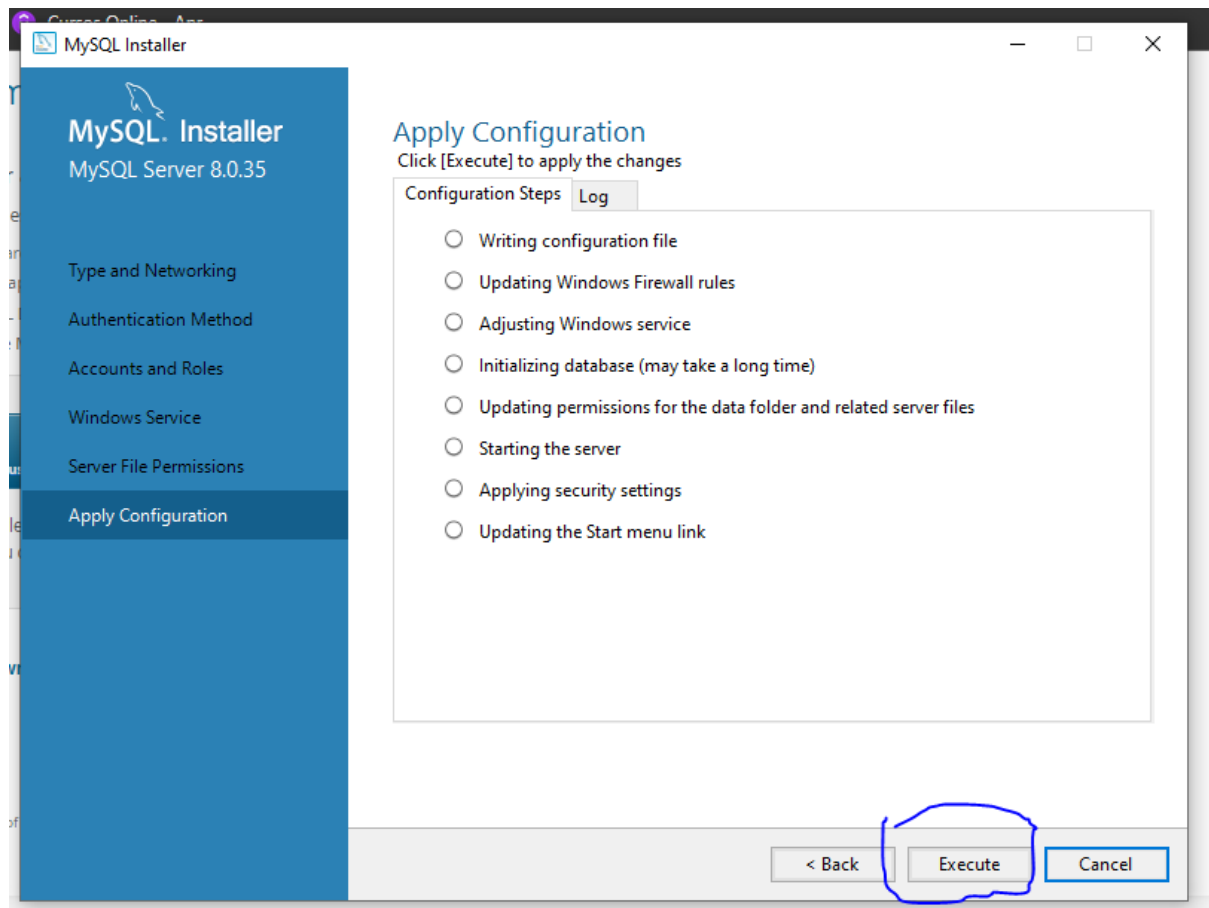
25. Deixa essas três opções marcadas e clique em “Next”



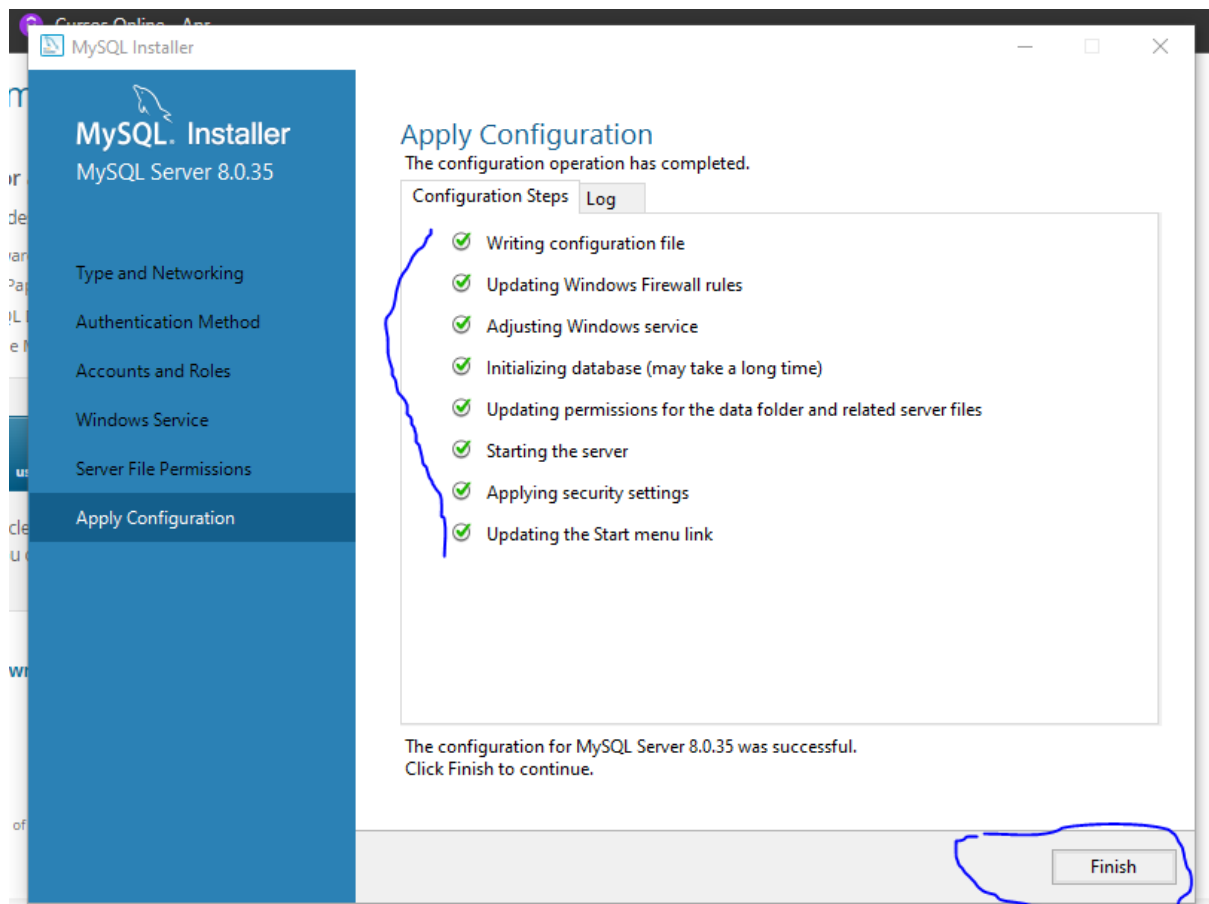
26. Deixa a primeira opção marcada e clica em “Next”



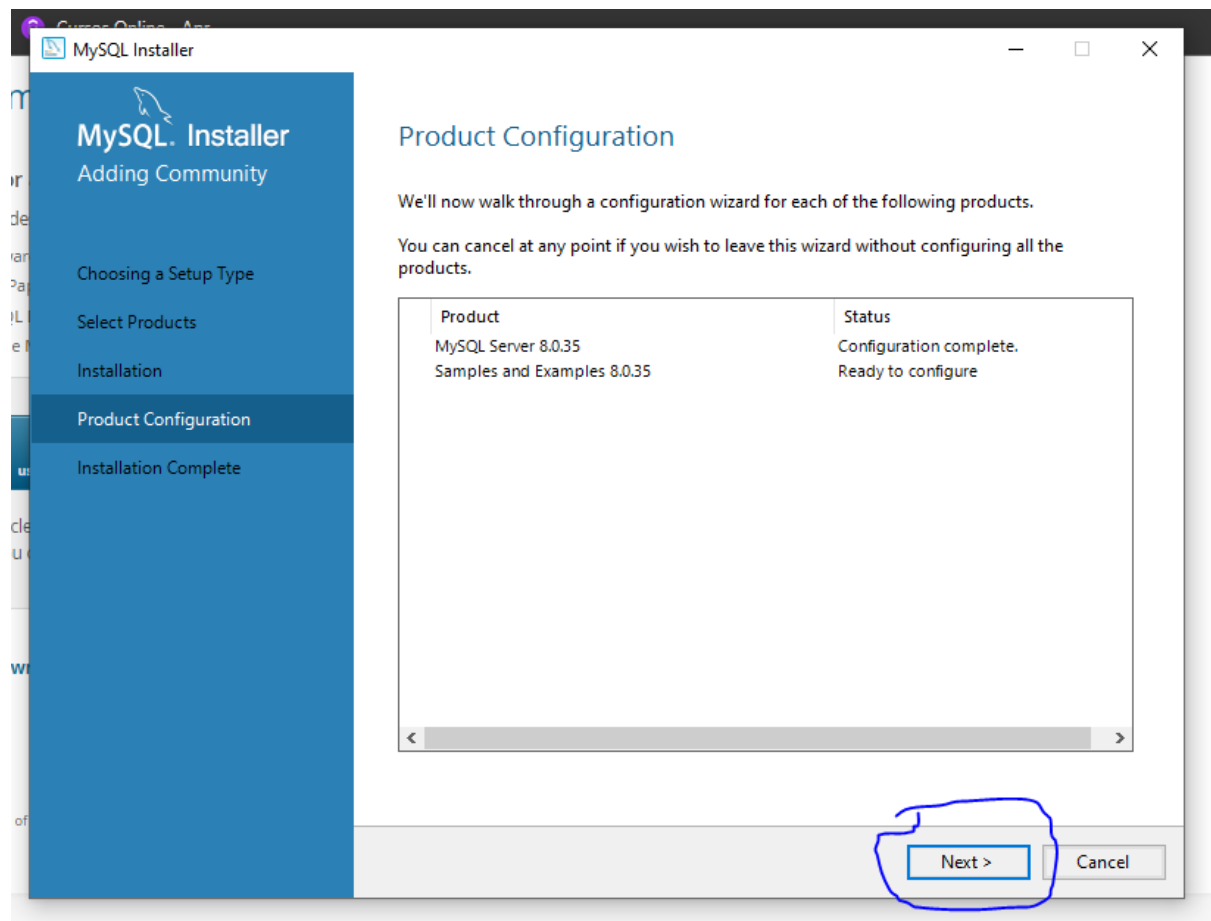
27. Aqui é será feita algumas configurações, e é só clicar em **“Execute”**



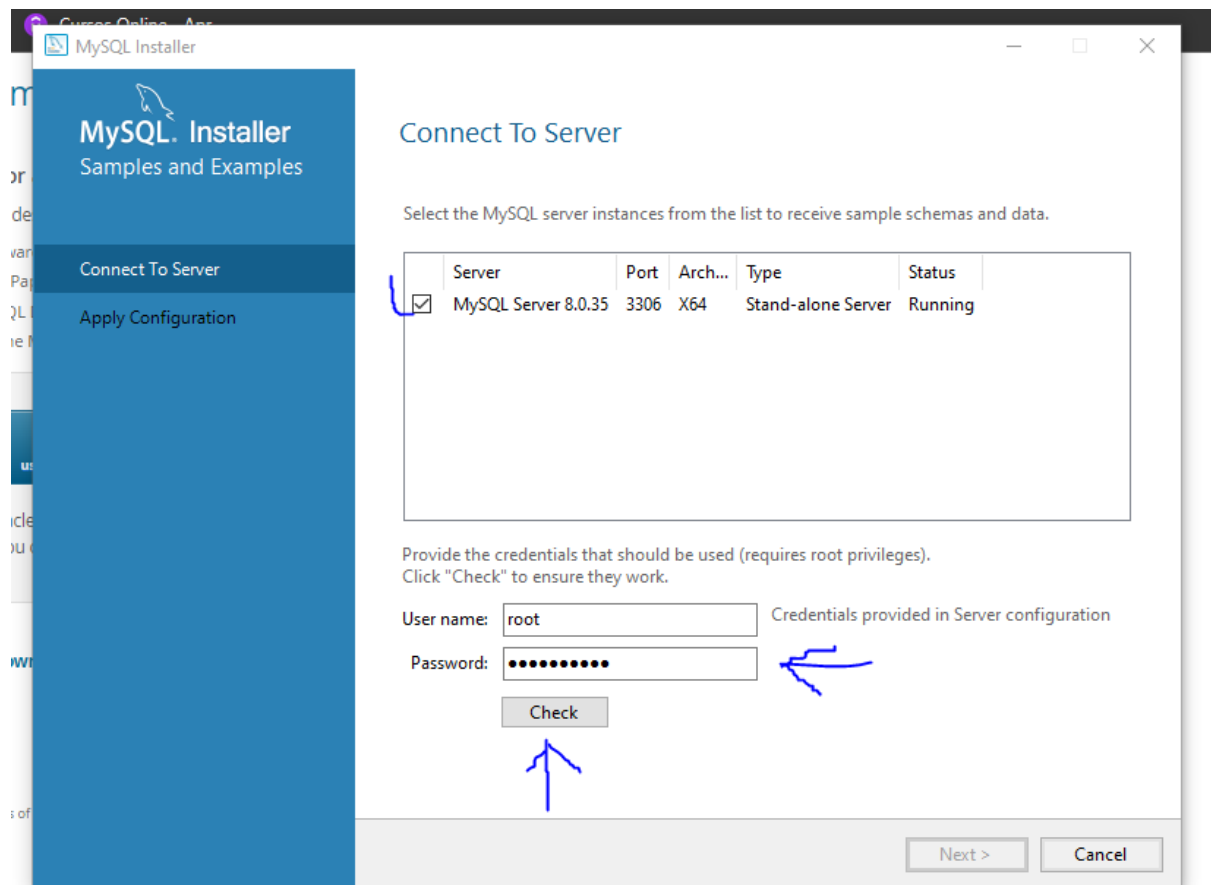
28. Depois de configurado, clicar em **“Finish”**



