**Tutorial Flutter: Integração com APIs**

* **Recursos necessários:**
  + - Ambiente para desenvolvimento Flutter devidamente configurado
    - Node.JS para criação da API
    - Visual Studio Code (Extensões Flutter)
* **Sinopse:**

Neste tutorial iremos realizar a criação de uma API REST própria e integração com Aplicação Flutter.

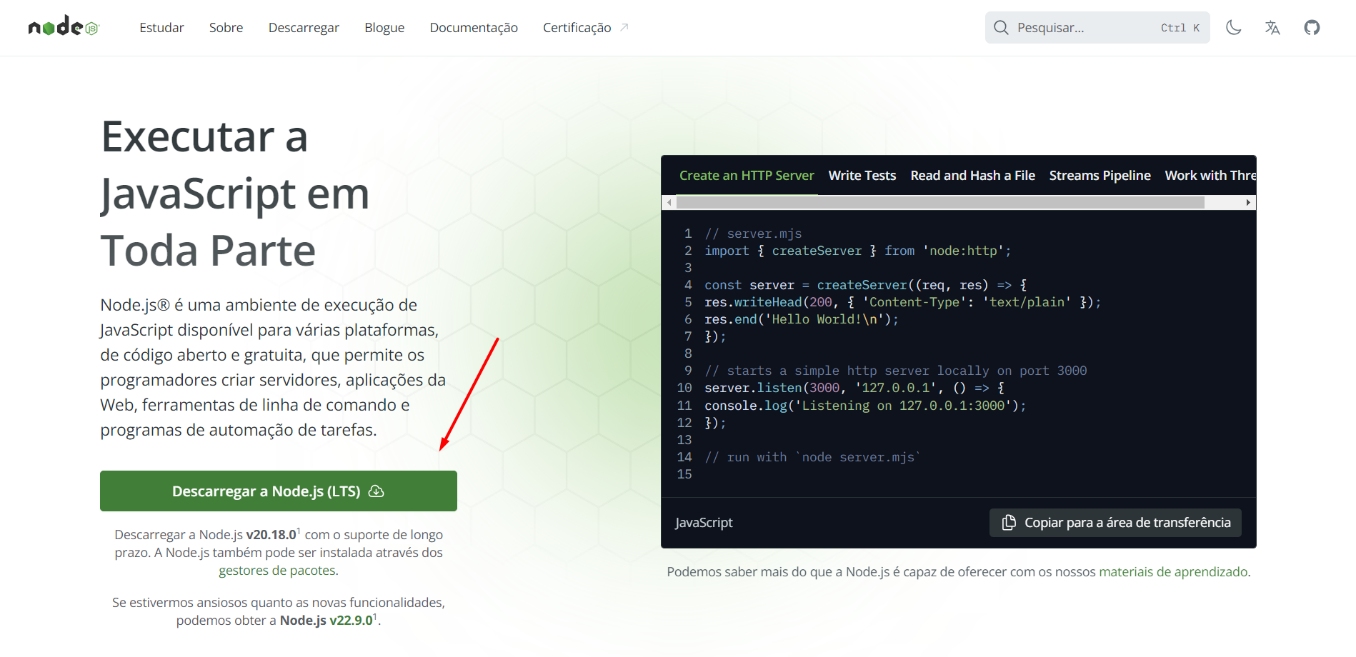
* **1. O que são APIs ?**

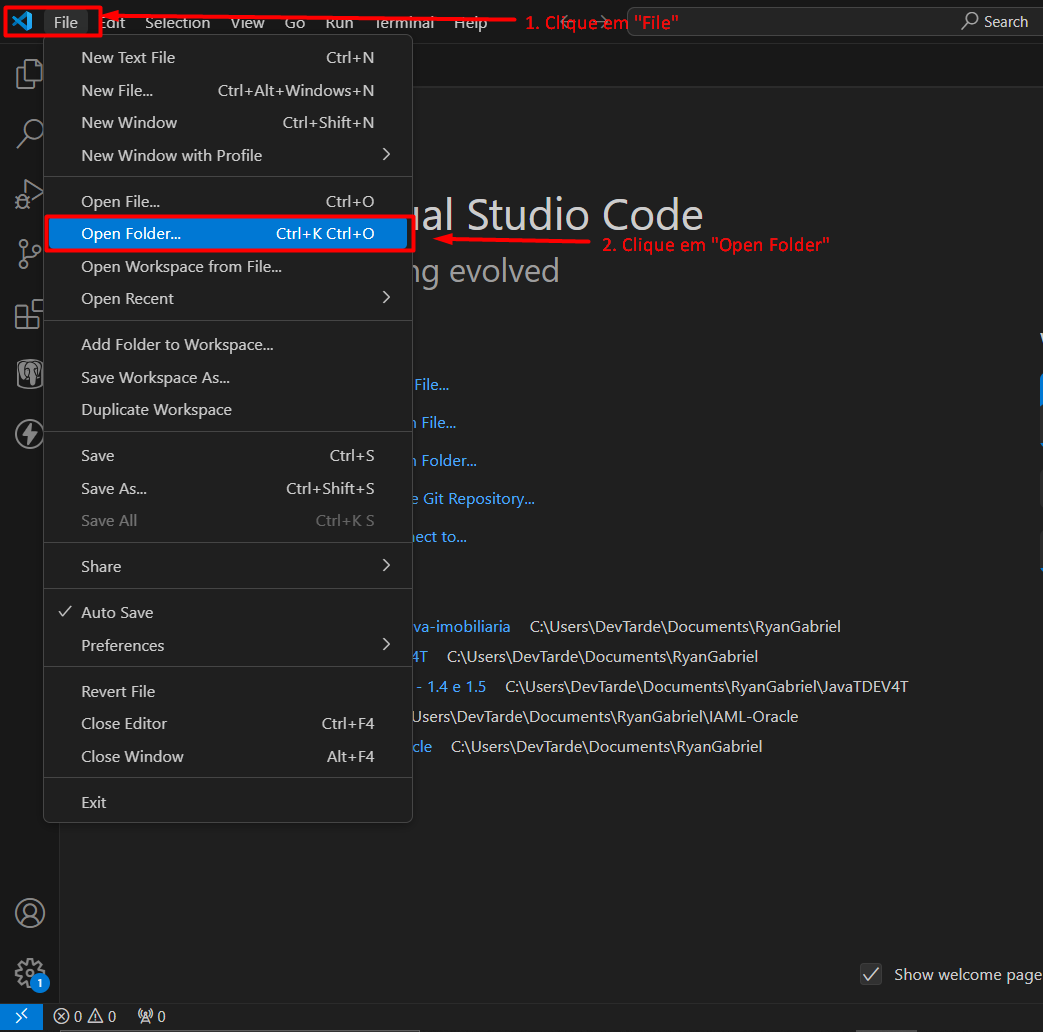
APIs (Application Programming Interfaces) são como pontes que permitem que diferentes sistemas, aplicativos ou serviços se comuniquem entre si. Imagine