

Início da aula

Baixar o VSCode

Baixar GitBash

Baixar o Node.js

Explicar o que é uma API

<https://www.youtube.com/watch?v=s7wmiS2mSXY>

Formatos em que os dados são enviados

Padronização da comunicação

A partir do momento que a API permite a interoperabilidade entre usuários e aplicações, é fundamental ficar atento a importância da padronização da comunicação, a fim de que fique fácil a representação e compreensão por humanos e máquinas. A seguir alguns padrões de troca de dados:

Representação XML

```
<endereco>
  <rua>
    Rua Super
  </rua>
  <cidade>
    Super Geeks City
  </cidade>
</endereco>
```

Representação JSON

```
{ endereco:
  {
    rua: Rua Super,
    cidade: Super Geeks City
  }
}
```

Representação YAML

```
endereco:
  rua: rua Super
  cidade: Super Geeks City
```

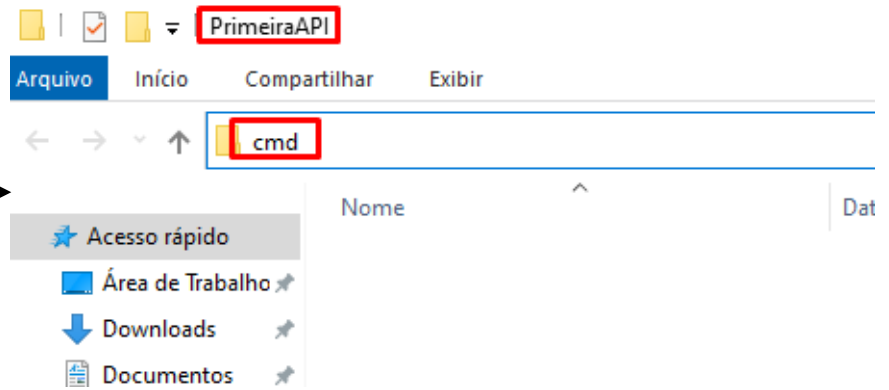
API REST, o que é e tipos e endPoints

GET, POST, PUT, DELETE

Explicar a estrutura de uma URI

Criar uma pasta chamada "PrimeiraAPI"

Dentro da pasta "PrimeiraAPI", digitar cmd para abrir o prompt



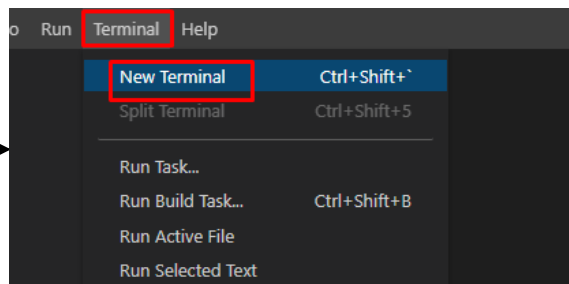
Digitar o comando para iniciar o projeto, dar enter em tudo

```
>npm init  
package.json
```

Entrar dentro da pasta pelo terminal e digitar "code ." -> Para abrir o VSCode

```
>code .
```

Abrir um terminal dentro do VSCode



No terminal do VSCode colocar o comando -> npm i express

```
> npm i express
```

Para instalar um conjunto de bibliotecas no projeto para ajudar na construção da aplicação

Colocar um arquivo index.js no projeto

```
const express = require('express')  
const server = express()  
  
server.listen(3000, () => {  
  console.log('Servidor está funcionando...')  
})  
  
server.get('/', (req, res) => {  
  return res.json({mensagem: 'Nossa API está funcionando'})  
})  
  
server.get('/usuario', (req, res) => {  
  return res.json({nome: 'Arthur', idade: 23})  
})
```

no terminal rodar o comando -> node index.js

```
PS D:\Vinicius\Cursos\SuperGeeks\Experts\SegundaAPI> node index.js  
Servidor está funcionando...
```

entrar no navegador com o localhost:3000/ e localhost:3000/usuario