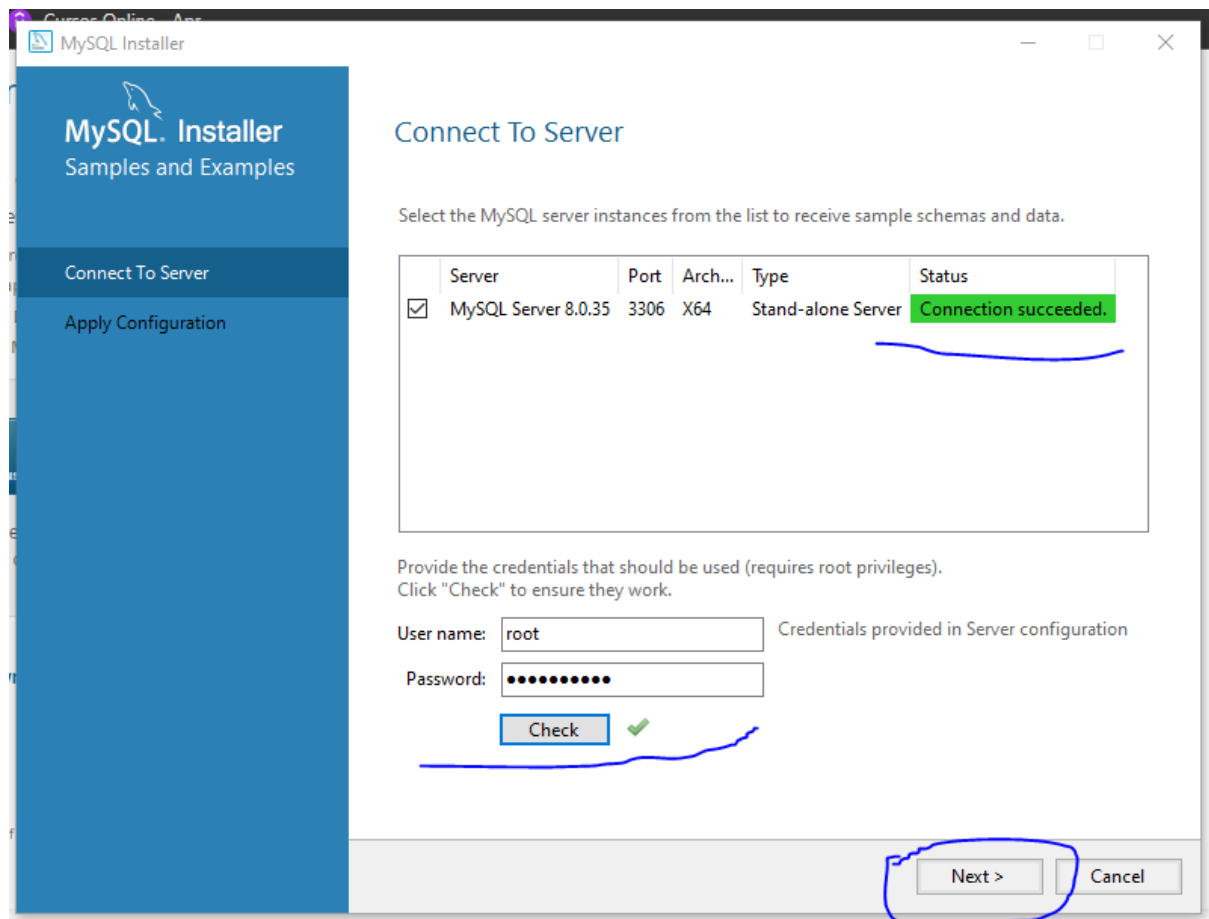
29. Clique em “Next” para avançar



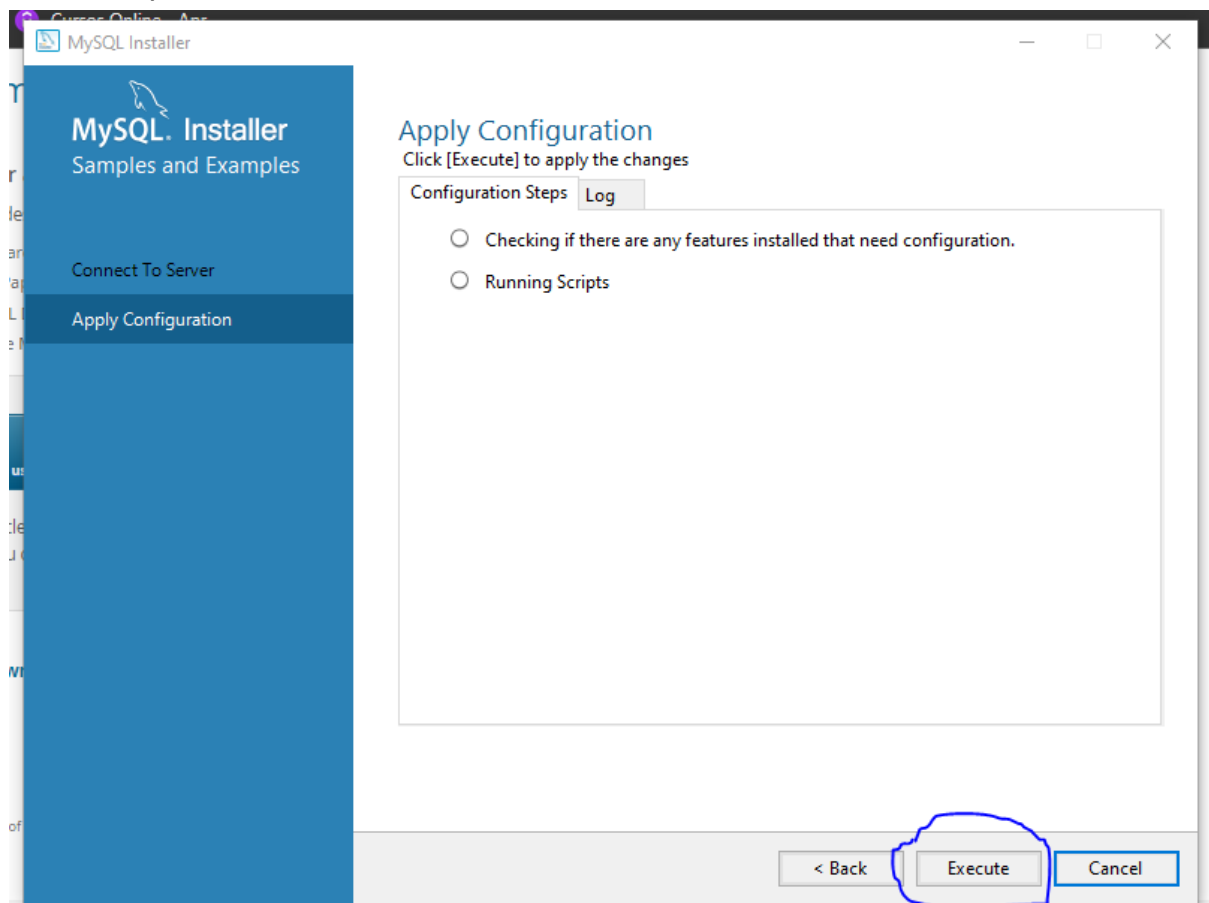
30. Aqui você irá preencher com senha que cadastrou no **passo 24**, e clique em “Check” também lembre-se do seu “User name”.



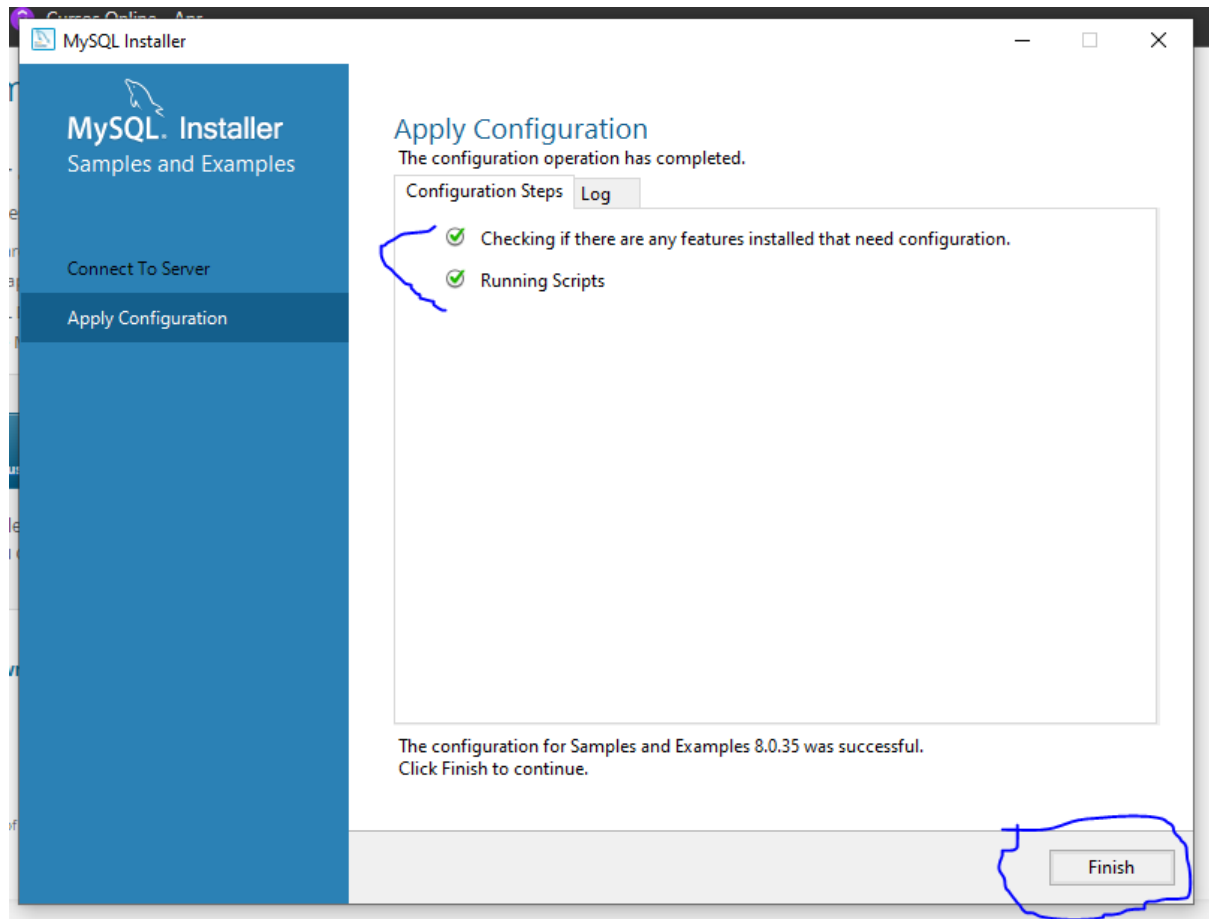
31. Depois de Verificado, clique em "Next"