que você está em um restaurante: o garçom é a "API". Você (o cliente) não precisa ir até a cozinha para preparar a comida; você só faz o pedido ao garçom, e ele leva sua solicitação para a cozinha (o sistema) e, depois, traz de volta a comida pronta (a resposta).

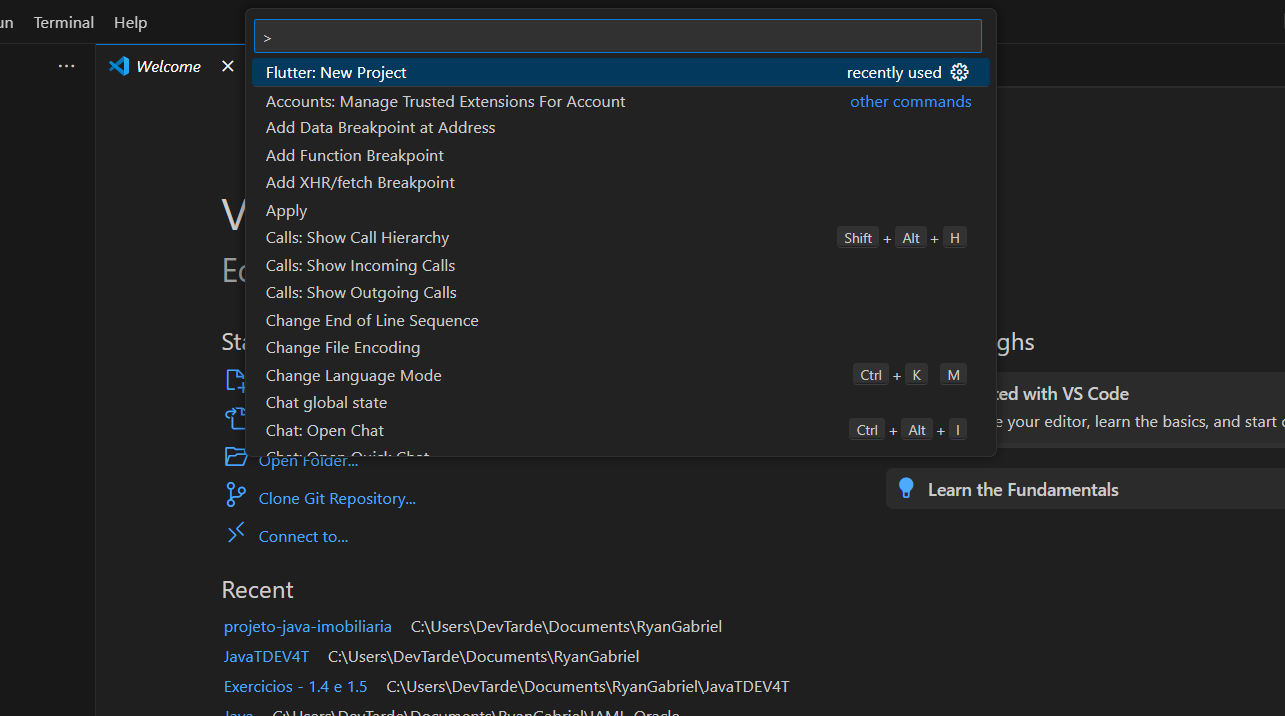
Quando você usa um app, ele pode precisar buscar informações de outro lugar, como dados de previsão do tempo ou os posts mais recentes do Instagram. Em vez de o app fazer tudo por conta própria, ele faz um "pedido" para a API, que se comunica com o servidor e traz a resposta pronta.

As APIs estão em, praticamente,

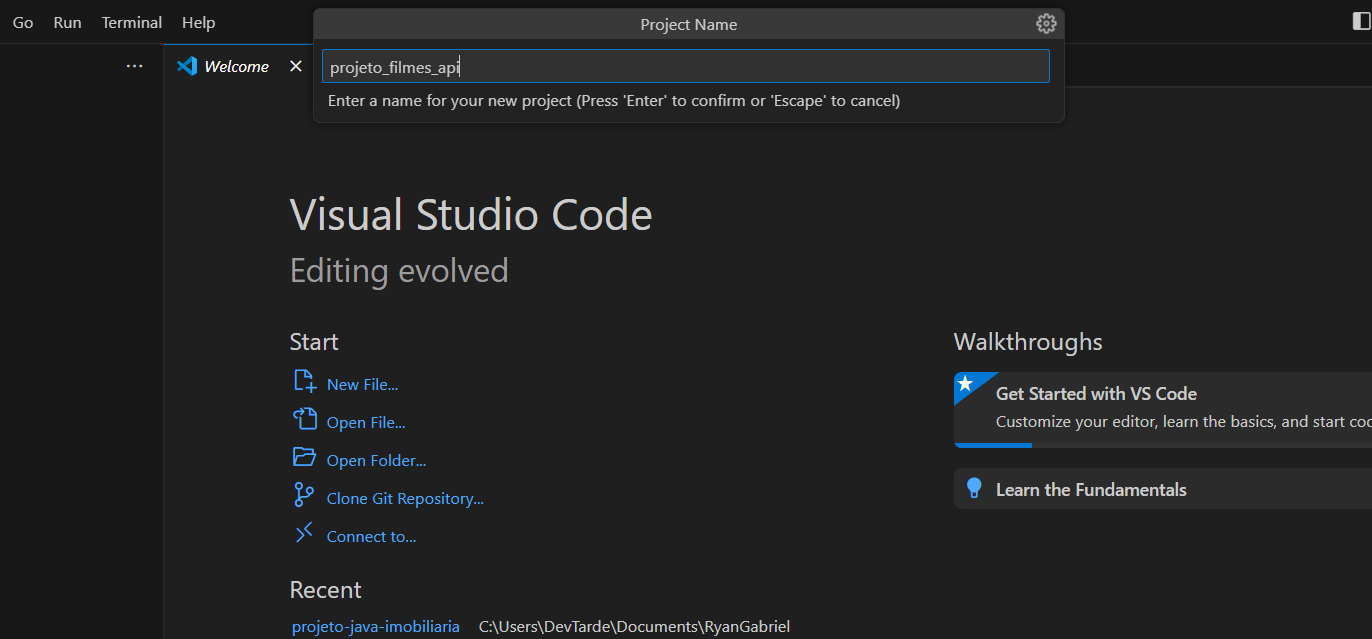
* + **Benefícios das APIs:**
    - **Facilidade de Integração:**  
      APIs facilitam a conexão entre diferentes sistemas e serviços, permitindo que eles troquem dados sem complicação. Você pode usar APIs para conectar seu app a serviços como pagamentos ou dados externos, sem precisar construir tudo do zero.
    - **Reutilização de Funcionalidades:**  
      Com uma API, você pode usar funções já prontas e testadas por outras empresas ou desenvolvedores. Um exemplo clássico é usar a API do Google Maps para mostrar mapas no seu app.
    - **Escalabilidade:**  
      APIs ajudam seu sistema a crescer sem perder eficiência. Elas permitem que partes diferentes do sistema trabalhem juntas de forma eficiente, tornando mais fácil adicionar novas funcionalidades ou suportar mais usuários.
    - **Segurança:**  
      APIs oferecem controle sobre quem pode acessar seus dados, exigindo autenticação e verificações, o que garante que apenas pessoas autorizadas possam usar certos recursos.
    - **Padronização:**  
      As APIs seguem regras e formatos bem definidos, como REST ou GraphQL, facilitando o uso e a integração entre desenvolvedores e sistemas, pois todos sabem como as coisas devem funcionar.
    - **Economia de Tempo e Recursos:**  
      Usar APIs economiza tempo e dinheiro, porque você aproveita infraestrutura e funcionalidades já existentes em vez de gastar para criar tudo do zero.
    - **Manutenção Fácil:**  
      APIs organizam e isolam as funcionalidades, o que facilita a atualização e manutenção do código sem quebrar outras partes do sistema.
* 2. Criando uma API:
  + 2.1 – Instalando o NODE:
    - Link: <https://nodejs.org/pt> ;
    - Clique no botão principal “ Descarregar a Node.js (LTS) “;