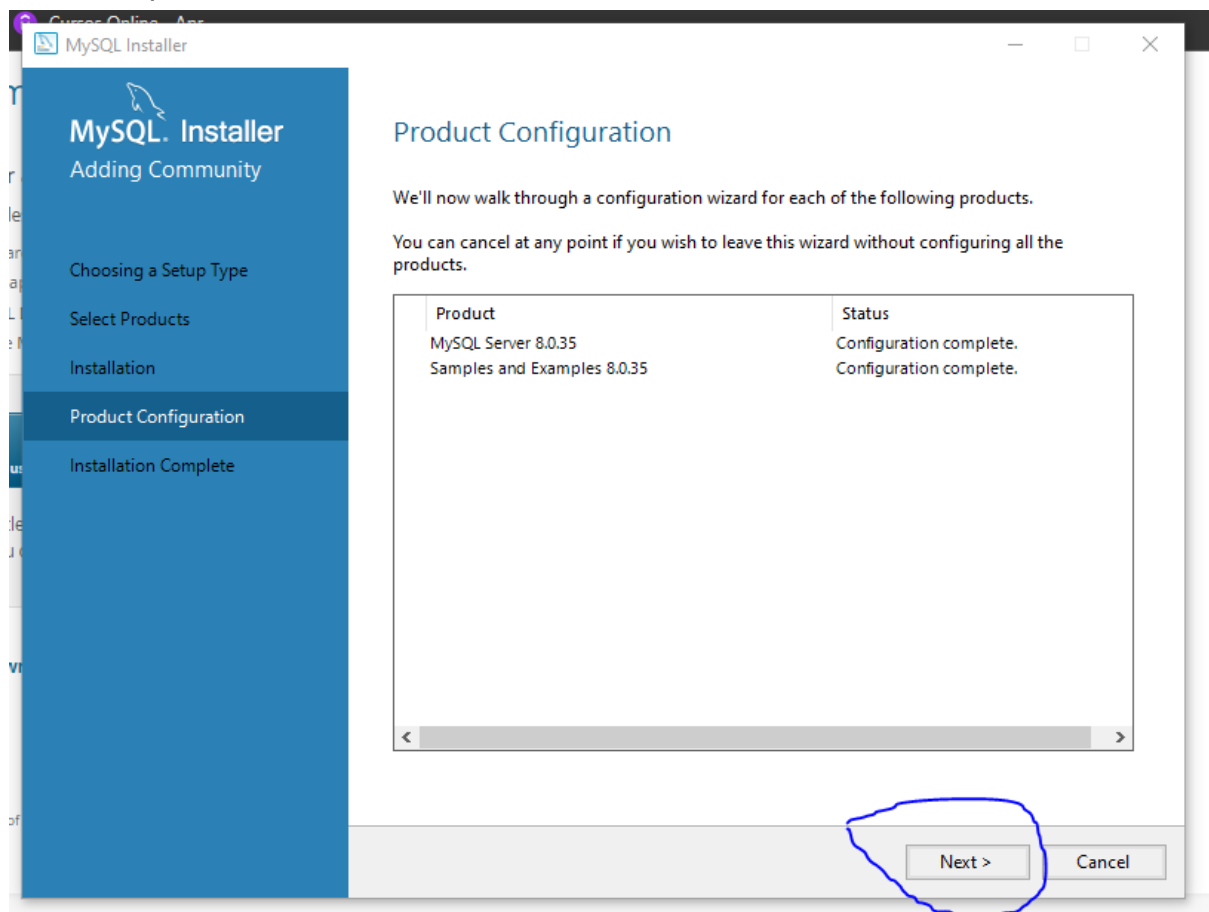
32. Clique em "Execute"



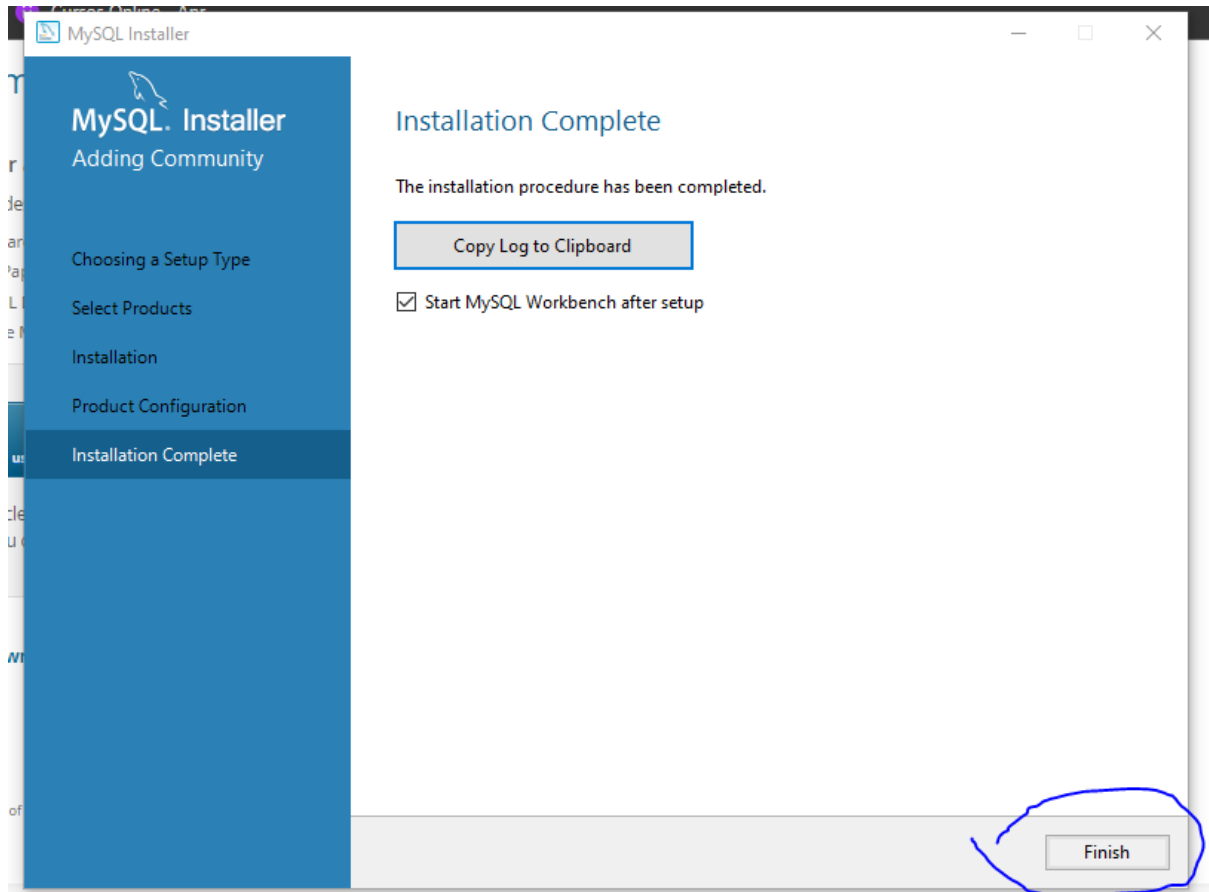
33. Depois de verificado, clique em **“Finish”**



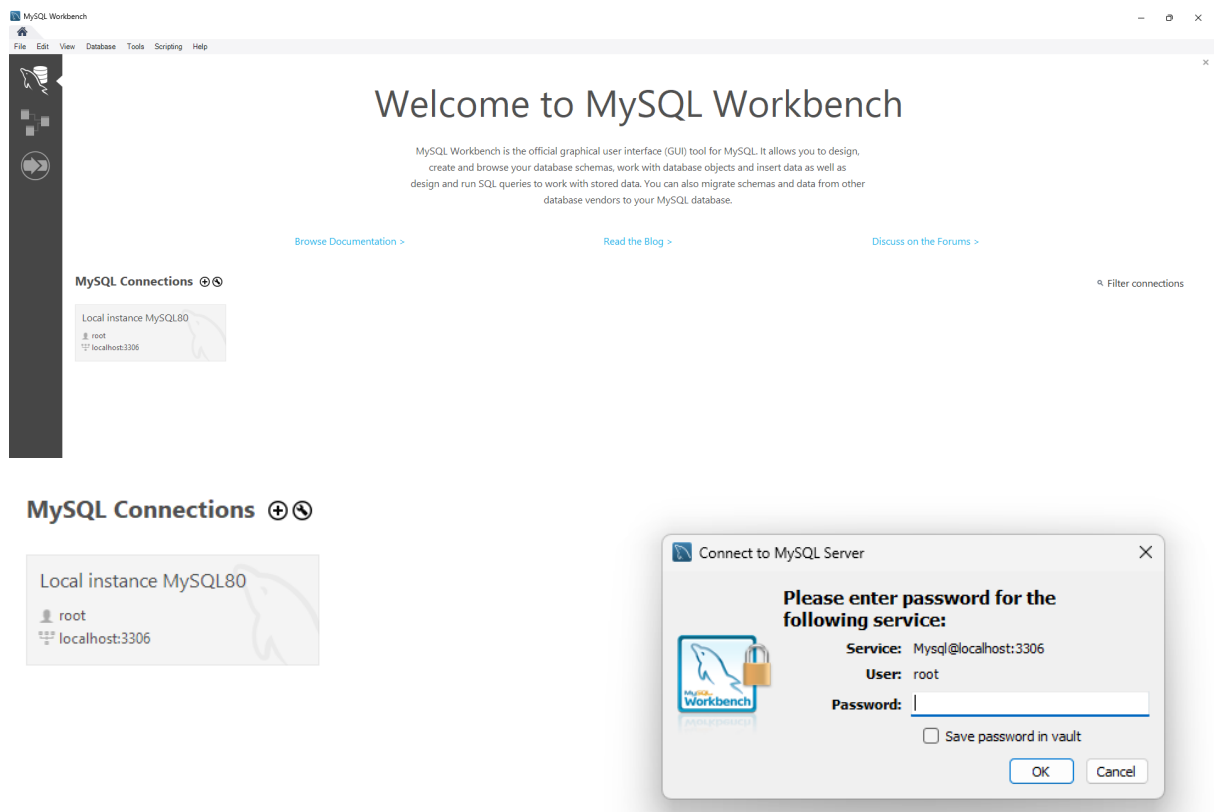
34. Clique em **“Next”**



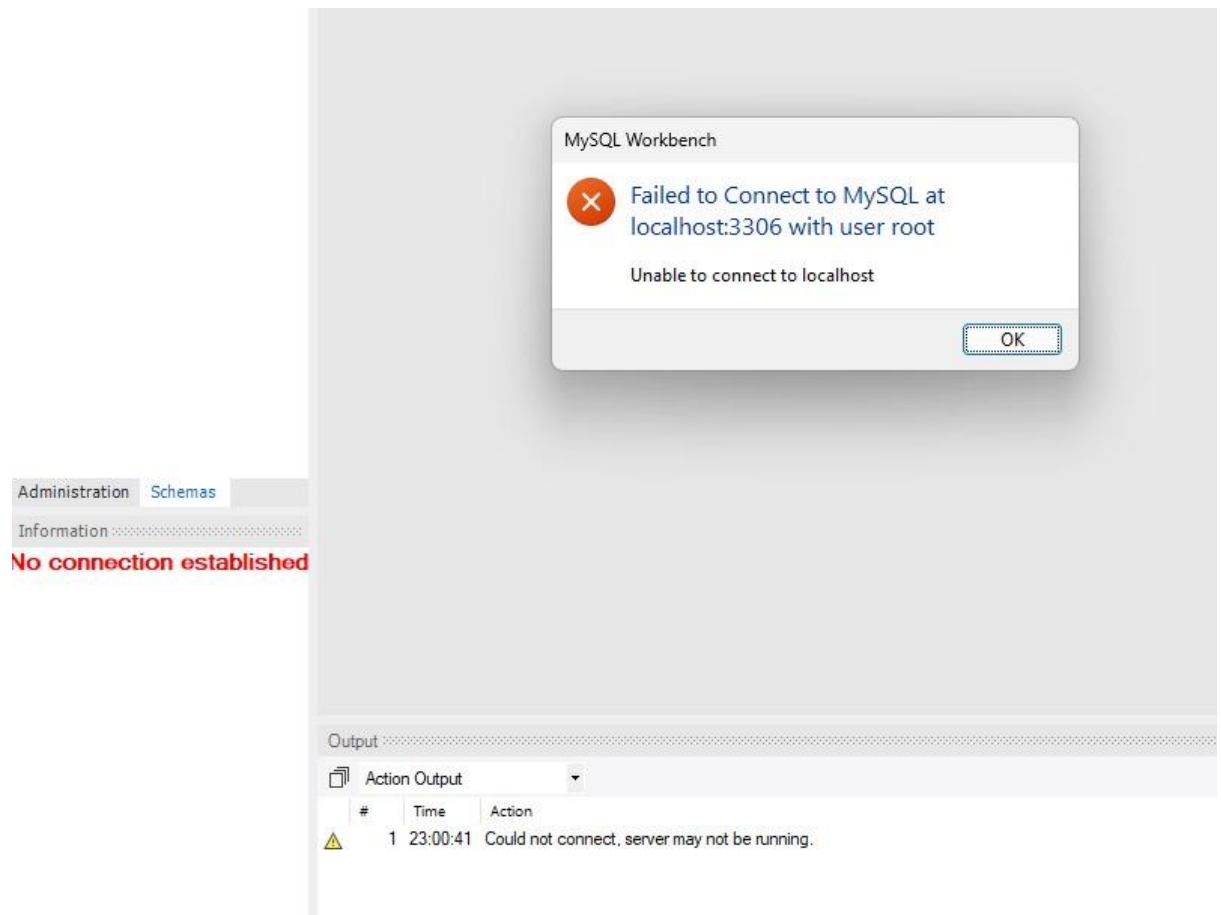
35. Agora clique em “**Finish**” para encerrar a instalação



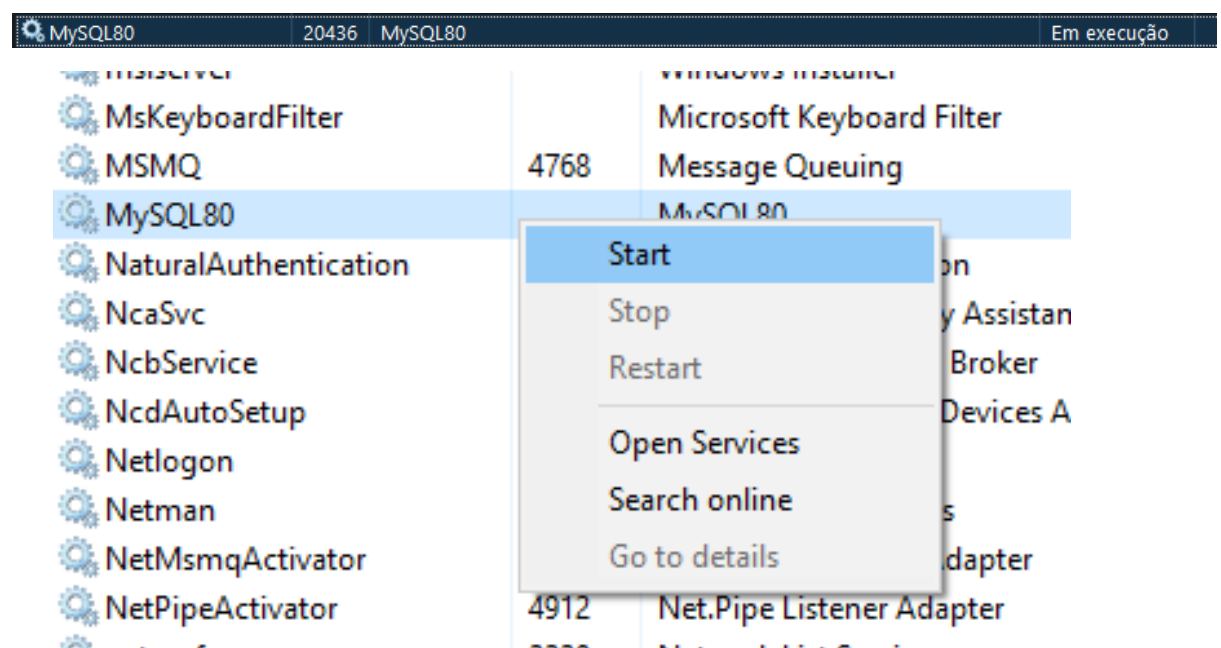
36. Em “MySQL Connections” clique em “Local instance MySQL80” e insira sua senha que foi configurada durante a instalação.



37. Caso apareça este erro, abra o gerenciador de tarefas e inicie o serviço “MySQL80”.

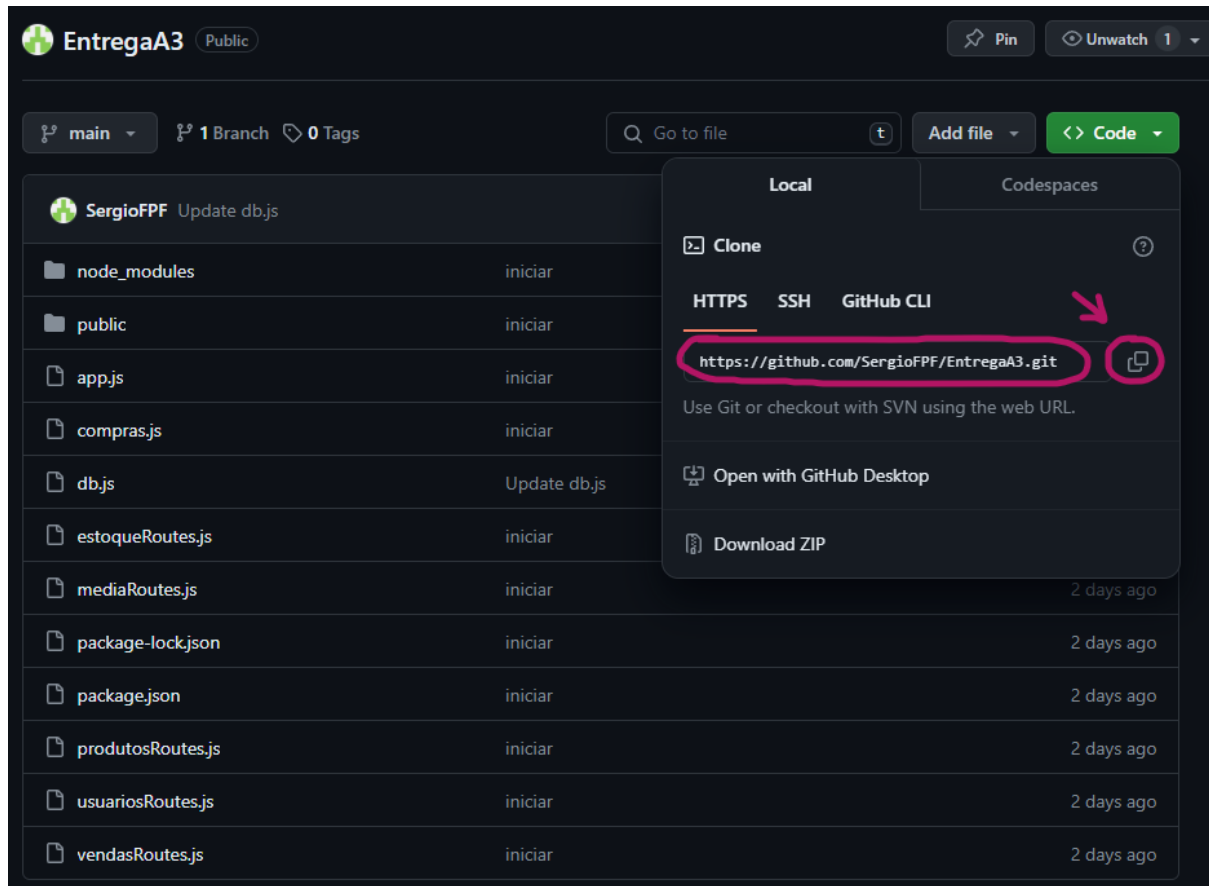


38. Se tiver parado clique com o botão direito e clique em iniciar.

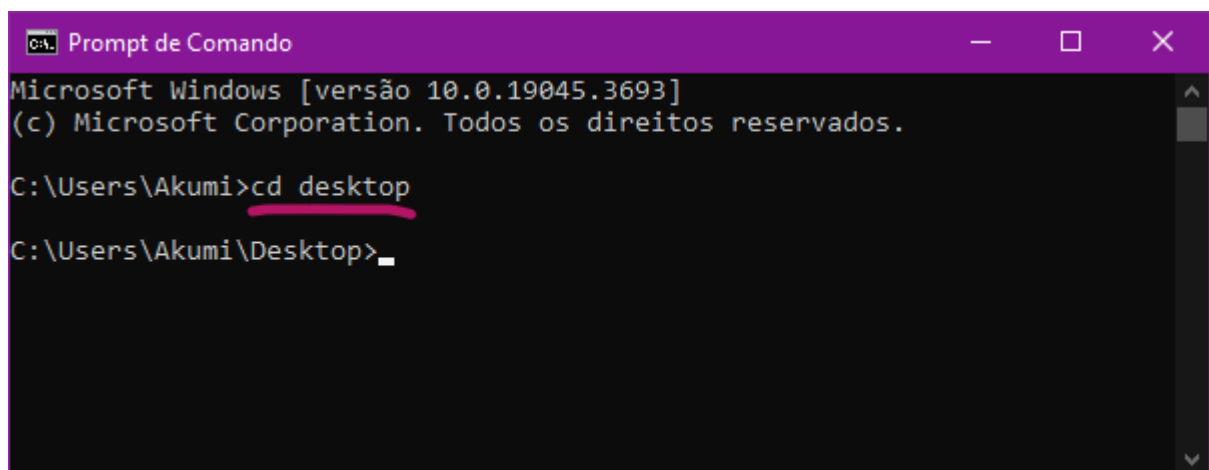


Download do código:

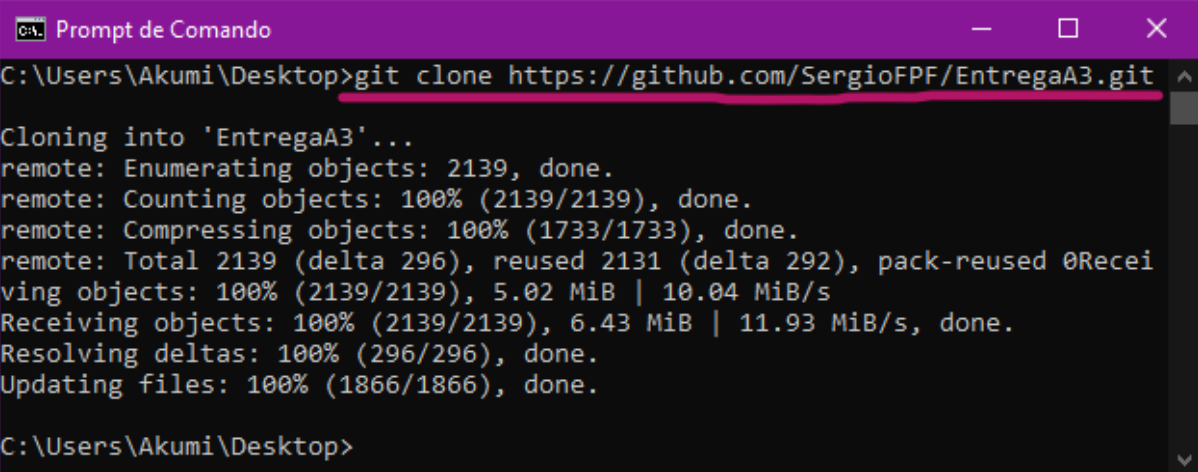
1. No repositório Clique no botão verde "Code" (ou "Código") perto do topo da página.
 - Certifique-se de que a opção "HTTPS" esteja selecionada.
 - Copie a URL do repositório.



2. Abra o prompt de comando no seu computador e use o comando "cd" para navegar até o diretório onde você deseja baixar o repositório. Neste exemplo "cd desktop" ou "cd C:\Users\SeuUsuárioAqui\Desktop" é para iniciar área de trabalho.



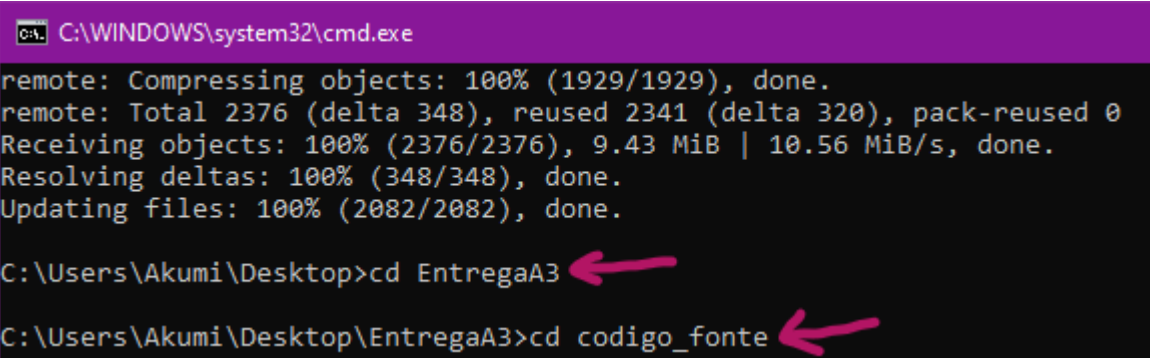
3. Cole a URL do repositório que você copiou anteriormente e execute o comando "git clone https://github.com/SergioFPF/EntregaA3.git".



```
Prompt de Comando
C:\Users\Akumi\Desktop>git clone https://github.com/SergioFPF/EntregaA3.git
Cloning into 'EntregaA3'...
remote: Enumerating objects: 2139, done.
remote: Counting objects: 100% (2139/2139), done.
remote: Compressing objects: 100% (1733/1733), done.
remote: Total 2139 (delta 296), reused 2131 (delta 292), pack-reused 0
Receiving objects: 100% (2139/2139), 5.02 MiB | 10.04 MiB/s
Receiving objects: 100% (2139/2139), 6.43 MiB | 11.93 MiB/s, done.
Resolving deltas: 100% (296/296), done.
Updating files: 100% (1866/1866), done.
C:\Users\Akumi\Desktop>
```

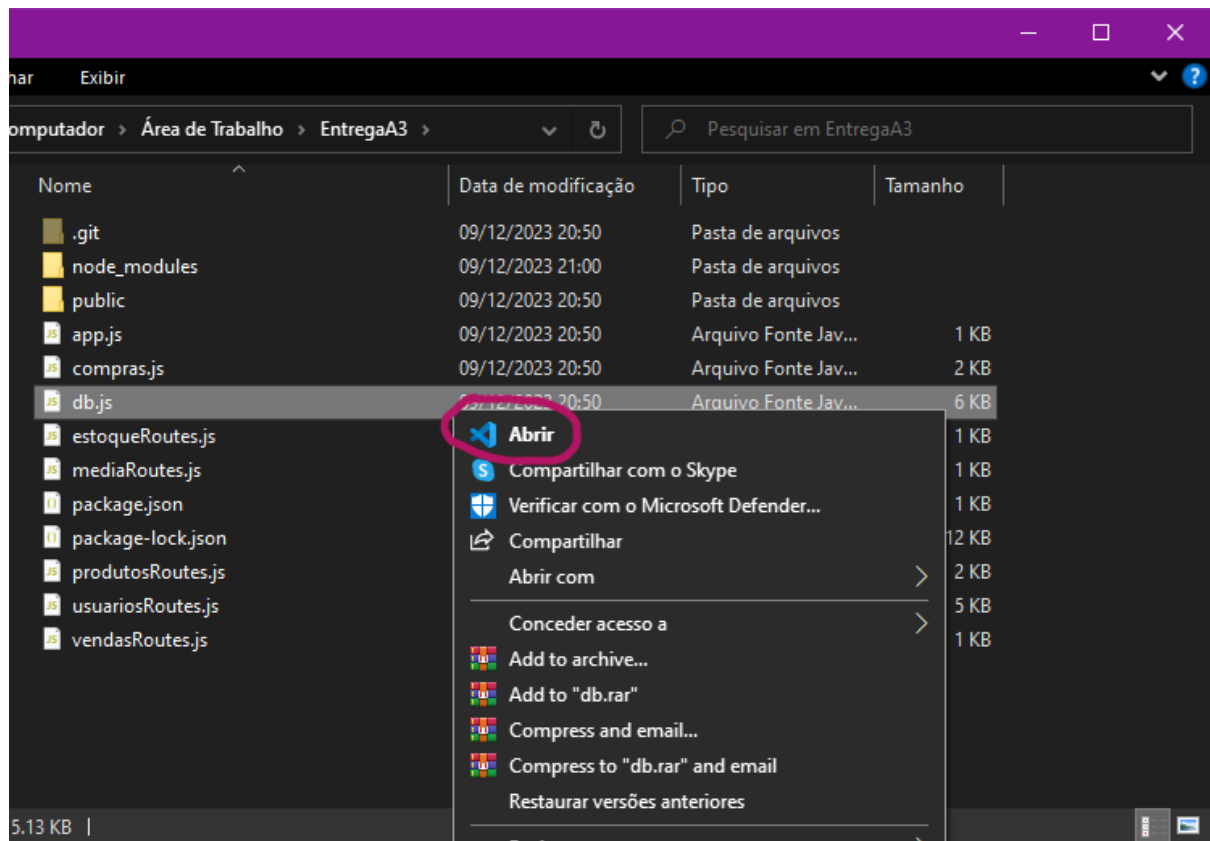
Pode-se notar que a pasta chamada "EntregaA3" foi criada no local escolhido.

4. digite "cd EntregaA3" e em seguida "cd codigo_fonte" para acessar o conteúdo da aplicação.

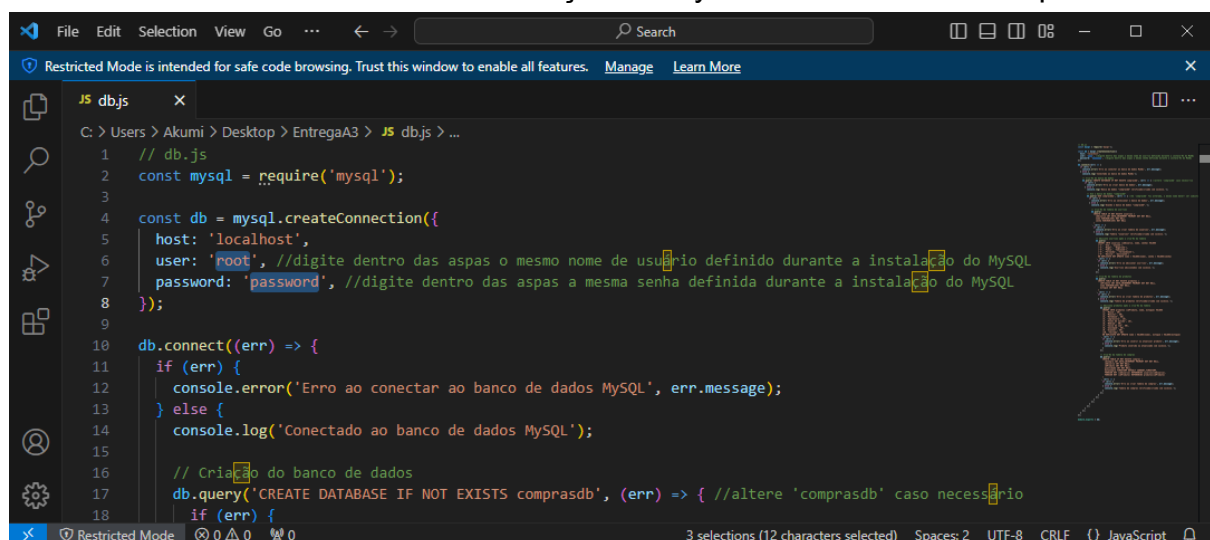


```
C:\WINDOWS\system32\cmd.exe
remote: Compressing objects: 100% (1929/1929), done.
remote: Total 2376 (delta 348), reused 2341 (delta 320), pack-reused 0
Receiving objects: 100% (2376/2376), 9.43 MiB | 10.56 MiB/s, done.
Resolving deltas: 100% (348/348), done.
Updating files: 100% (2082/2082), done.
C:\Users\Akumi\Desktop>cd EntregaA3
C:\Users\Akumi\Desktop\EntregaA3>cd codigo_fonte
```

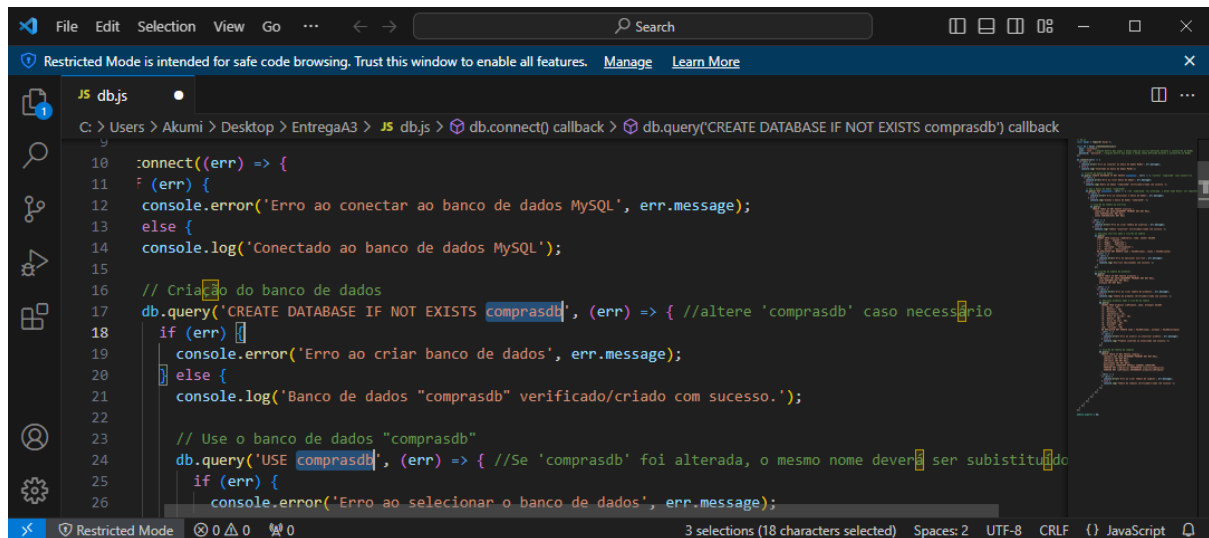

5. Deixaremos o Prompt de Comando de lado por um instante, pois é preciso alterar uma parte crucial do nosso código para que a aplicação funcione corretamente. Abra a pasta e clique com o botão direito em “db.js”, depois selecione para abrir com Visual Studio Code.



6. Na linha 6 e linha 7 edite 'root' e 'password' para o mesmo usuário e senha definidos no momento da instalação do MySQL sem remover os apóstrofes.

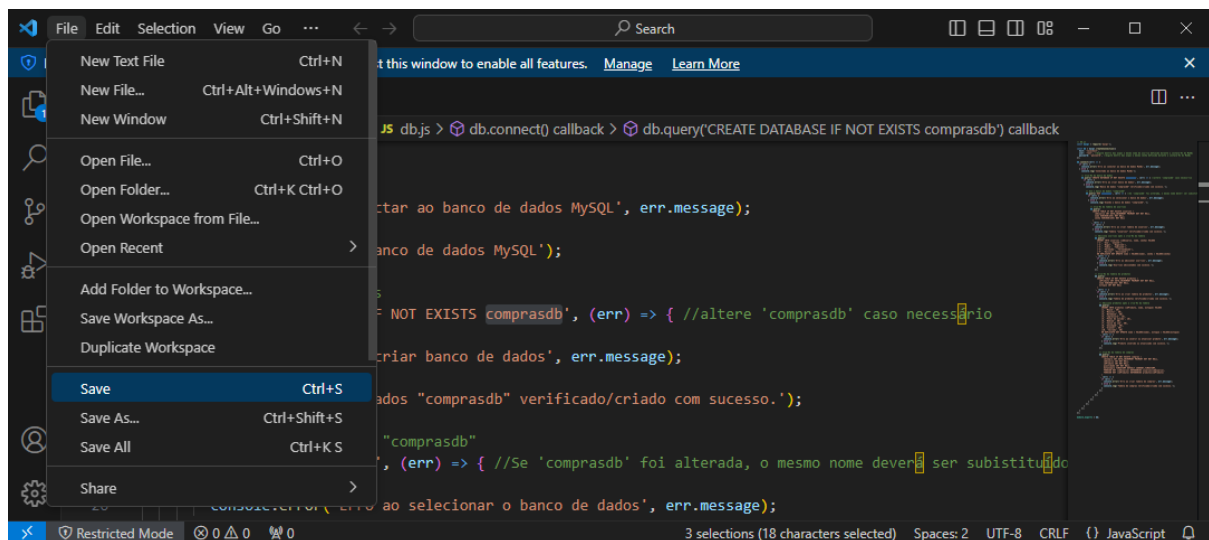


7. Outra parte importante é checar no MySQL se existe alguma base de dados chamado “comprasdb”, caso exista, é necessário excluir ou alterar na linha 17 somente “comprasdb” para o nome da base de dados que preferir e então a aplicação irá criar essa base de dados caso não exista. É preciso alterar na linha 24 “comprasdb” para o mesmo nome que foi alterado na linha 17, isso porque a aplicação irá escolher essa base de dados para executar suas operações.



```
10 connect((err) => {
11   if (err) {
12     console.error('Erro ao conectar ao banco de dados MySQL', err.message);
13   } else {
14     console.log('Conectado ao banco de dados MySQL');
15   }
16   // Criação do banco de dados
17   db.query('CREATE DATABASE IF NOT EXISTS comprasdb', (err) => { //altere 'comprasdb' caso necessário
18     if (err) {
19       console.error('Erro ao criar banco de dados', err.message);
20     } else {
21       console.log('Banco de dados "comprasdb" verificado/criado com sucesso.');
```

8. Terminado as alterações, pressione “ctrl s” em seu teclado ou se preferir, vá no canto superior esquerdo do Visual Studio Code e siga os passos: File > Save.



```
File Edit Selection View Go ... Search
New Text File Ctrl+N
New File... Ctrl+Alt+Windows+N
New Window Ctrl+Shift+N
Open File... Ctrl+O
Open Folder... Ctrl+K Ctrl+O
Open Workspace from File...
Open Recent
Add Folder to Workspace...
Save Workspace As...
Duplicate Workspace
Save Ctrl+S
Save As... Ctrl+Shift+S
Save All Ctrl+K S
Share
Restricted Mode 0 0 0 0 3 selections (18 characters selected) Spaces: 2 UTF-8 CRLF () JavaScript
```

9. 1)Voltando ao Prompt de comando, utilize o comando para instalar as dependências: “npm install”.
- 2)Após instalar, para verificar se ocorreu corretamente o download utilize o seguinte comando: “npm ls”
- 3)Após ter certeza de que está no diretório certo e com o Mysql na máquina, inicie o servidor com o uso do “npm start”.

```
C:\WINDOWS\system32\cmd.exe
C:\Users\Akumi\Desktop\EntregaA3\codigo_fonte>npm install 1
up to date, audited 214 packages in 885ms
18 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities

C:\Users\Akumi\Desktop\EntregaA3\codigo_fonte>npm ls 2
apicompras@1.0.0 C:\Users\Akumi\Desktop\EntregaA3\codigo_fonte
+-- express@4.18.2
+-- mysql@2.18.1
+-- mysql2@3.6.5
+-- nodemon@3.0.2
+-- sqlite3@5.1.6

C:\Users\Akumi\Desktop\EntregaA3\codigo_fonte>npm start 3
> apicompras@1.0.0 start
> nodemon app.js

[nodemon] 3.0.2
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node app.js`
Servidor rodando em http://localhost:3000
Conectado ao banco de dados MySQL
Banco de dados "comprasdb" verificado/criado com sucesso.
Usando o banco de dados "comprasdb".
Tabela "usuarios" verificada/criada com sucesso.
Usuários adicionados com sucesso.
Tabela de produtos verificada/criada com sucesso.
Produto inserido ou atualizado com sucesso.
Tabela de compras verificada/criada com sucesso.
```

A aplicação:

Após a instalação das dependências (npm install) e iniciar a aplicação (npm start), Poderemos realizar as operações através de ferramentas como “Postman” ou “Insomnia”, uma dessas ferramentas serão necessárias para a realização do CRUD nesta aplicação, será explicado mais para frente neste documento como utilizar o “Postman”. Também é possível realizar algumas delas pelo próprio navegador de sua máquina, pois através de HTML foram feitos ajustes para o usuário ver e até interagir com a API, como por exemplo a ação de comprar um produto.

Ao digitar “<http://localhost:3000/>” no navegador, vemos essa página HTML:



Compra de Produto

ID do Usuário:	<input type="text"/>	É preciso utilizar um ID de usuário existente na lista de usuarios
ID do Produto:	<input type="text"/>	É preciso utilizar um ID de produto existente na lista de produtos
Quantidade:	<input type="text"/>	A quantidade não pode ser maior do que possui em estoque
<input type="button" value="Realizar Compra"/>		

Uma nova aba abrirá ao clicar nos links das listas abaixo

[Ver lista de usuários cadastrados](#)

[Ver lista de produtos](#)

[Ver compras realizadas](#) (É preciso **realizar alguma compra** primeiro, então será possível às ver clicando aqui)

[Ver Produtos com Estoque Baixo](#) (Considerado abaixo de 20 no estoque)

[Média de compras por usuário](#) (É preciso realizar uma ou mais compras de um ou mais usuários para visualizar essa lista)

[Produtos mais vendidos](#) (É preciso realizar alguma compra para ser mostrado as vendas feitas nessa lista)

Como explicado no documento, é preciso usar a ferramenta "Postman" para realizar CRUD de usuários e produtos.

Logo de cara vemos que é possível realizar a compra de um produto. Mas que produto? clicando em “[Ver lista de produtos](#)” abrirá uma nova página no navegador com todos os produtos cadastrados, se preferir também é possível digitar o link “<http://localhost:3000/produtos>” para acessar a lista de produtos. Já existem 10 produtos preparados para o teste cadastrados, cada um com seu **ID, nome e estoque**, será necessário lembrar do ID do produto que deseja comprar. Para comprar um produto é muito simples, só precisa pegar o ID do usuário que irá comprar o produto, para escolher um usuário que é só clicar em “[Ver lista de usuários cadastrados](#)” ou acessando através do link “<http://localhost:3000/usuarios>” onde irá encontrar prontamente 5 usuários para teste com **ID, nome e senha**. Com o **ID** de usuário e **ID** de produto voltaremos a nossa página principal e digitamos nos campos “ID do Usuário” e “ID do Produto” respectivamente (lembrando que o ID precisa existir na lista de usuários e lista de produtos), então escolhemos a quantidade inferior à quantidade máxima existente em estoque e clicamos em “Realizar compra” logo abaixo.

Podemos agora testar à vontade mais compras com produtos diferentes e usuários diferentes. Poderá notar o estoque do produto diminuindo após realizar uma compra, novas compras serem cadastradas, novas médias de compras por usuários

sendo calculadas e a atualização dos produtos mais vendidos, todos em seus respectivos url.

Pronto! Agora podemos ver todos os outros relatórios requisitados para esse projeto:

Geração de relatório de produtos mais vendidos: Clicando em [Produtos mais vendidos](#) na página inicial "<http://localhost:3000/>" ou acessando "<http://localhost:3000/produtos-mais-vendidos>".

Em código MySQL uma junção interna (INNER JOIN) é feita de compras com produtos, é somado o número de vendas e ordenado do maior para o menor. Pode ser conferido em: **vendasRoutes.js**.

Geração de relatório de produto por cliente: Clicando em [Ver compras realizadas](#) na página inicial "<http://localhost:3000/>" ou acessando "<http://localhost:3000/compras>".

Combinando dados das tabelas "compras", "usuarios" e "produtos" em código MySQL, uma junção interna (INNER JOIN) é feita, o nome do usuário que fez a compra, o nome do produto comprado, a quantidade comprada e a data da compra e incluindo o ID da compra. Pode ser conferido em **compras.js**.

Geração de relatório de consumo médio do cliente: Clicando em [Média de compras por usuário](#) na página inicial "<http://localhost:3000/>" ou acessando "<http://localhost:3000/media>".

Em código MySQL, é feito a média da quantidade de produtos que um usuário comprou com base nas informações registradas na tabela "compras", através da realização de uma junção interna (INNER JOIN) com a tabela usuários. Pode ser conferido em **mediaRoutes.js**.

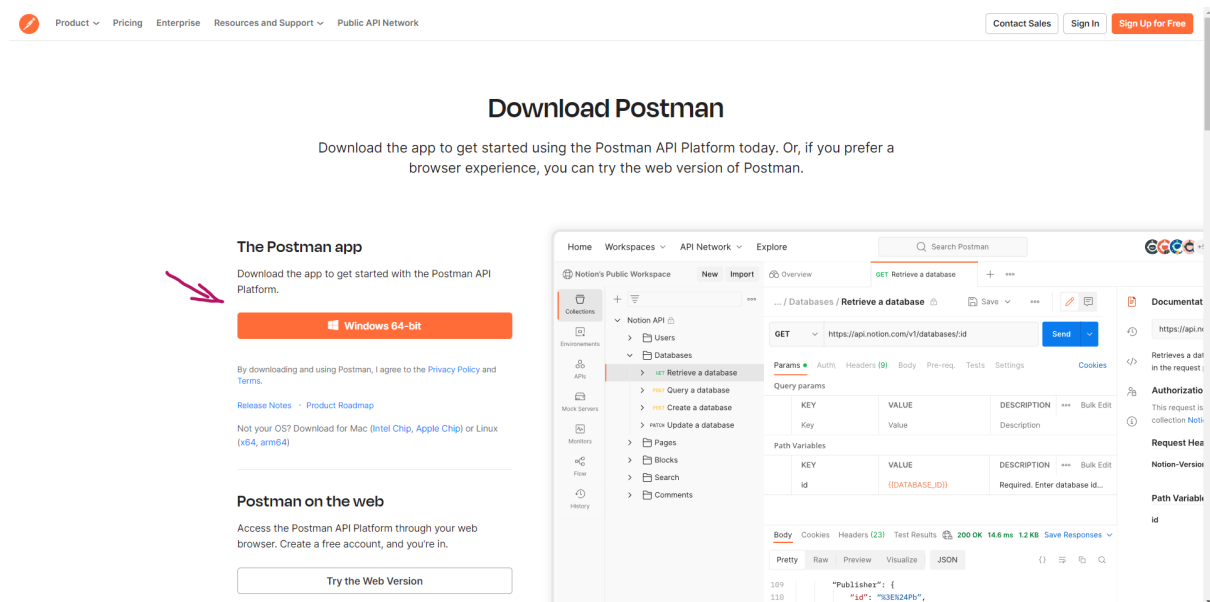
Geração de relatório de produtos com baixo estoque: Clicando em [Ver Produtos com Estoque Baixo](#) na página inicial "<http://localhost:3000/>" ou acessando "<http://localhost:3000/estoque-baixo>".

Um simples código MySQL que pode ser conferido em **estoqueRoutes.js** seleciona todos os produtos com o estoque abaixo de 20.

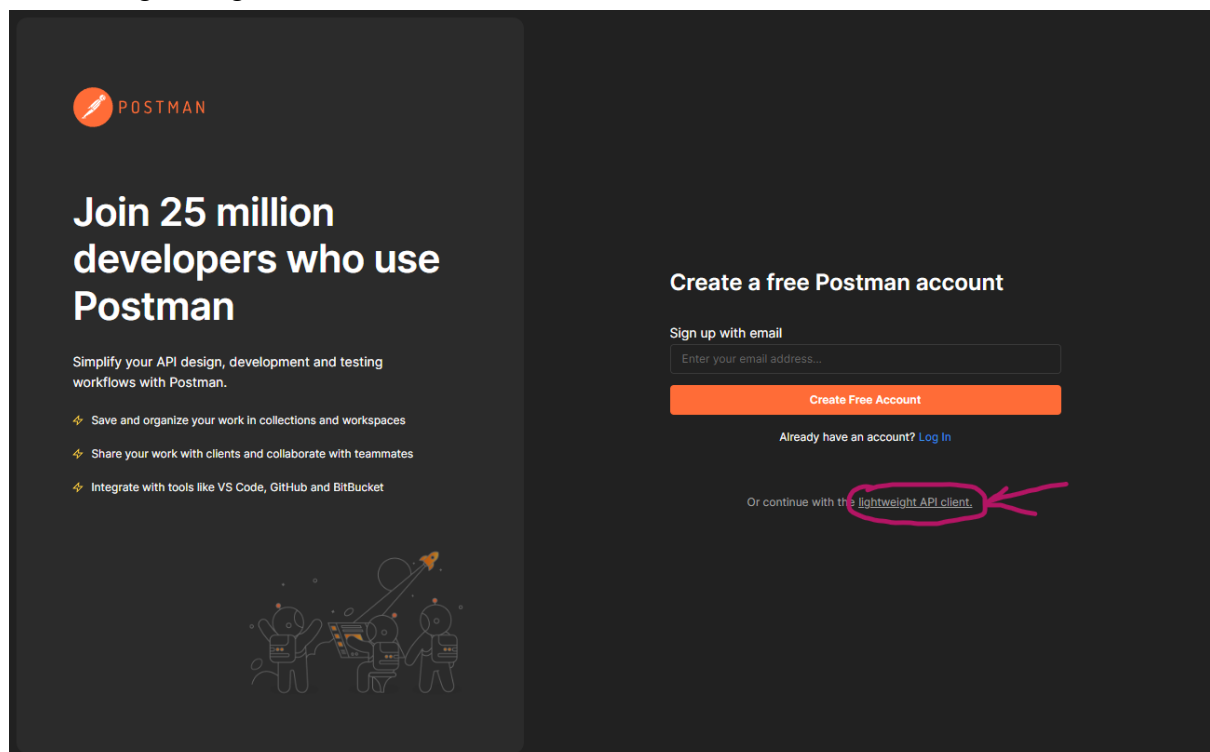
CRUD:

Para realizar o CRUD de usuários e produtos, é necessário utilizar ferramentas como Postman ou Insomnia. Para este projeto, será ensinado como realizar as operações com o Posman

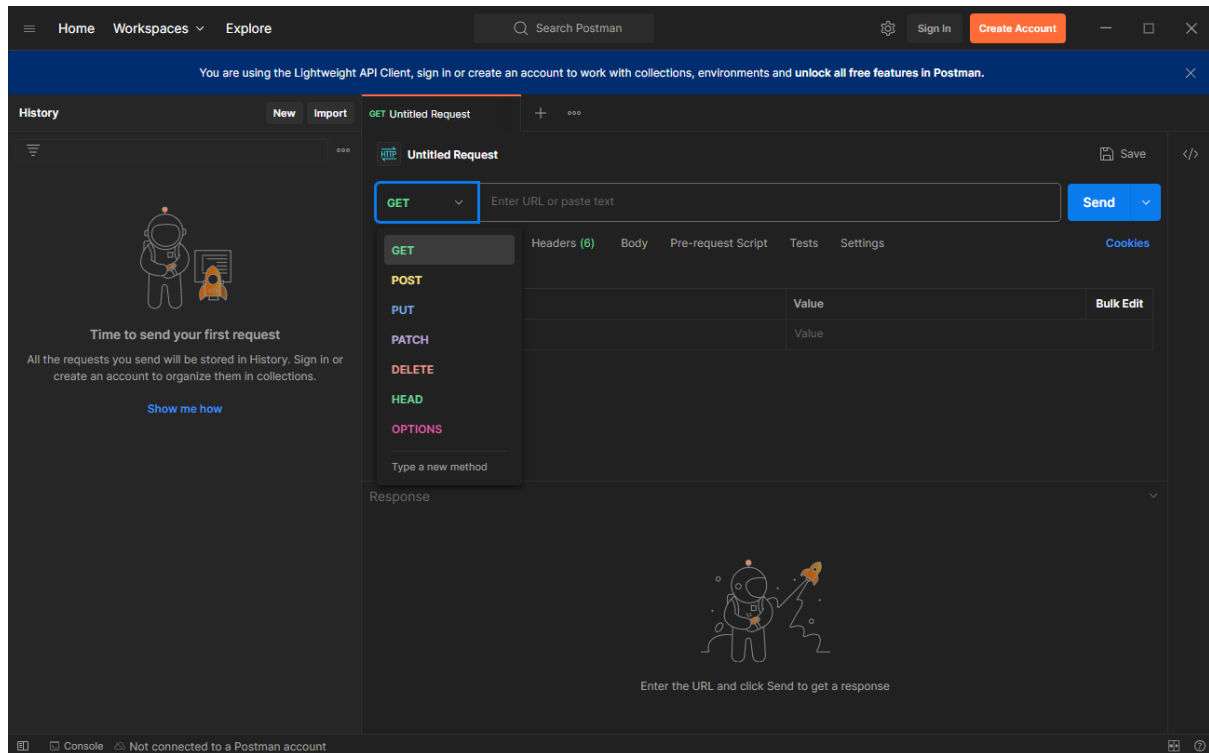
1. instalação: digite em seu navegador: <https://www.postman.com/downloads/>



2. Para realizar as operações, não é preciso criar uma conta, clique em “Lightweight API client”.



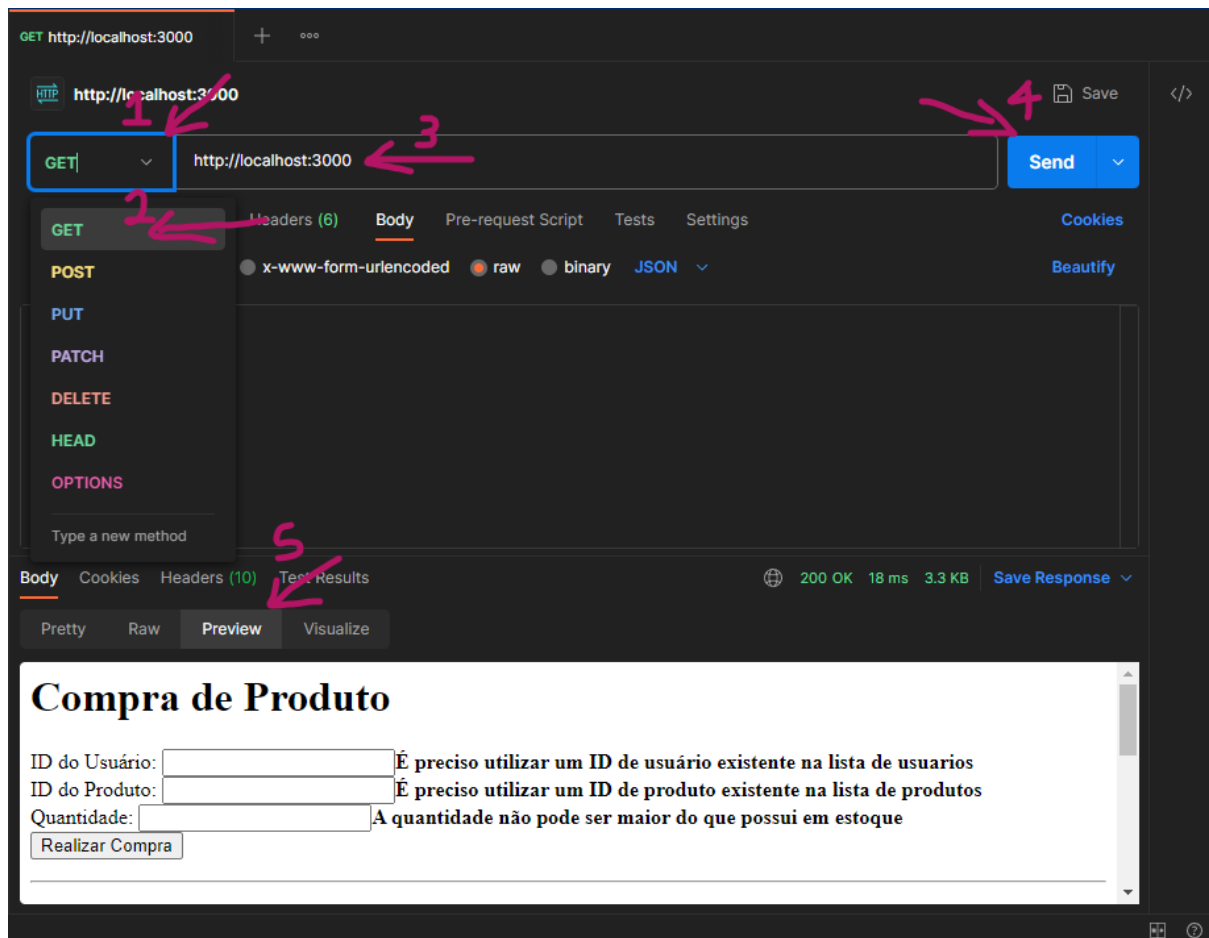
3. Agora é possível realizar as operações “GET”, “POST”, “PUT”, e “DELETE”.



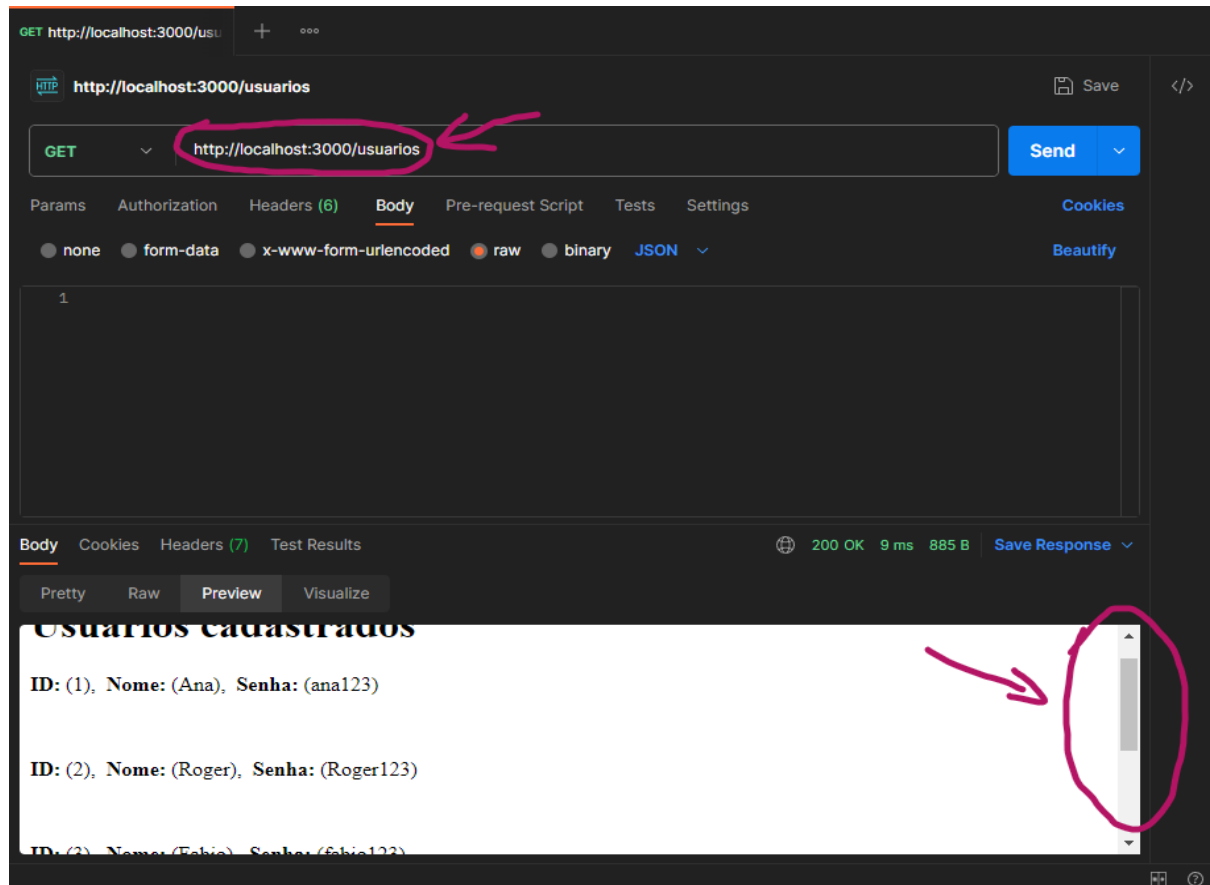
4. Verifique se o servidor foi iniciado corretamente no prompt de comando. É possível ver a url necessária para realizar o get “<http://localhost:3000>”. O usuário deverá seguir os passos como mostrado na imagem:

- 1)Selecionar o método
- 2)Selecionar “GET”
- 3)digitar no campo “<http://localhost:3000>”
- 4)Clicar em “Send”
- 5)Clicar na aba “Preview”

Essa é a nossa página inicial em HTML que também pode ser visualizada no navegador digitando “<http://localhost:3000>”. Porém apenas por essa ferramenta será possível realizar CRUD de usuários e produtos.



5. Agora que sabemos utilizar o Postman, vamos realizar um “GET” para ver todos os usuários e produtos cadastrados. “<http://localhost:3000/usuarios>” deve ser digitado na barra de pesquisa, essa é a nossa rota para ver todos os usuários cadastrados, e “<http://localhost:3000/produtos>” para ver produtos, após clicar em “Send” será exibido todos os usuários ou os produtos na aba “Preview”. Também é possível rolar para baixo utilizando a barra lateral para poder ver os demais usuários cadastrados.



6. Para cadastrar um usuário, utilize a rota “<http://localhost:3000/usuarios>” com o método “**POST**”, é preciso passar os parâmetros em “JSON” para realizar a operação, a nossa aplicação precisa de um nome e uma senha para registrar no banco de dados, não pode ser nulo. “Novo usuário” e “nova senha” será onde deve ser alterado para o nome e senha respectivamente. Depois será possível ver qual o ID desse novo usuário. Siga os passos:

1)Selecionar “**POST**”

2)digitar “<http://localhost:3000/usuarios>”

3)Selecionar “**body**”

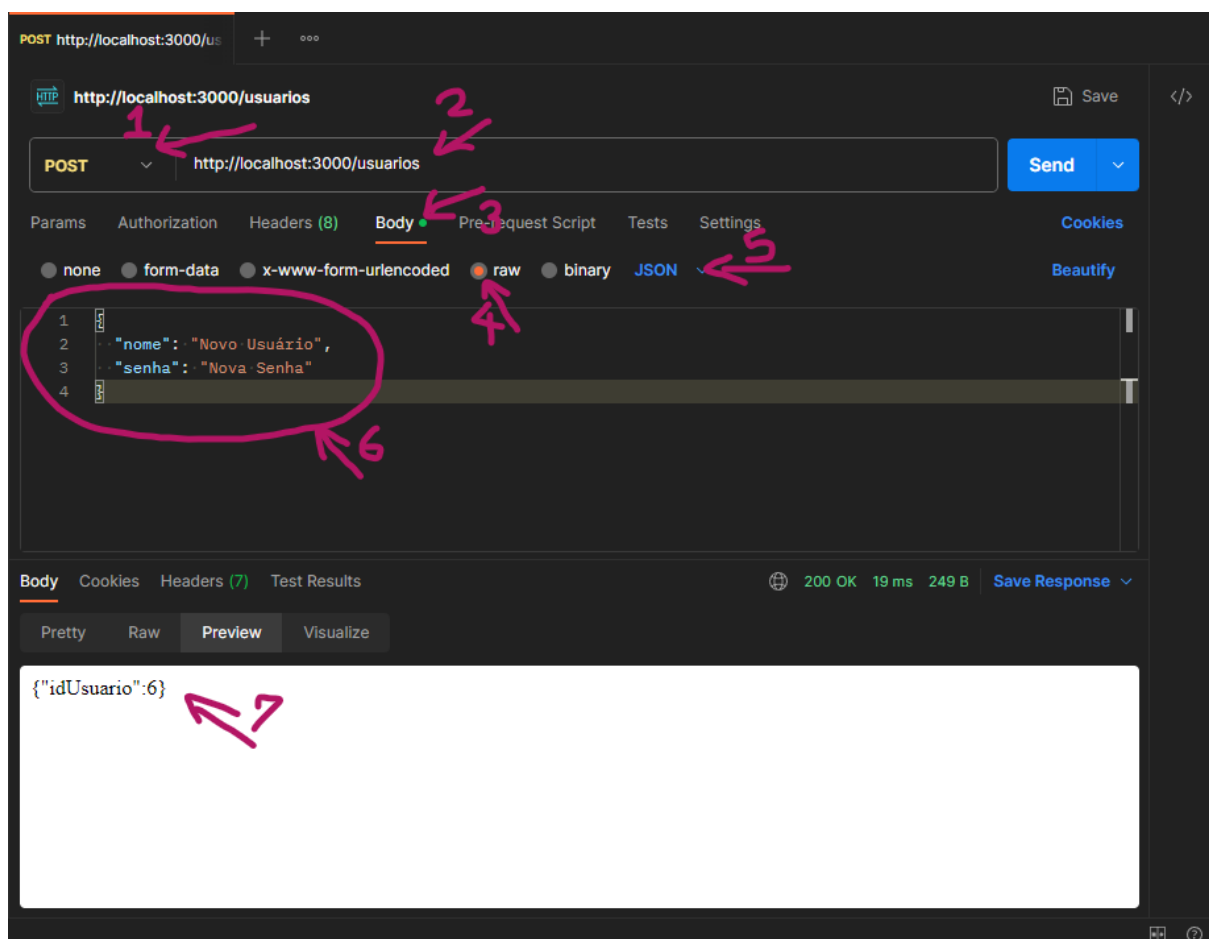
4)Selecionar “**raw**”

5)Selecionar “**JSON**”

6)Digitar no corpo em formato “**JSON**”:

```
{  
  "nome" : "Novo usuário",  
  "senha" : "Nova senha"  
}
```

7)Será Mostrado o ID do novo usuário inserido.



Para produtos a fórmula é a mesma, porém o “nome” é o nome do produto e não tem senha, mas é preciso definir um valor inteiro para o estoque.

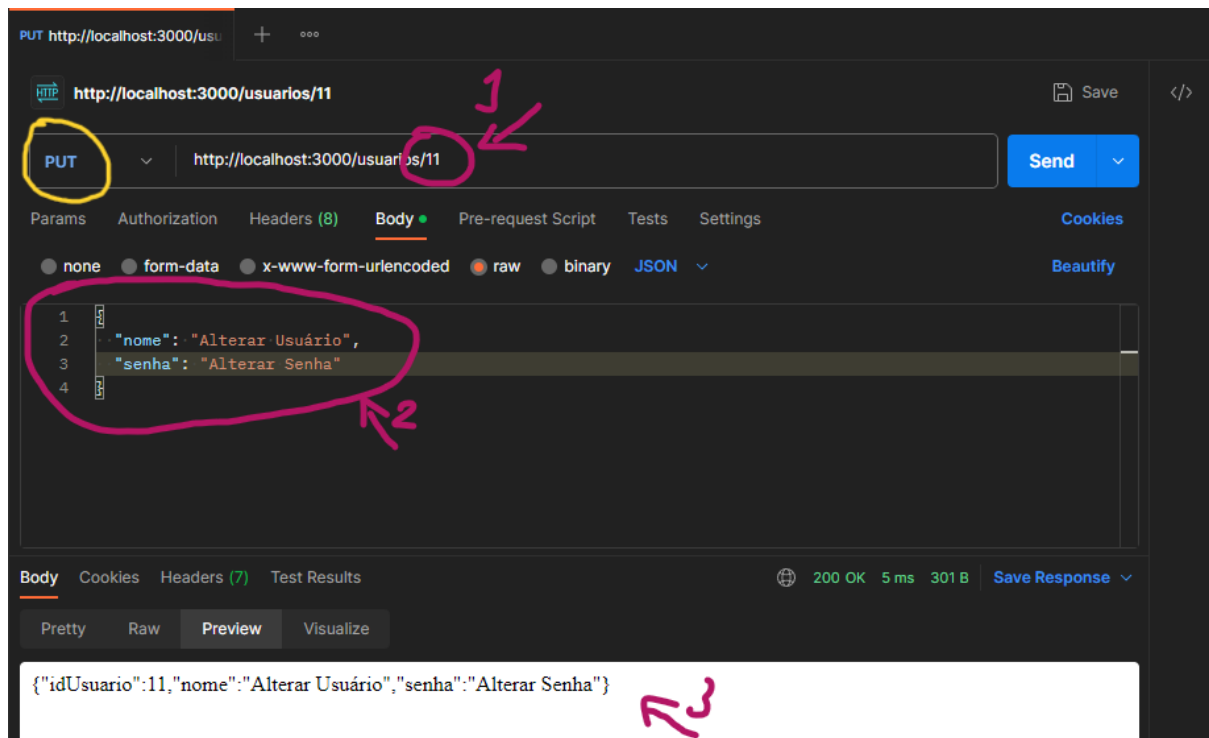
***Nota: 19 ou menos é considerado “estoque baixo” pela aplicação.**

The screenshot shows a REST client interface with the following details:

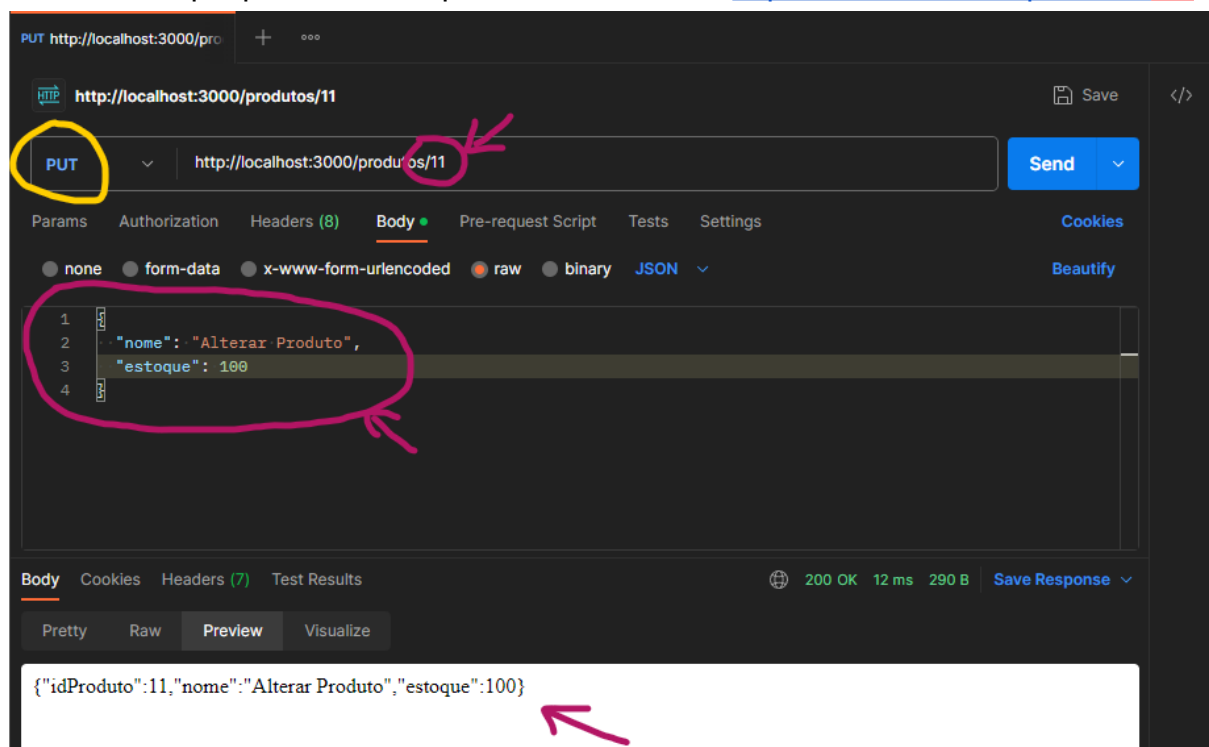
- Method:** POST
- URL:** http://localhost:3000/produtos
- Body (Request):** A JSON object with two fields: "nome": "Novo Produto," and "estoque": 50. This section is circled in pink.
- Status:** 200 OK, 34 ms, 251 B
- Body (Response):** A JSON object with one field: {"idProduto": 11}. A pink arrow points to this response.

7. O método **“PUT”** é para atualizar dados, podemos fazer isso para usuários e produtos, é preciso notar que é necessário indicar o ID do usuário ou produto no fim da url digitada, depois digitar no corpo o JSON o novo nome e senha ou produto e estoque, e assim poderá ver a alteração feita em preview.

Exemplo para alterar o usuário com ID 11: <http://localhost:3000/usuarios/11>

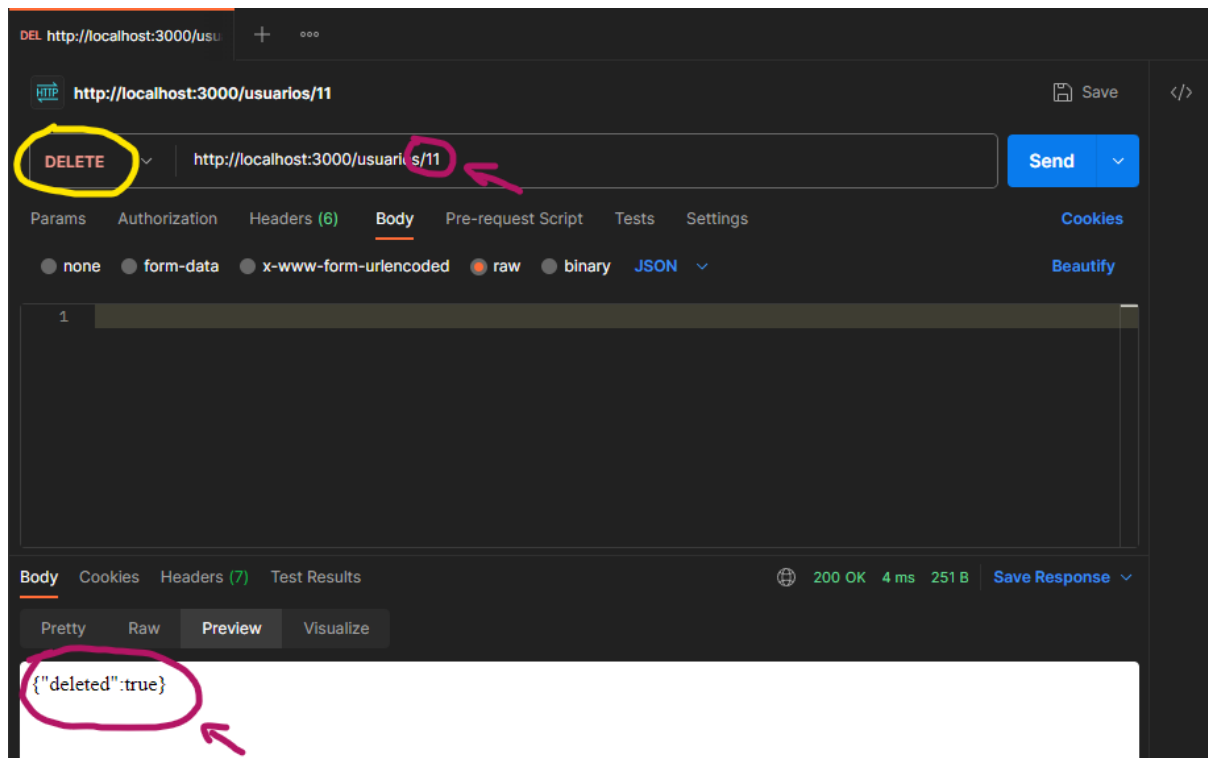


Exemplo para alterar o produto com ID 11: <http://localhost:3000/produtos/11>



8. Por fim, “DELETE” para deletar algo do banco de dados, apenas escolher a rota de usuário e produto com o ID no final da URL. Em preview é possível ver que a ação foi validada como “verdadeira”.

Exemplo para deletar o usuário com ID 11: “<http://localhost:3000/usuarios/11>”



Exemplo para deletar o produto com ID 11: “<http://localhost:3000/produtos/11>”