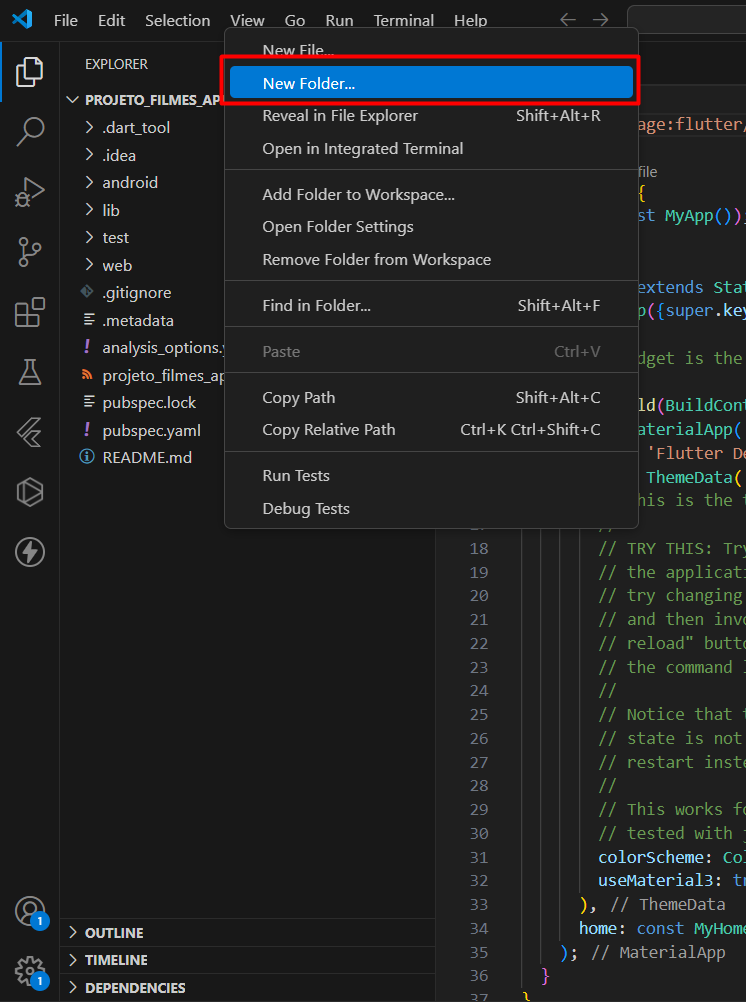


* Após baixar o instalador, execute-o e apenas prossiga com a instalação;
* **2.2 – Criando um projeto Flutter no VSCode:**
  + Abra uma nova pasta no seu VSCode;
  + Selecione o seu diretório preferido e abra, assim que aberto pressione sucessivamente “CTRL+SHIFT+P”, e procure pela opção: ‘ Flutter: New Project ‘

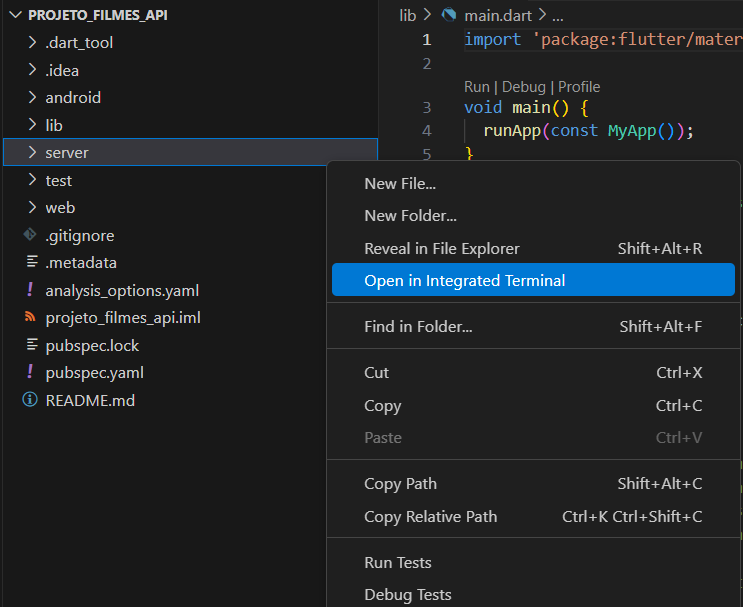


* + Pressione ‘ENTER’ e depois escolha a opção: ‘Application’;
  + Selecione o mesmo diretório que você está e depois coloque o nome do projeto com projeto\_filmes\_api;



* + Pressione ‘ENTER’ e aguarde o projeto ser criado;
  + Assim que criado crie uma nova pasta na raiz do projeto chamada ‘server’;

* + Clique com o botão direito na pasta server e abra um terminal integrado a ela:



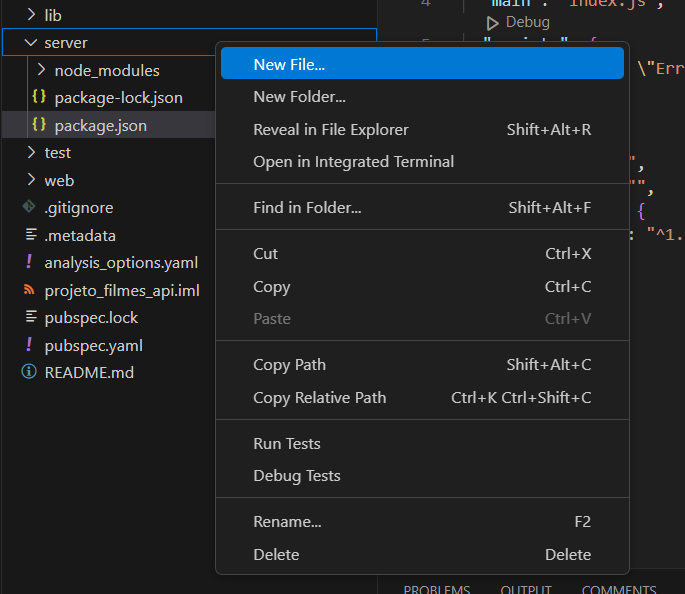
* + No terminal digite:

npm init -y

* + Aperte ENTER e aguarde um instante, depois digite novamente no terminal:

npm install json-server

* + Pressione ENTER novamente e aguarde a instalação..
  + Depois de instalado verifique se foi criado 2 arquivos JSON e um diretório (node\_modules);
  + Agora, dentro da pasta ‘server’, crie o arquivo “db.json”:



* + Dentro do arquivo “db.json” escreva a seguinte coleção:

{

    "filmes": [

      {

        "id": "cd68",

        "nome": "Os Vingadores",

        "categoria": "Acao",

        "duracao": 143,

        "ano": 2012,

        "classificacao": 9

      },

      {

        "id": "bbec",

        "nome": "Harry Potter e a Pedra Filosofal",

        "categoria": "Fantasia",

        "duracao": 152,

        "ano": 2001,

        "classificacao": 10

      },

      {

        "id": "ce65",

        "nome": "Esposa de Mentirinha",

        "categoria": "Comedia/Romance",

        "duracao": 117,

        "ano": 2011,

        "classificacao": 12

      },

      {

        "id": "b0e7",

        "nome": "Gente Grande 2",

        "categoria": "Comedia",

        "duracao": 101,

        "ano": 2013,

        "classificacao": 11

      },

      {

        "id": "5c6c",

        "nome": "Bad Boys IV",

        "categoria": "Ação",

        "duracao": 150,

        "ano": 2024,

        "classificacao": 10

      }

    ]

  }

* + Agora voltando ao terminal digite: