SDKs K8Bank Pagamentos

Fevereiro/2021

Índice

1. Objetivo 4

2. Estrutura física 4

3. Utilização das SDKs 4

3.1. Linguagem C# (Microsoft .Net) 4

3.1.1. Pré Requisitos 4

3.1.2. Instalação/Compilação 4

3.1.3. Parametrização 6

3.1.4. Execução 8

3.2. Linguagem Java 9

3.2.1. Pré Requisitos 9

3.2.2. Instalação/Compilação 9

3.2.3. Parametrização 12

3.2.4. Execução 12

3.3. Linguagem PHP 15

3.3.1. Pré Requisitos 15

3.3.2. Instalação/Compilação 15

3.3.3. Parametrização 17

3.3.4. Execução 18

3.4. Linguagem Javascript/Typescript (Node.js) 20

3.4.1. Pré Requisitos 20

3.4.2. Instalação/Compilação 20

3.4.3. Parametrização 22

3.4.4. Execução 23

3.5. Linguagem Python 25

3.5.1. Pré Requisitos 25

3.5.2. Instalação/Compilação 25

3.5.3. Parametrização 27

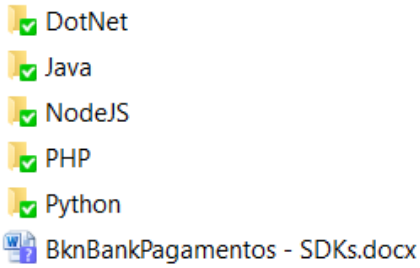
3.5.4. Execução 28

1. Objetivo

O objetivo deste documento é detalhar o uso das SDKs para integração com o sistema K8Bank Pagamentos em cinco linguagens de programação: C# (.NET), Java, PHP, Typescript/Javascript (Node.JS) e Python.

2. Estrutura física

As pastas da SDK são:



As 5 pastas representam cada uma das linguagens de programação. Ao adicional, está agregado na pasta raiz este documento descrevendo a instalação e uso de cada uma das SDKs.

3. Utilização das SDKs

3.1. Linguagem C# (Microsoft .Net)

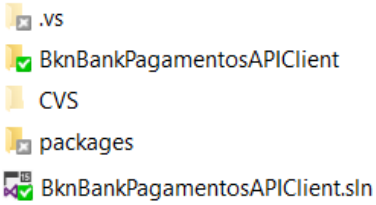
3.1.1. Pré Requisitos

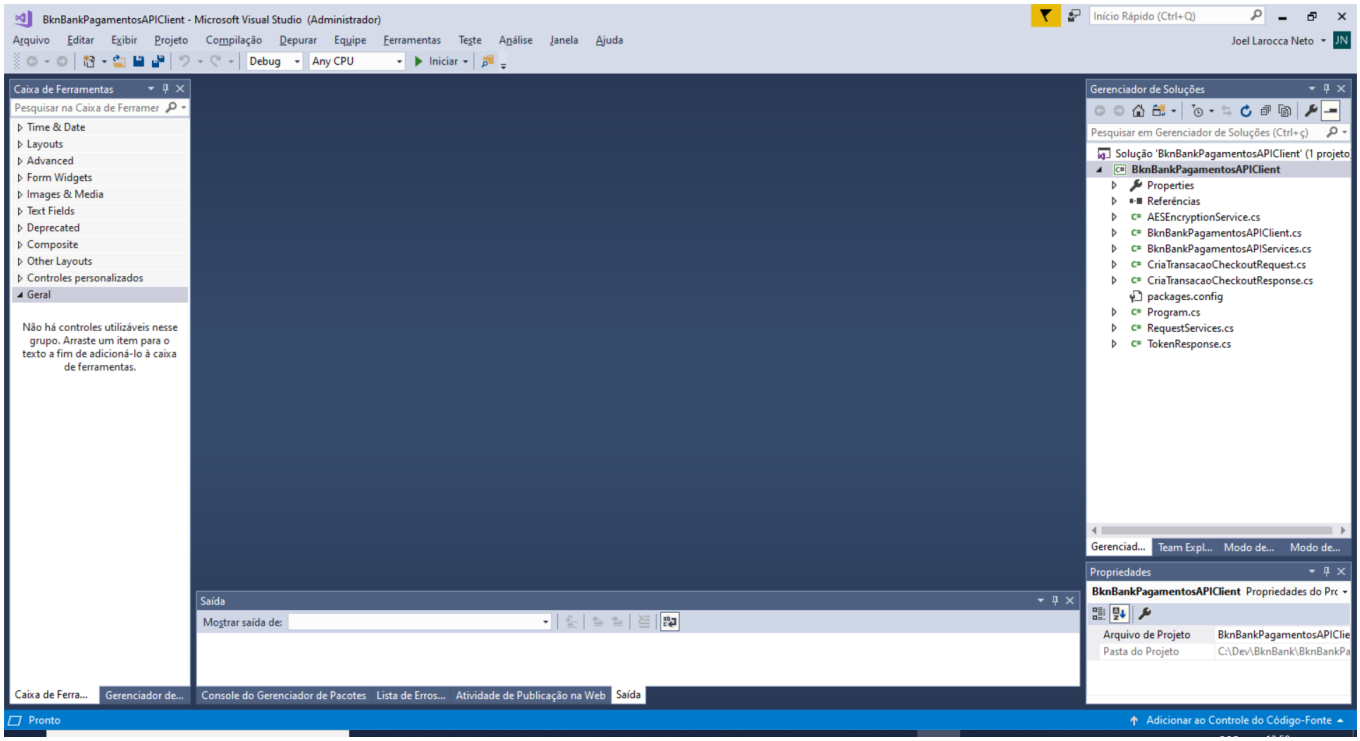
Para uso da SDK .NET, é necessário possuir o seguinte software instalado:

- Microsoft Visual Studio 2017/2019 Community Edition ou superior.

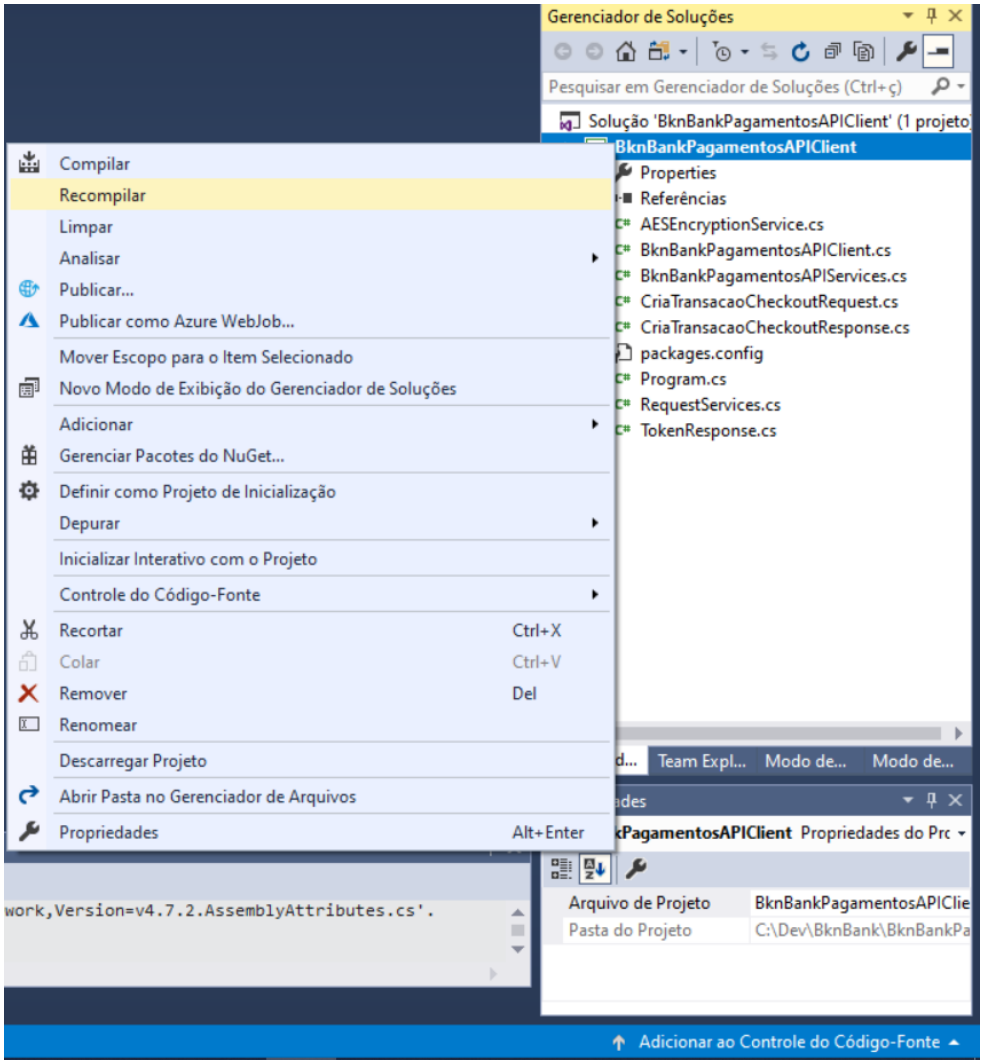
3.1.2. Instalação/Compilação

Para compilar a SDK en .NET, abrir o arquivo BknBankPagamentosAPIClient.sln na pasta DotNet com o Visual Studio:

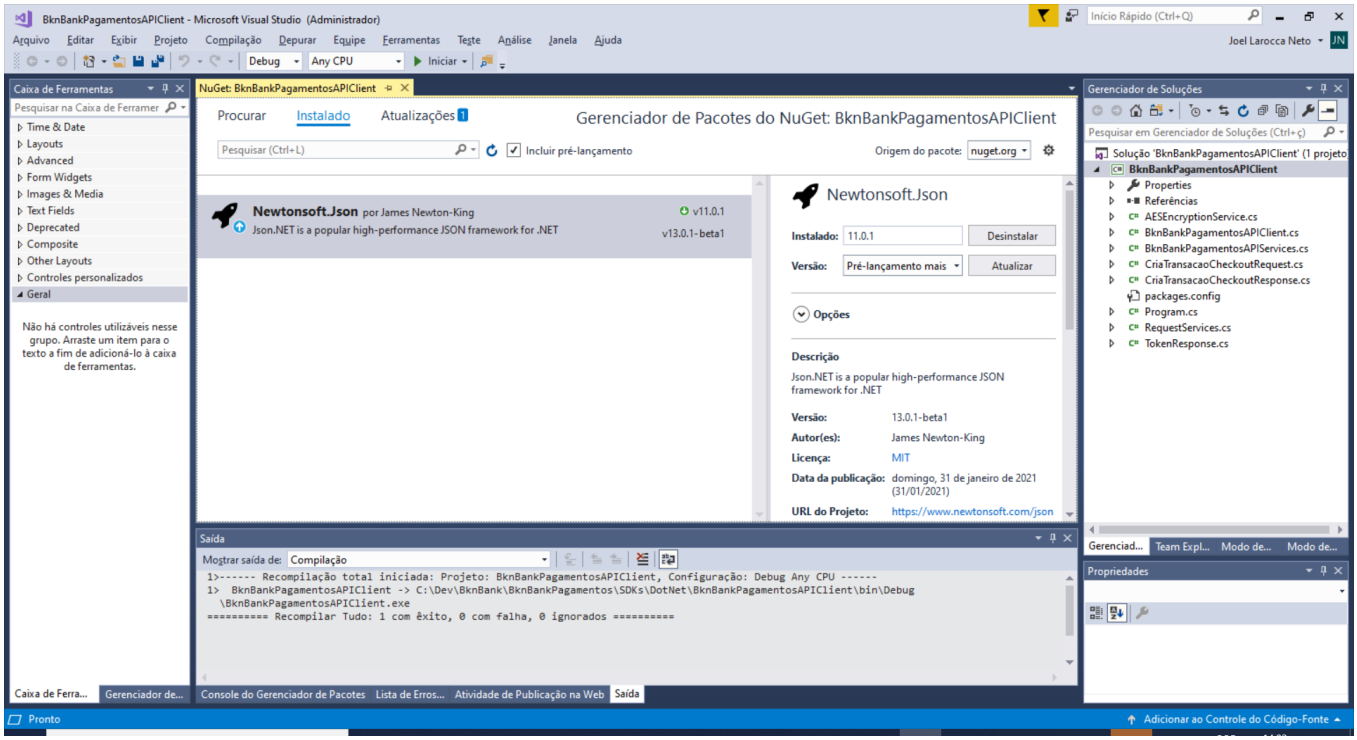




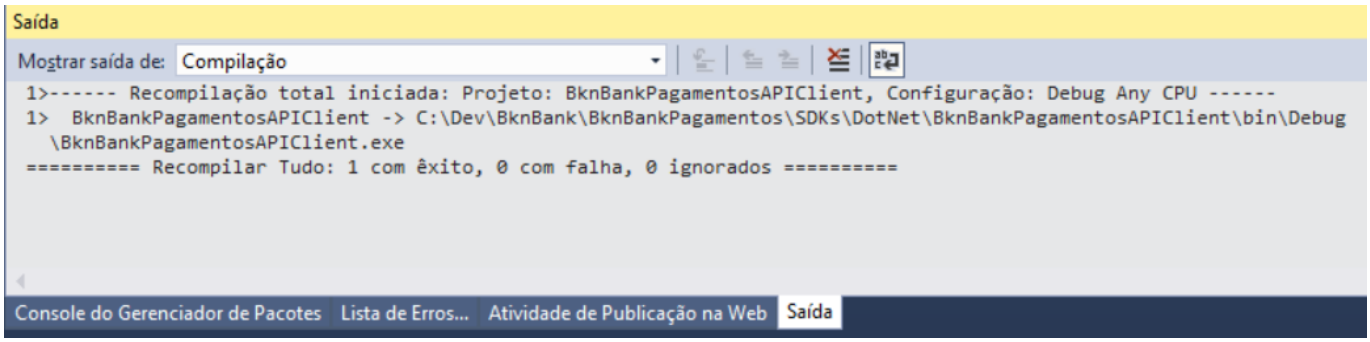
Para compilar o projeto, clicar sobre o nome da solução e selecionar a opção ”Recompilar”:



Notar a dependência do pacote Newtonsoft.JSON:



Na janela de Saída, dever ser indicado que foi compilado sem erros:



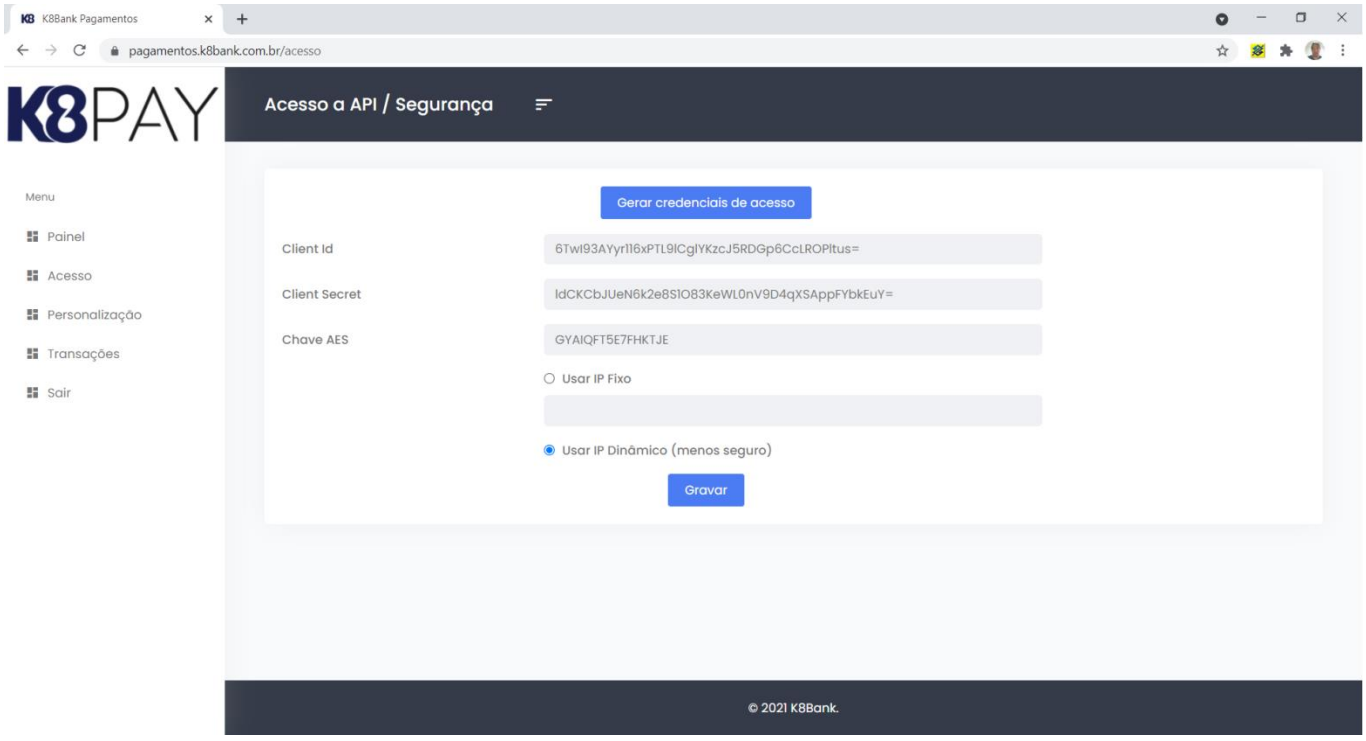
3.1.3. Parametrização

Na classe Program.cs, notar os valores dos parâmetros URL, ClientId, ClientSecret e AESKey:

```
Program.cs
BknBankPagamentosAPIClient | BknBankPagamentosAPIClient.Program | URL
```

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using Newtonsoft.Json;

namespace BknBankPagamentosAPIClient
{
    static class Program
    {
        const string URL = "https://localhost/BknBankPagamentosAPI";
        const string ClientId = "AUuc3ehyAhlE/vHNqMaBX14wT+LyUt+NlGvan9f4990=";
        const string ClientSecret = "XJLj3M43AyFvXiZUrxHkVRN/0/tV2vGpIVhZ7xf4G/M=";
        const string AesKey = "VA68BUEI0H3BJ6J7";

        /// <summary>
        /// Ponto de entrada principal para o aplicativo.
        /// </summary>
        static void Main()
        {
            Console.WriteLine("Exemplo de Integração .NET - BknBank Pagamentos\r\n");

            BknBankPagamentosAPIClient sdk = new BknBankPagamentosAPIClient(URL,ClientId,ClientSecret,AesKey);
            sdk.SetTimeout(60000);
            sdk.SetCertificateIgnoreInvalid(true);

            CriaTransacaoCheckoutRequest request = new CriaTransacaoCheckoutRequest();
            request.Valor = 100;
            request.BoletoDataVencimento = DateTime.Now.Date.AddDays(5);
            request.ClienteCPFCNPJ = "11111111111";
            request.ClienteNome = "Cliente Teste";
            request.ClienteIP = "127.0.0.1";
            request.ClienteDescricao = "Compra de Teste";
            request.ClienteEmail = "bknbankpagamentos@jlnsoftware.com.br";
            request.PixIDTransacao = 123;
```

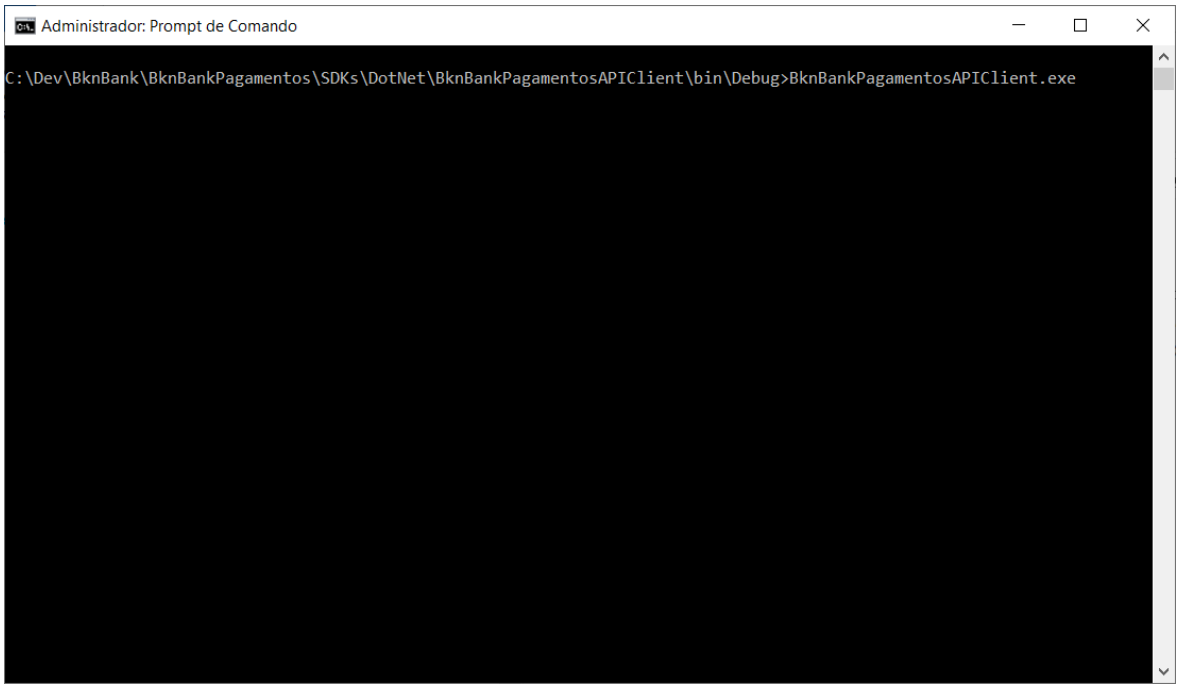
Para obter os valores a serem utilizados, logar na interface do K8Bank Pagamentos e selecionar a opção de menu Acesso:



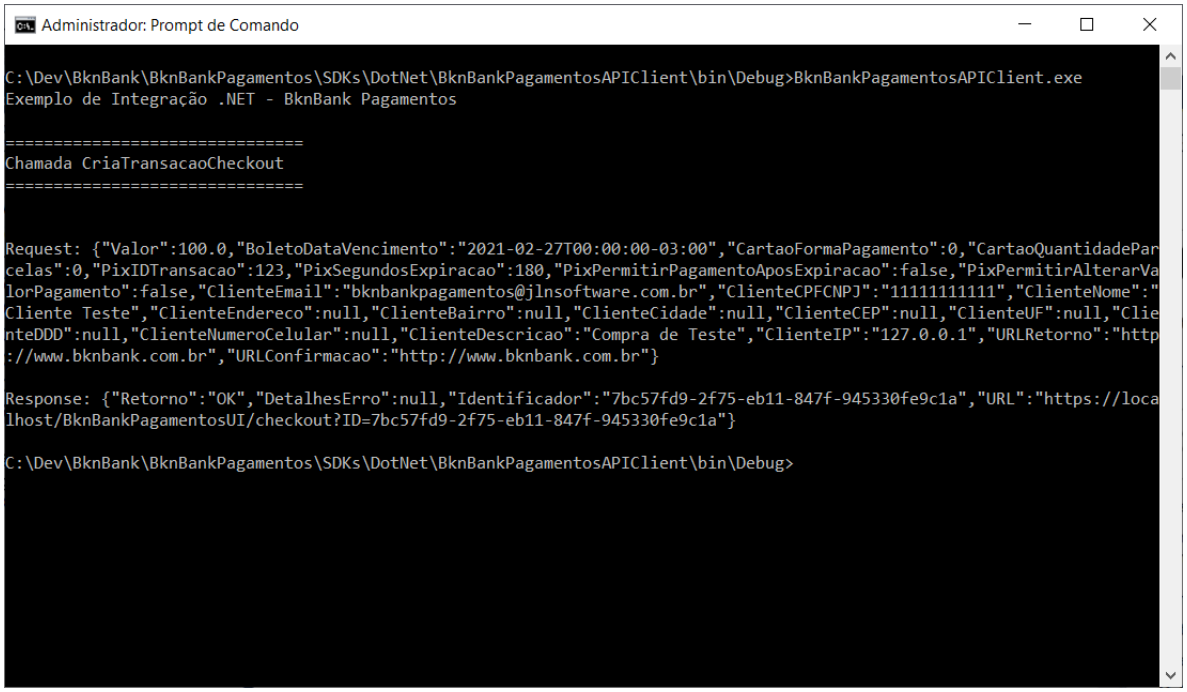
Notar os parâmetros ClientId, Client Secret e Chave AES. Notar ainda que é possível especificar o uso de um IP fixo ou de IP dinâmico. Ajustar ainda a URL de produção do sistema.

3.1.4. Execução

Para executar a SDK .NET, invocar o programa executável do projeto (em modo console):



Notar a saída, invocando o método CriaTransacaoCheckout, enviando os parâmetros e obtendo a respectiva resposta:



3.2. Linguagem Java

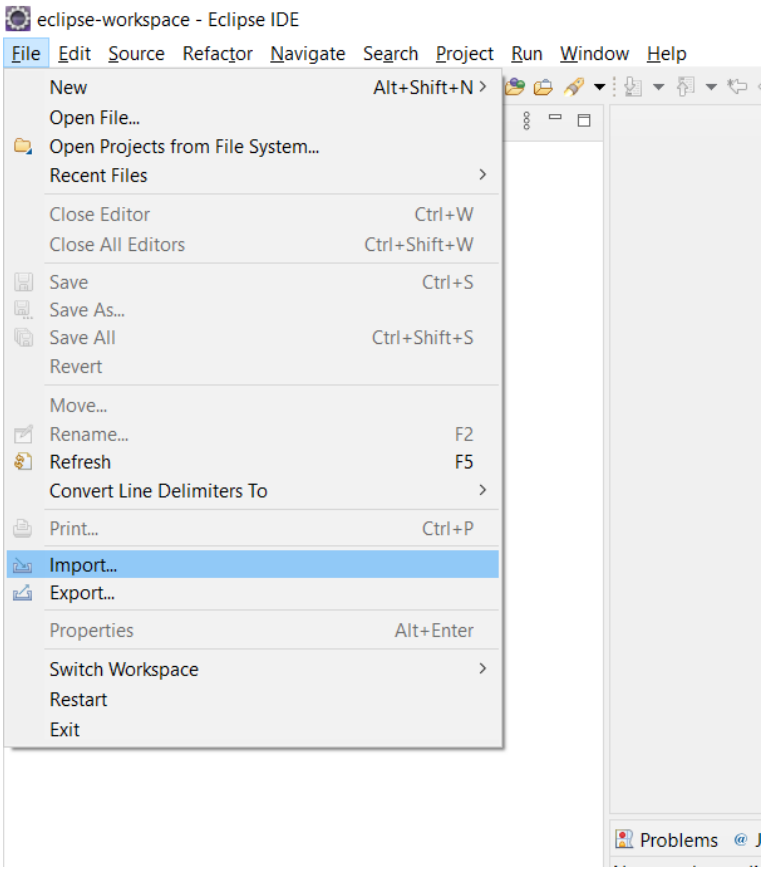
3.2.1. Pré Requisitos

Para utilizar a SDK Java, é necessário possuir os seguintes softwares instalados:

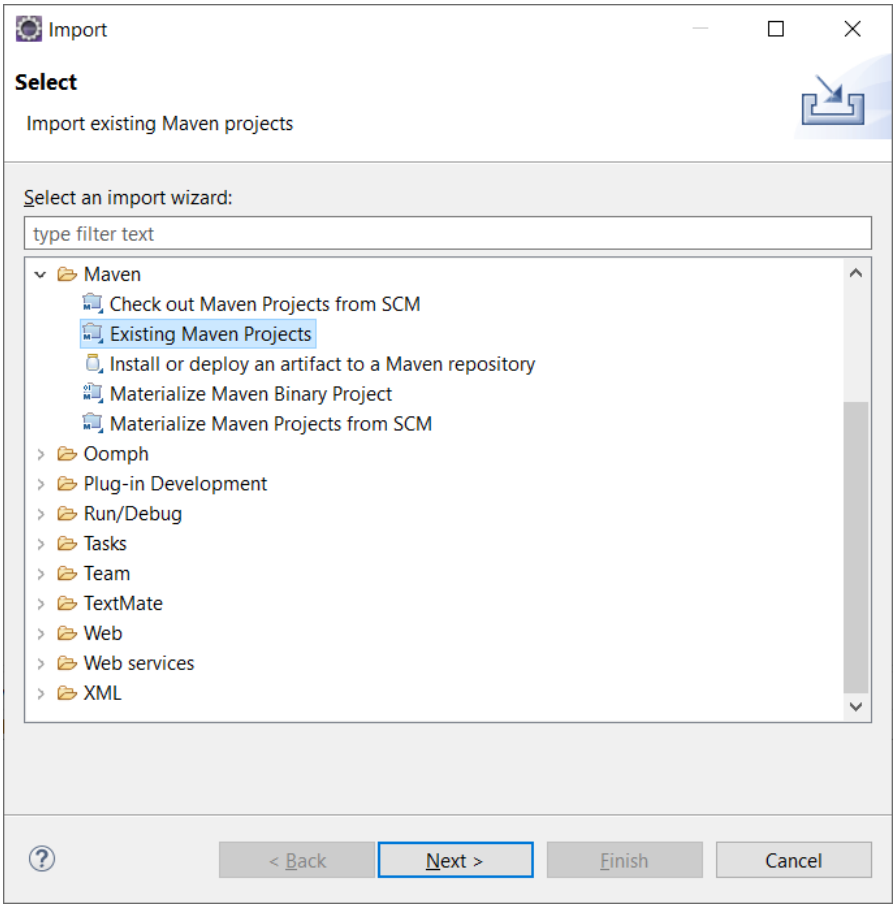
- Eclipse IDE for Enterprise Java Developers
    - https://www.eclipse.org/downloads/packages/
- Java SE Development Kit 8
    - https://www.oracle.com/br/java/technologies/javase/javase-jdk8-downloads.html

3.2.2. Instalação/Compilação

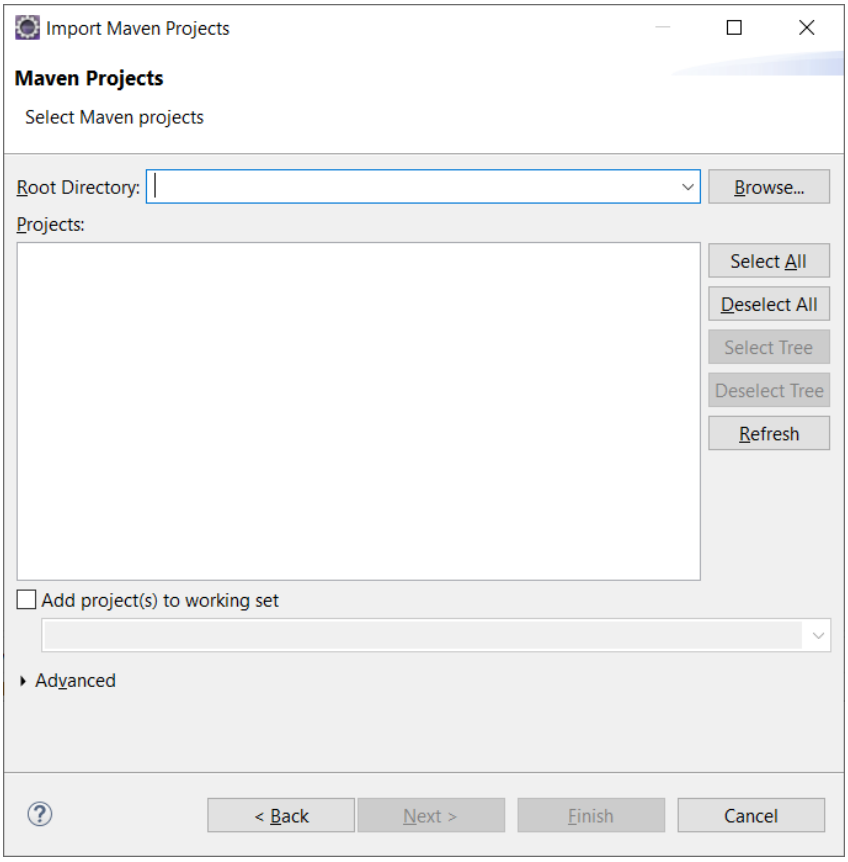
Para utilizar a SDK Java, abrir o programa Eclipse IDE e importar o projeto, clicando no menu File->Import:



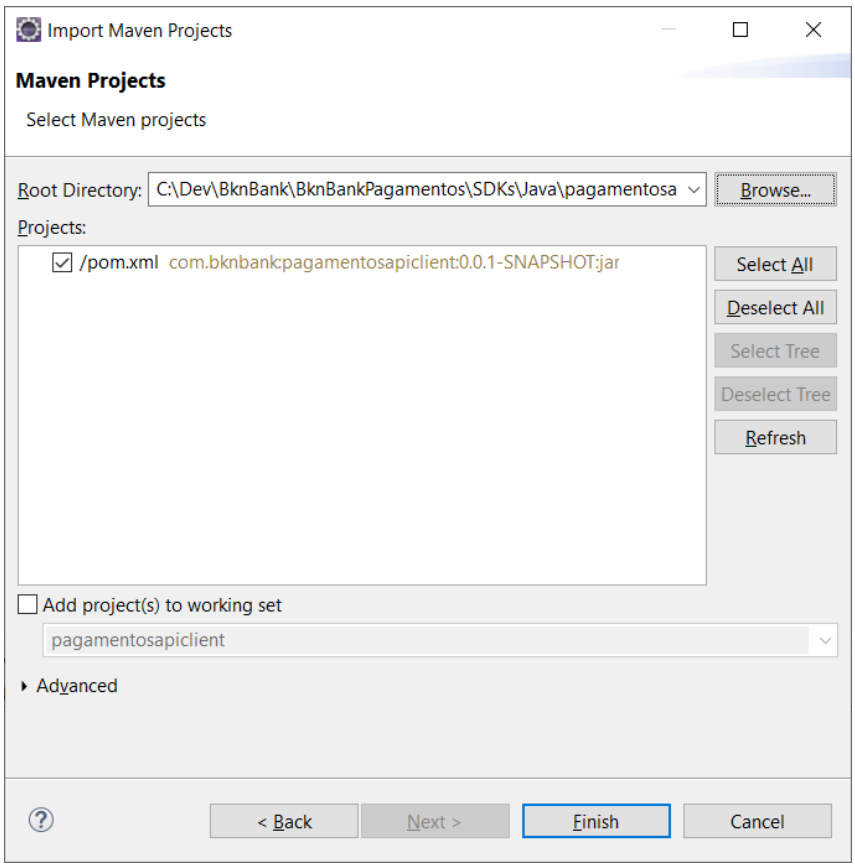
Selecionar a opção “Existing Maven projects”:



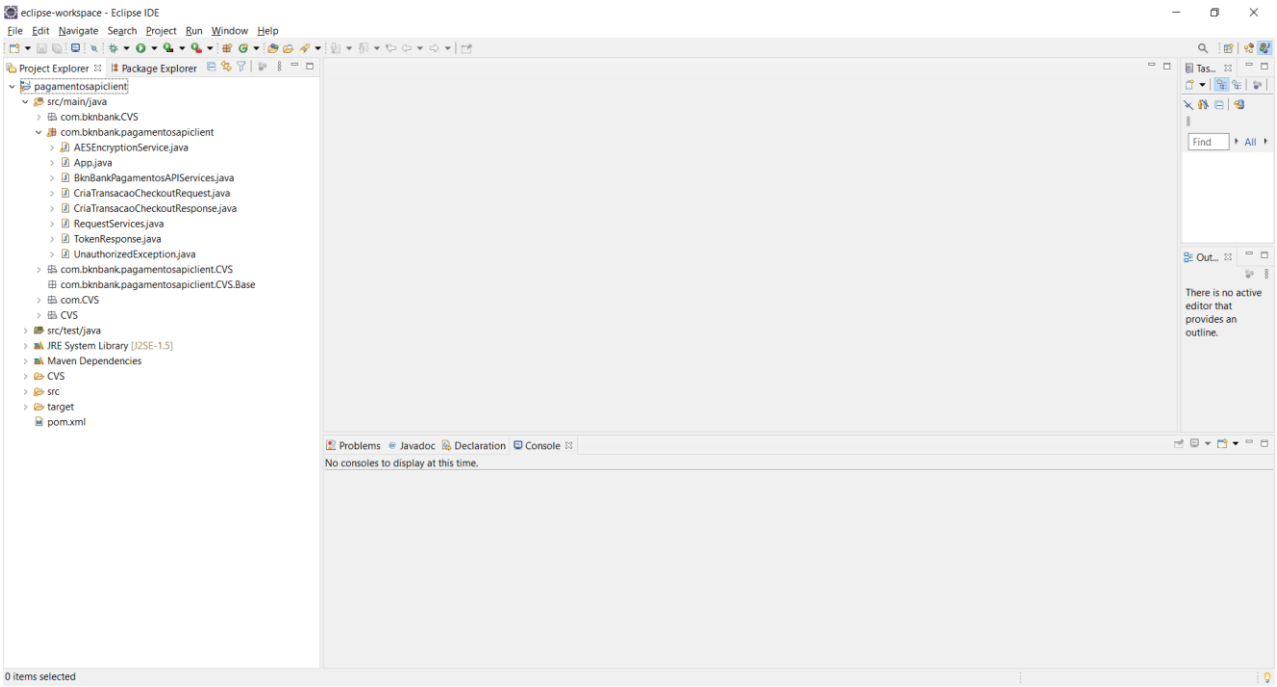
Clicar em Next:



Clicar em “Browse…” e buscar a pasta Java\pagamentosapiclient do SDK:

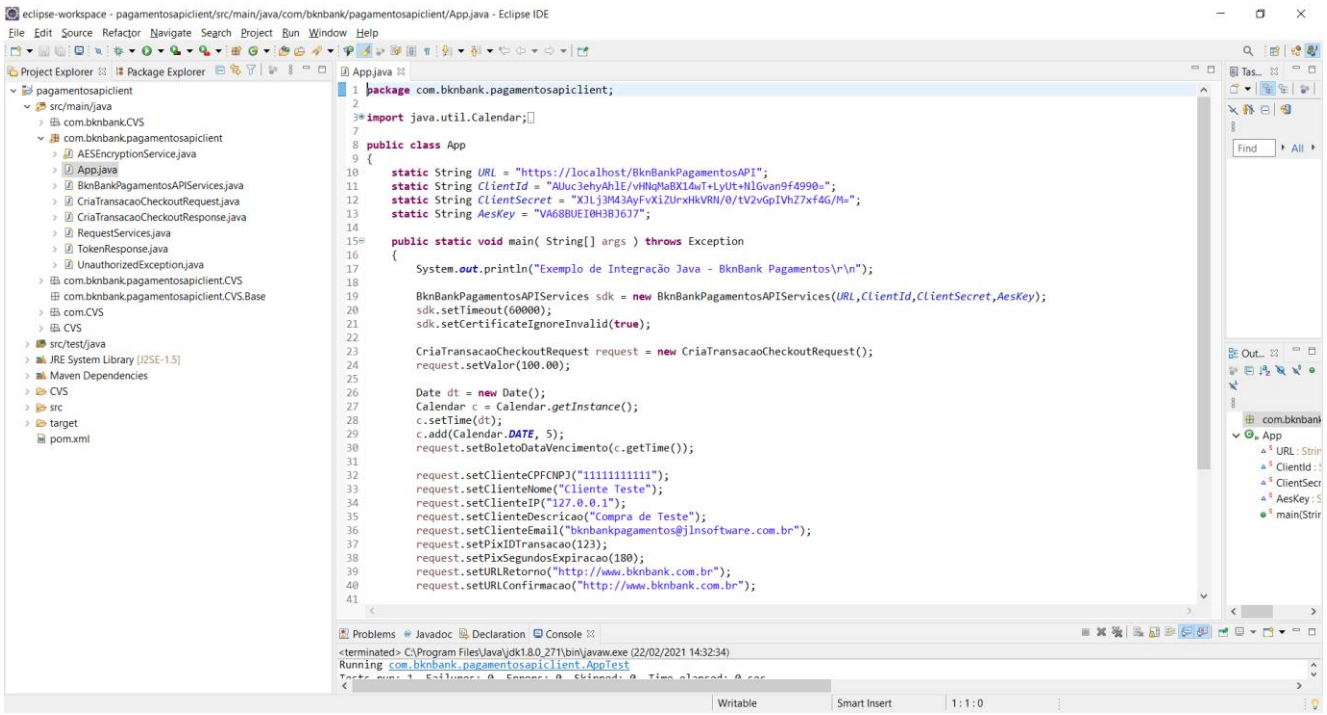


Clicar em “Finish”:

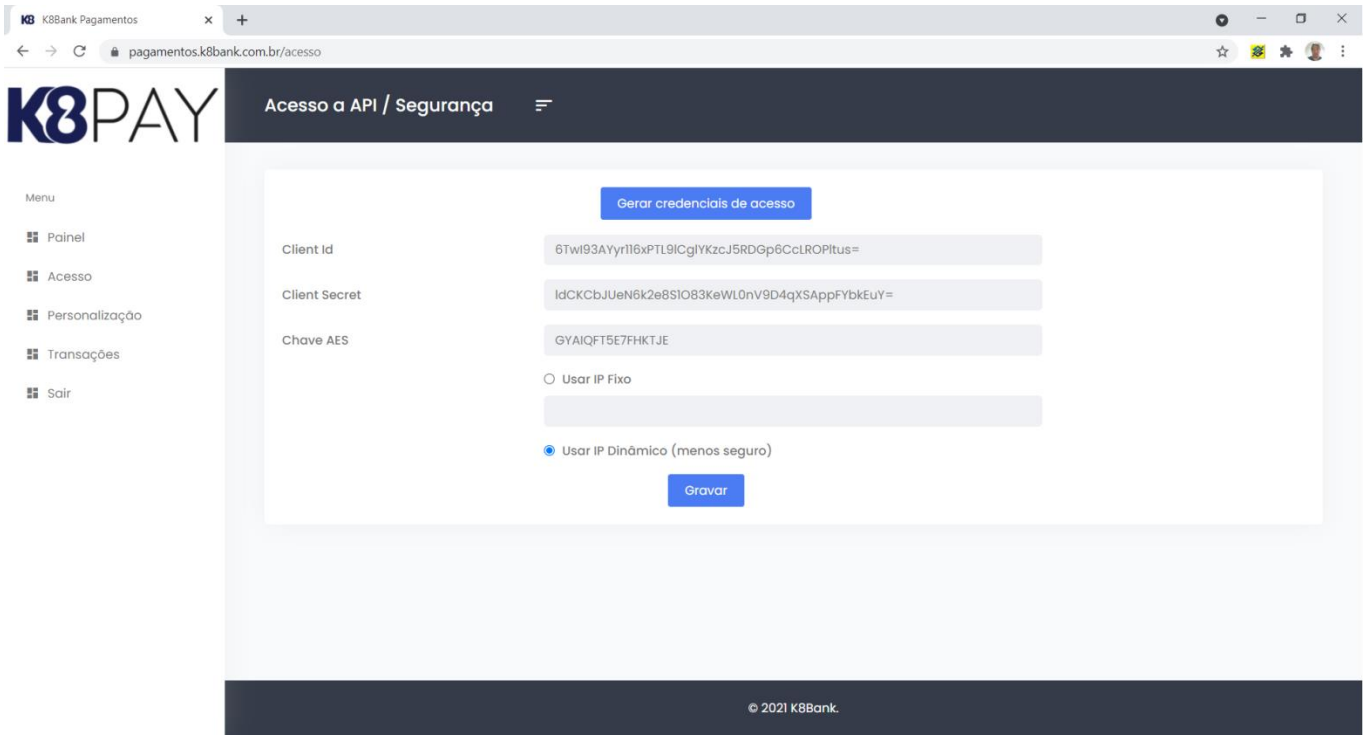


3.2.3. Parametrização

Abrir o código fonte da classe App.java e notar os valores dos parâmetros URL, ClientId, ClientSecret e AESKey:



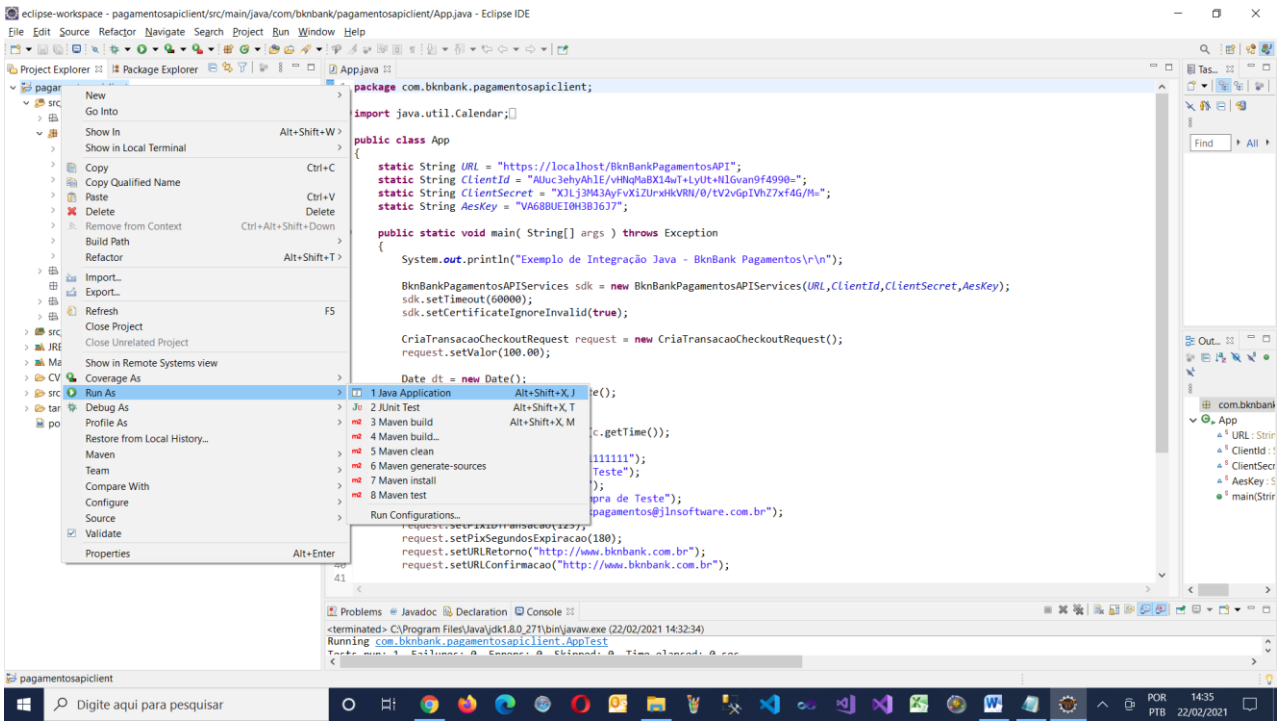
Para obter os valores a serem utilizados, logar na interface do K8Bank Pagamentos e selecionar a opção de menu Acesso:

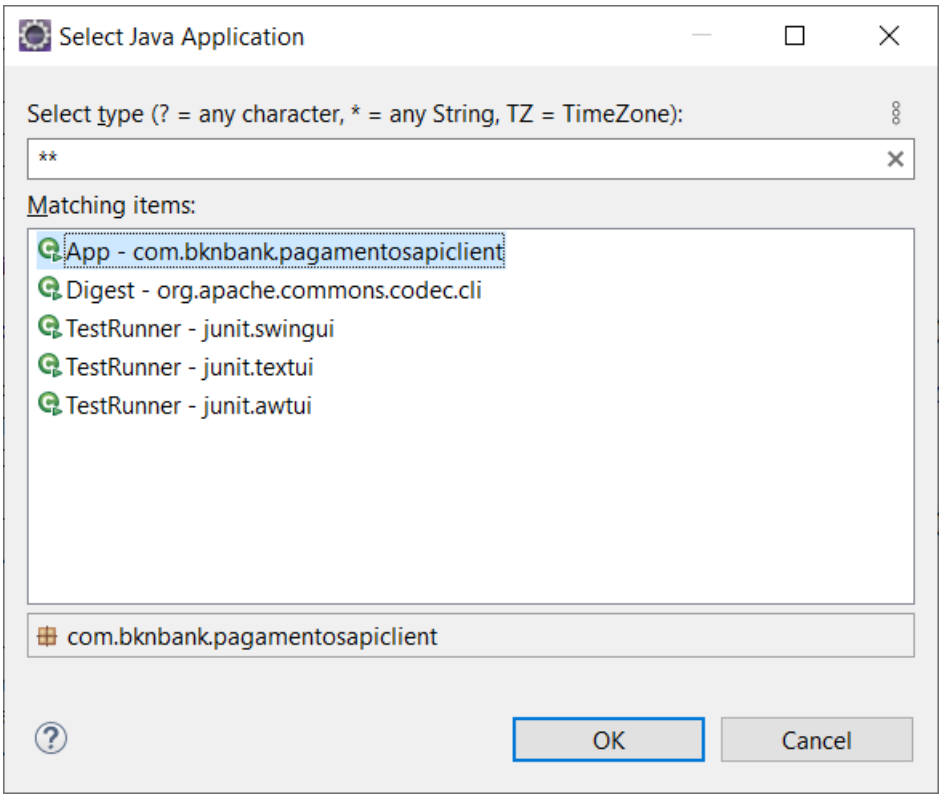


Notar os parâmetros ClientId, Client Secret e Chave AES. Notar ainda que é possível especificar o uso de um IP fixo ou de IP dinâmico. Ajustar ainda a URL de produção do sistema.

3.2.4. Execução

Para executar a SDK Java, clicar sobre o projeto e selecionar a opção “Run As” -> “Java Application”:





Selecionar a classe App e clicar em “OK”:

Na janela “Console”, notar a saída da execução do programa, invocando o método CriaTransacaoCheckout, enviando os parâmetros e obtendo a respectiva resposta.

3.3. Linguagem PHP

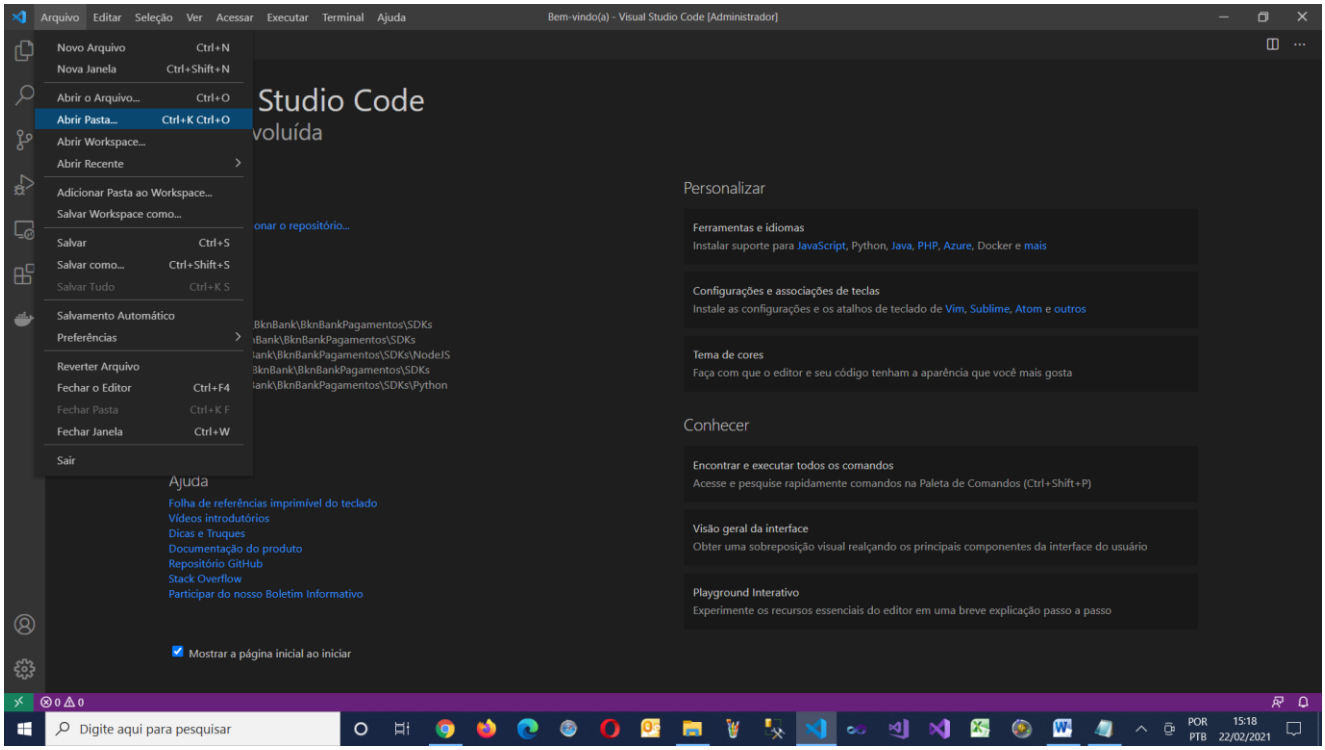
3.3.1. Pré Requisitos

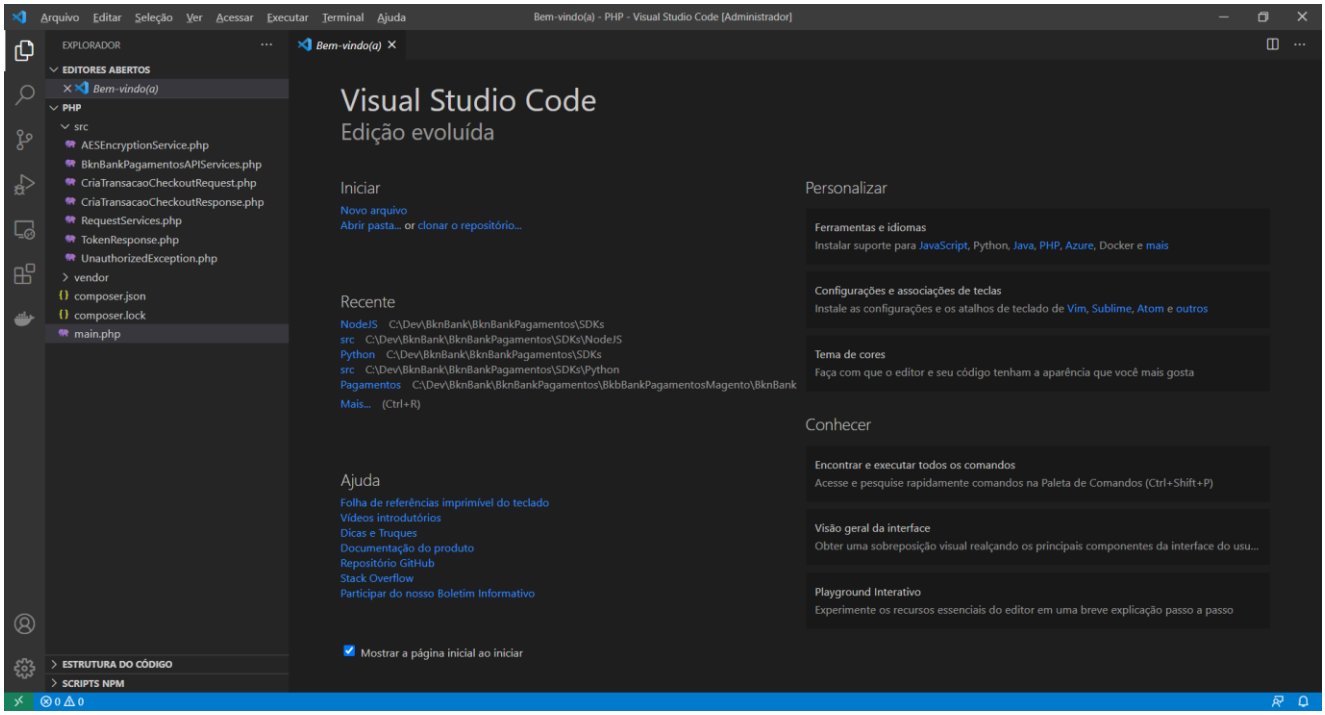
Para utilizar a SDK em PHP, é necessário ter os seguintes softwares instalados:

- PHP 7.2 ou superior
    - https://windows.php.net/download#php-7.2
    - Fazer o download do arquivo e descompactar em uma pasta local da máquina (ex: C:\php-7.2.33-1-nts-Win32-VC15-x64\). Lembrar de adicionar este diretório na variável Path do sistema, para executar o comando “php” na linha de comando.
- PHP Composer (gerenciador de pacotes PHP)
    - https://getcomposer.org/download/
    - Fazer o download do instalador para Windows e seguir as instruções, indicando a pasta aonde o PHP foi instalado.
- Visual Studio Code
    - https://code.visualstudio.com/download
    - Efetuar o download do instalador e realizar a instalação.

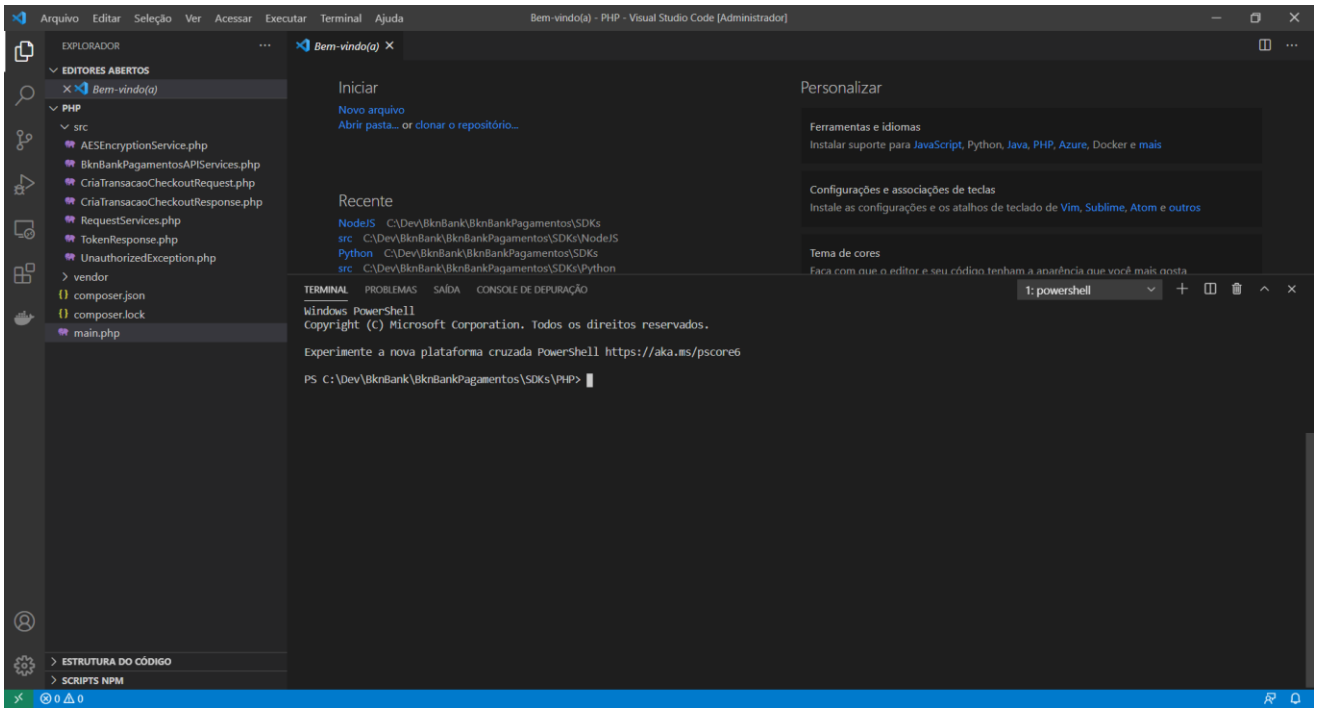
3.3.2. Instalação/Compilação

No programa Visual Studio Code, clicar em Arquivo , Abrir Pasta…, e abrir a pasta PHP abaixo do SDK:





Instalar os pacotes do PHP Composer, abrindo um terminal (menu Ver->Terminal do Visual Studio Code):



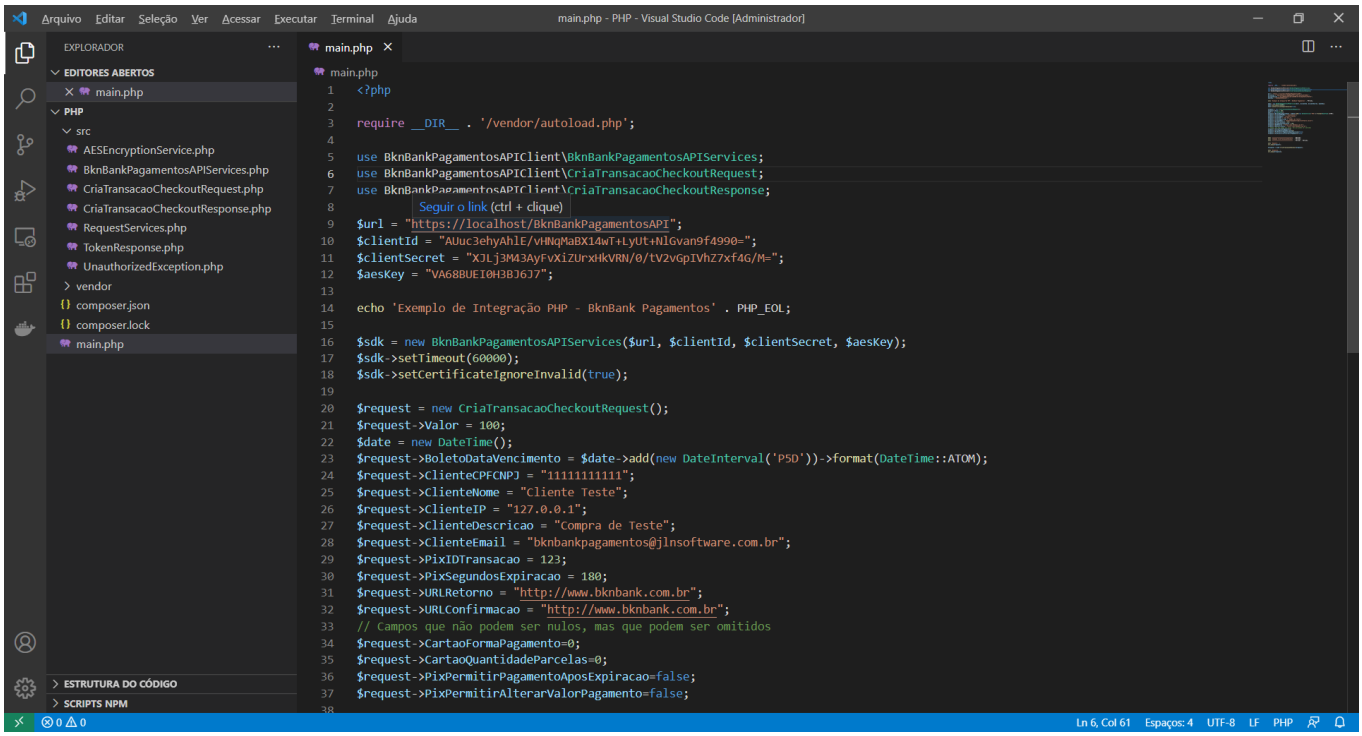
Execurar o comando composer install:

```
TERMINAL  PROBLEMAS  SAÍDA  CONSOLE DE DEPURAÇÃO                    1: powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos os direitos reservados.

Experimente a nova plataforma cruzada PowerShell https://aka.ms/pscore6

PS C:\Dev\BknBank\BknBankPagamentos\SDKs\PHP> composer install
Installing dependencies from lock file (including require-dev)
Verifying lock file contents can be installed on current platform.
Package operations: 7 installs, 0 updates, 0 removals
  - Installing psr/http-message (1.0.1): Extracting archive
  - Installing psr/http-client (1.0.1): Extracting archive
  - Installing ralouphie/getallheaders (3.0.3): Extracting archive
  - Installing guzzlehttp/psr7 (1.7.0): Extracting archive
  - Installing guzzlehttp/promises (1.4.0): Extracting archive
  - Installing guzzlehttp/guzzle (7.2.0): Extracting archive
  - Installing netresearch/jsonmapper (v3.1.1): Extracting archive
Generating autoload files
1 package you are using is looking for funding.
Use the `composer fund` command to find out more!
PS C:\Dev\BknBank\BknBankPagamentos\SDKs\PHP>
```

A execução cria uma pasta “vendor” abaixo do projeto, que tem as dependências instaladas.

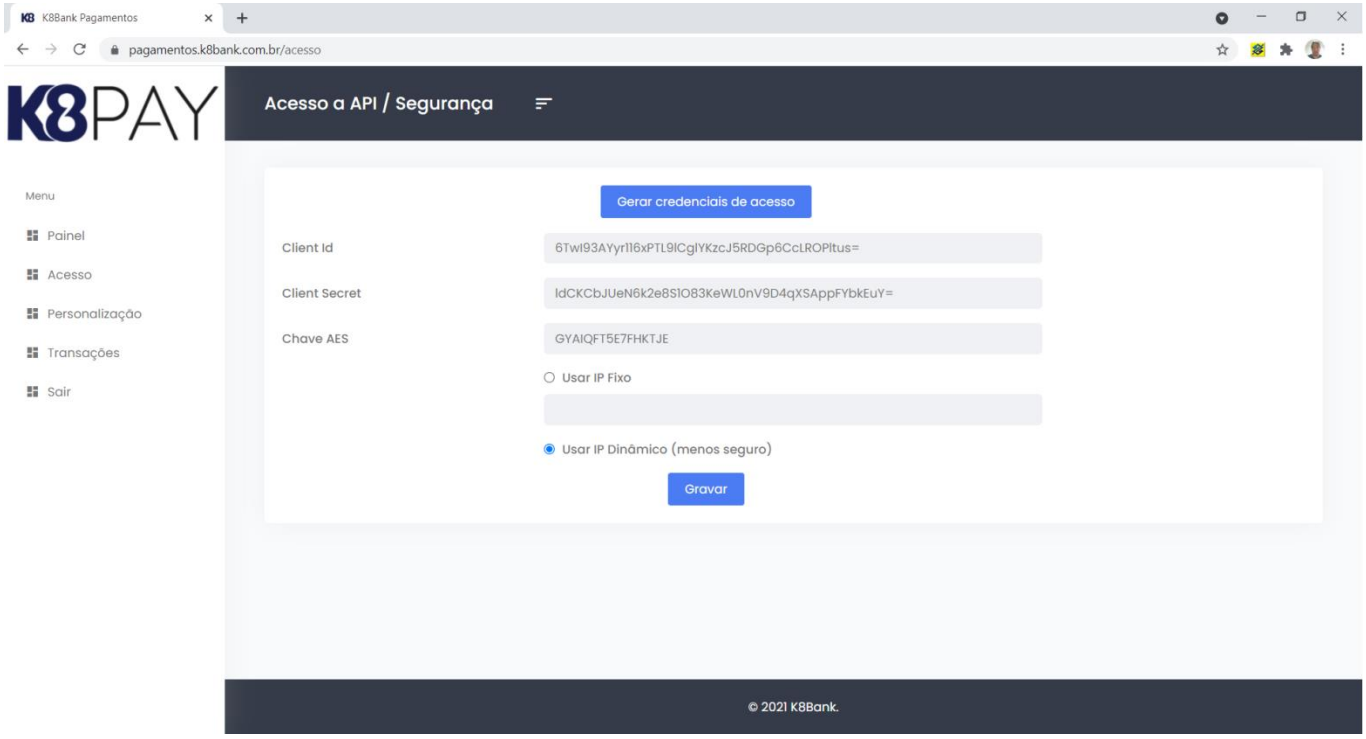
3.3.3. Parametrização

Abrir o arquivo main.php e notar os valores dos parâmetros $url, $clientId, $clientSecret e $aesKey:




```php
<?php

require __DIR__ . '/vendor/autoload.php';

use BknBankPagamentosAPIClient\BknBankPagamentosAPIServices;
use BknBankPagamentosAPIClient\CriaTransacaoCheckoutRequest;
use BknBankPagamentosAPIClient\CriaTransacaoCheckoutResponse;

$url = "https://localhost/BknBankPagamentosAPI";
$clientId = "AUuc3ehyAhlE/vHNqMaBX14wT+LyUt+NlGvan9f4990=";
$clientSecret = "XJLj3M43AyFvXiZUrxHkVRN/0/tV2vGpIVhZ7xf4G/M=";
$aesKey = "VA68BUEI0H3BJ6J7";

echo 'Exemplo de Integração PHP - BknBank Pagamentos' . PHP_EOL;

$sdk = new BknBankPagamentosAPIServices($url, $clientId, $clientSecret, $aesKey);
$sdk->setTimeout(60000);
$sdk->setCertificateIgnoreInvalid(true);

$request = new CriaTransacaoCheckoutRequest();
$request->Valor = 100;
$date = new DateTime();
$request->BoletoDataVencimento = $date->add(new DateInterval('P5D'))->format(DateTime::ATOM);
$request->ClienteCPFCNPJ = "11111111111";
$request->ClienteNome = "Cliente Teste";
$request->ClienteIP = "127.0.0.1";
$request->ClienteDescricao = "Compra de Teste";
$request->ClienteEmail = "bknbankpagamentos@jlnsoftware.com.br";
$request->PixIDTransacao = 123;
$request->PixSegundosExpiracao = 180;
$request->URLRetorno = "http://www.bknbank.com.br";
$request->URLConfirmacao = "http://www.bknbank.com.br";
// Campos que não podem ser nulos, mas que podem ser omitidos
$request->CartaoFormaPagamento=0;
$request->CartaoQuantidadeParcelas=0;
$request->PixPermitirPagamentoAposExpiracao=false;
$request->PixPermitirAlterarValorPagamento=false;
```

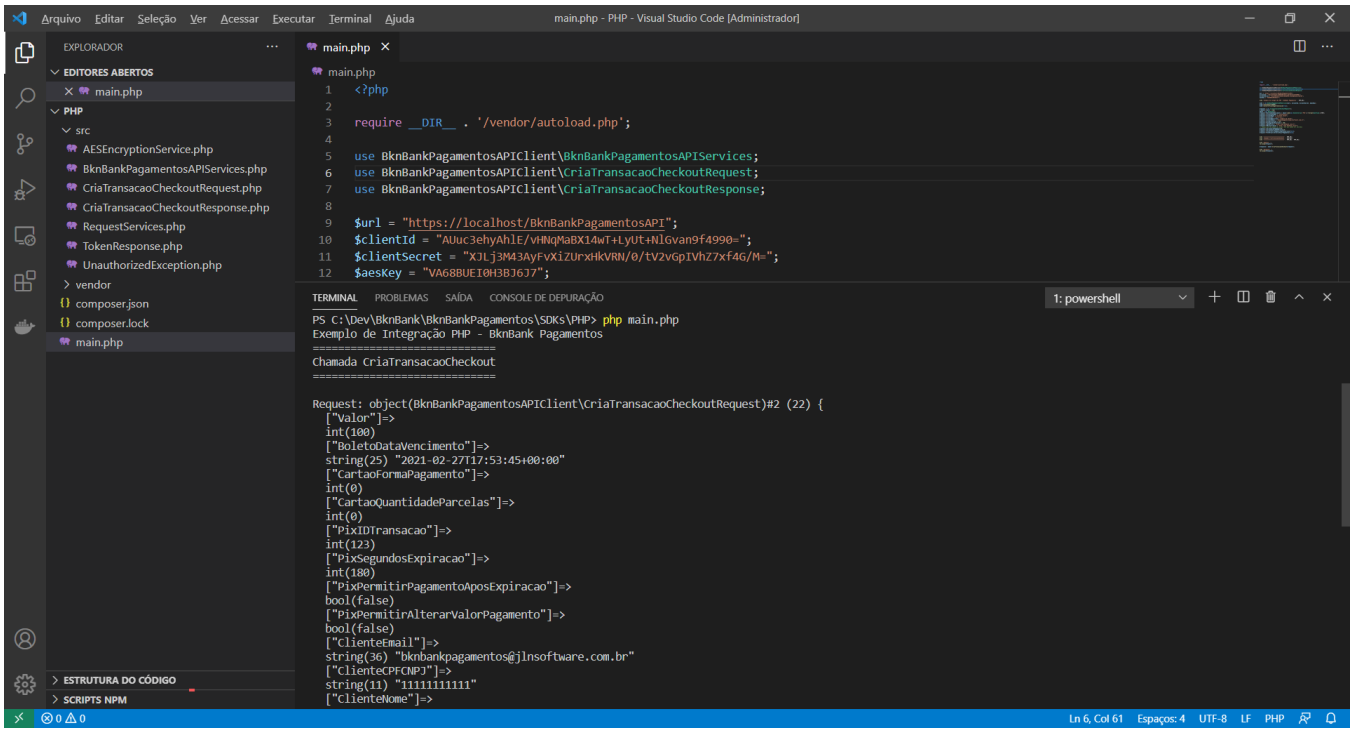
Para obter os valores a serem utilizados, logar na interface do K8Bank Pagamentos e selecionar a opção de menu Acesso:

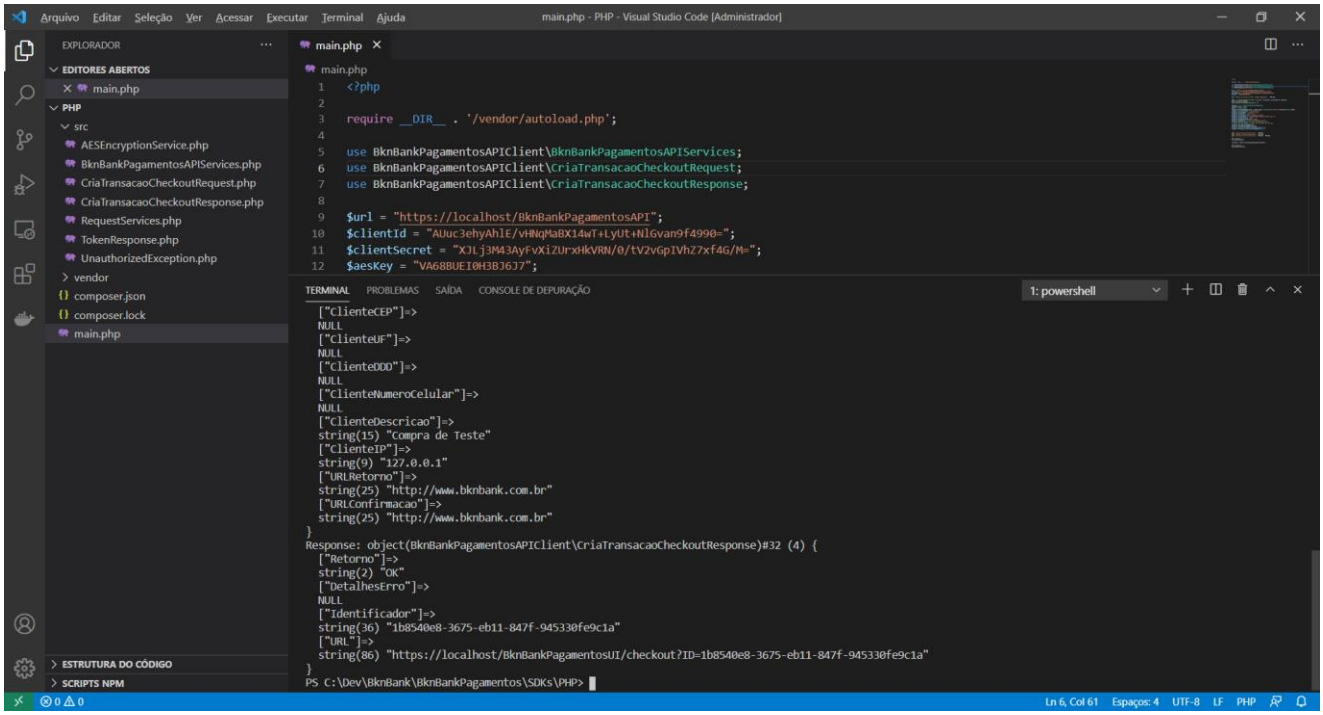


Notar os parâmetros ClientId, Client Secret e Chave AES. Notar ainda que é possível especificar o uso de um IP fixo ou de IP dinâmico. Ajustar ainda a URL de produção do sistema.

3.3.4. Execução

Para executar a SDK em PHP, abrir o terminal e executar o comando php main.php:





Notar a saída da execução do programa, invocando o método CriaTransacaoCheckout, enviando os parâmetros e obtendo a respectiva resposta.

3.4. Linguagem Javascript/Typescript (Node.js)

3.4.1. Pré Requisitos

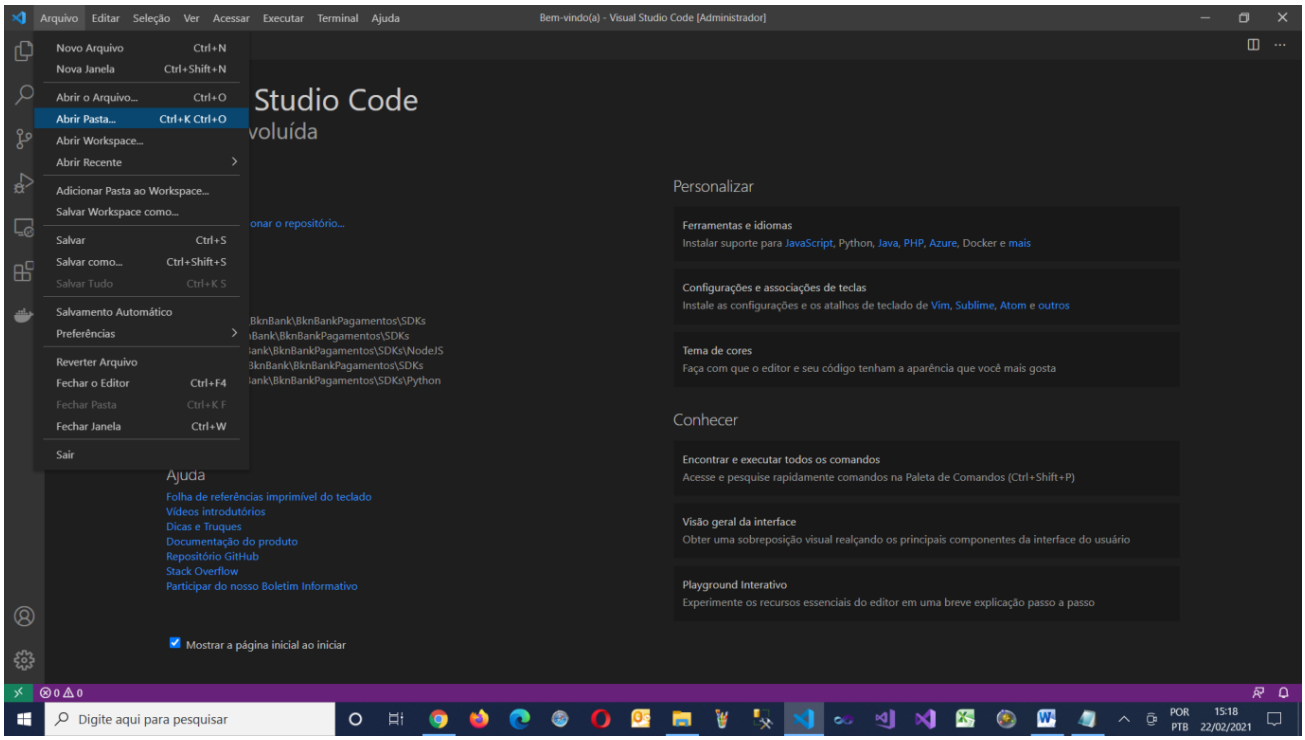
Para utilizar a SDK para Node.js, é necessário ter os seguintes softwares instalados:

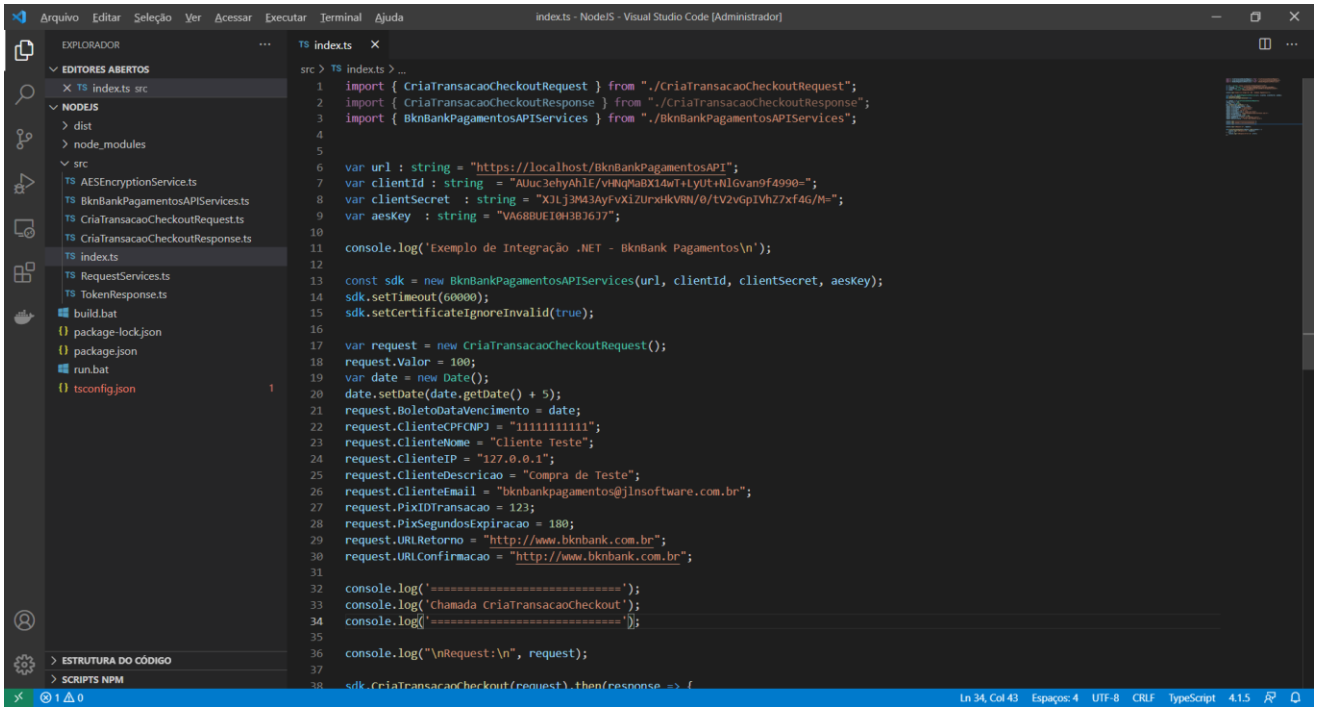
- Nodejs 12.19.0 ou superior
    - https://nodejs.org/pt-br/download/
    - Notar que a instalação já agrega os executáveis (de C:\Program Files\nodejs\) no Path do sistema operacional, e também o gerenciador de pacotes npm.
- Visual Studio Code
    - https://code.visualstudio.com/download

      Efetuar o download do instalador e realizar a instalação.

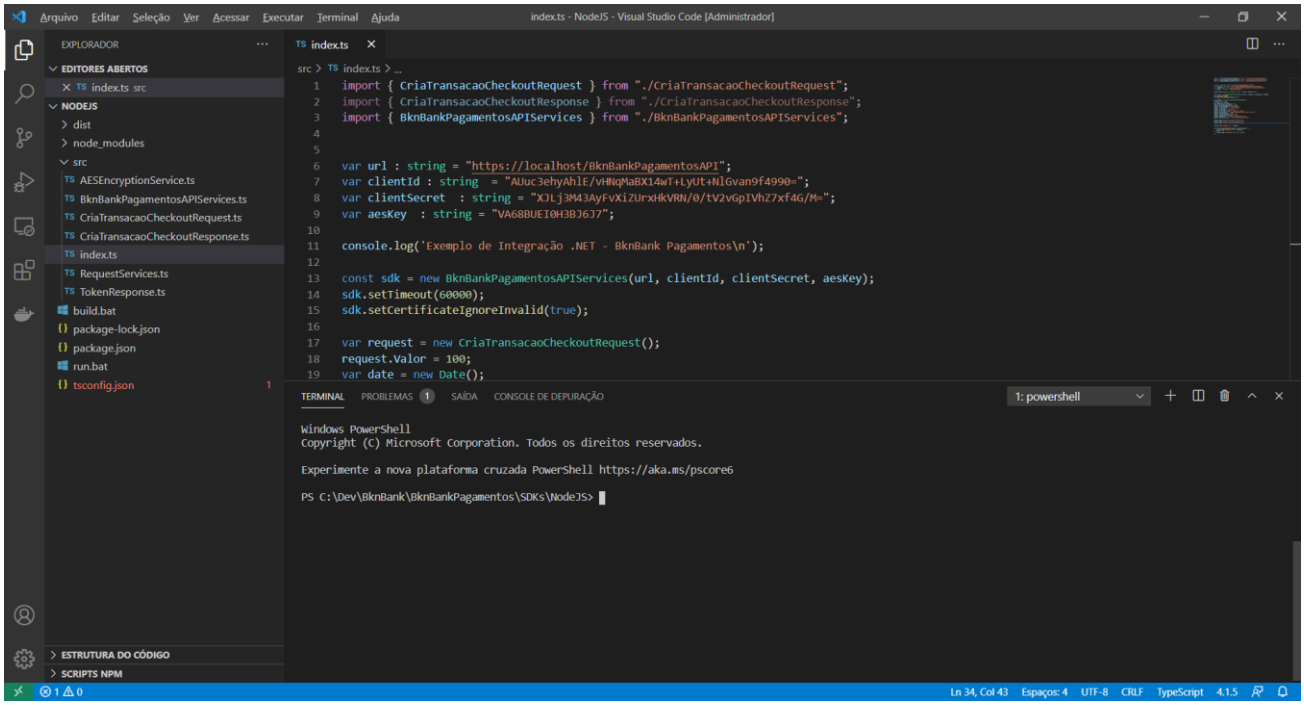
3.4.2. Instalação/Compilação

No programa Visual Studio Code, clicar em Arquivo , Abrir Pasta…, e abrir a pasta NodeJS abaixo do SDK:





Instalar os pacotes necessários com o comando **npm install**, abrindo um terminal (menu Ver->Terminal do Visual Studio Code):



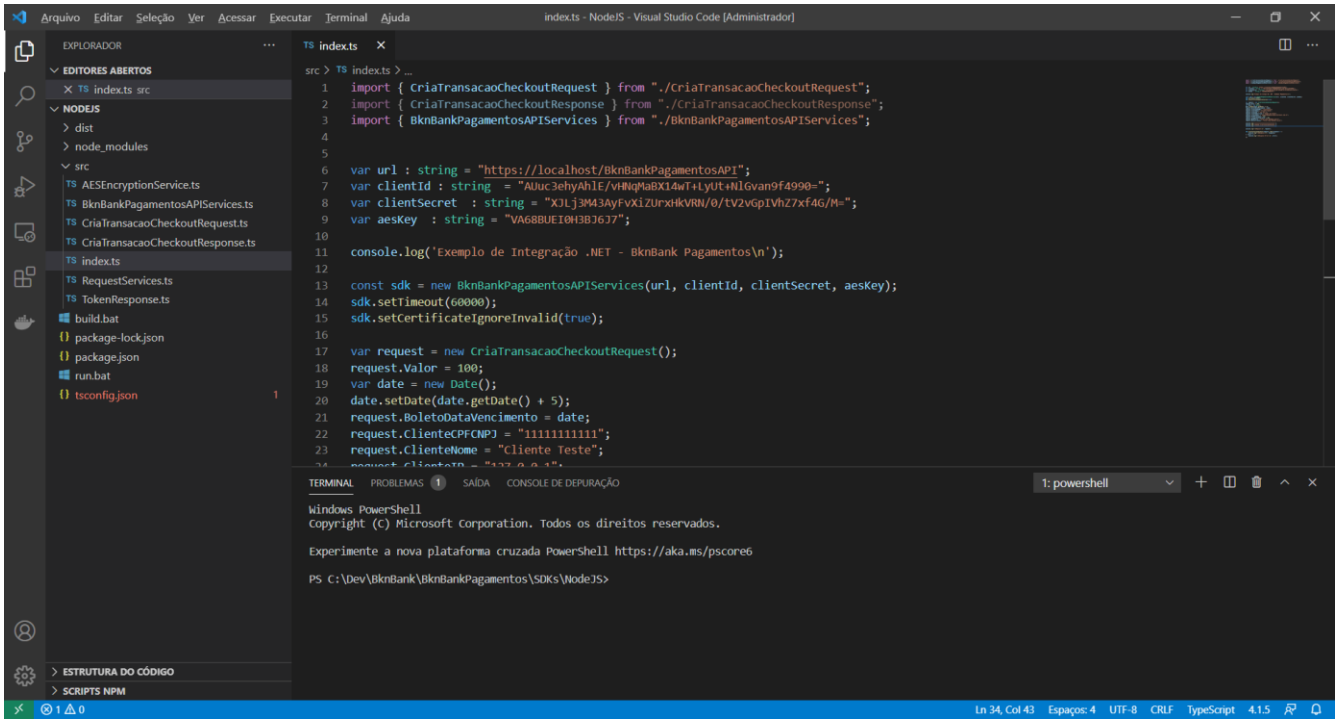
```
TERMINAL  PROBLEMAS 1  SAÍDA  CONSOLE DE DEPURAÇÃO                    1: powershell

Windows PowerShell
Copyright (C) Microsoft Corporation. Todos os direitos reservados.

Experimente a nova plataforma cruzada PowerShell https://aka.ms/pscore6

PS C:\Dev\BknBank\BknBankPagamentos\SDKs\NodeJS> npm install
npm WARN bknbankpagamentosapiclient@1.0.0 No description
npm WARN bknbankpagamentosapiclient@1.0.0 No repository field.

added 3 packages from 44 contributors and audited 3 packages in 1.894s
found 0 vulnerabilities

PS C:\Dev\BknBank\BknBankPagamentos\SDKs\NodeJS>
```

A execução cria uma pasta “node_modules” no projeto, que tem as dependências instaladas.

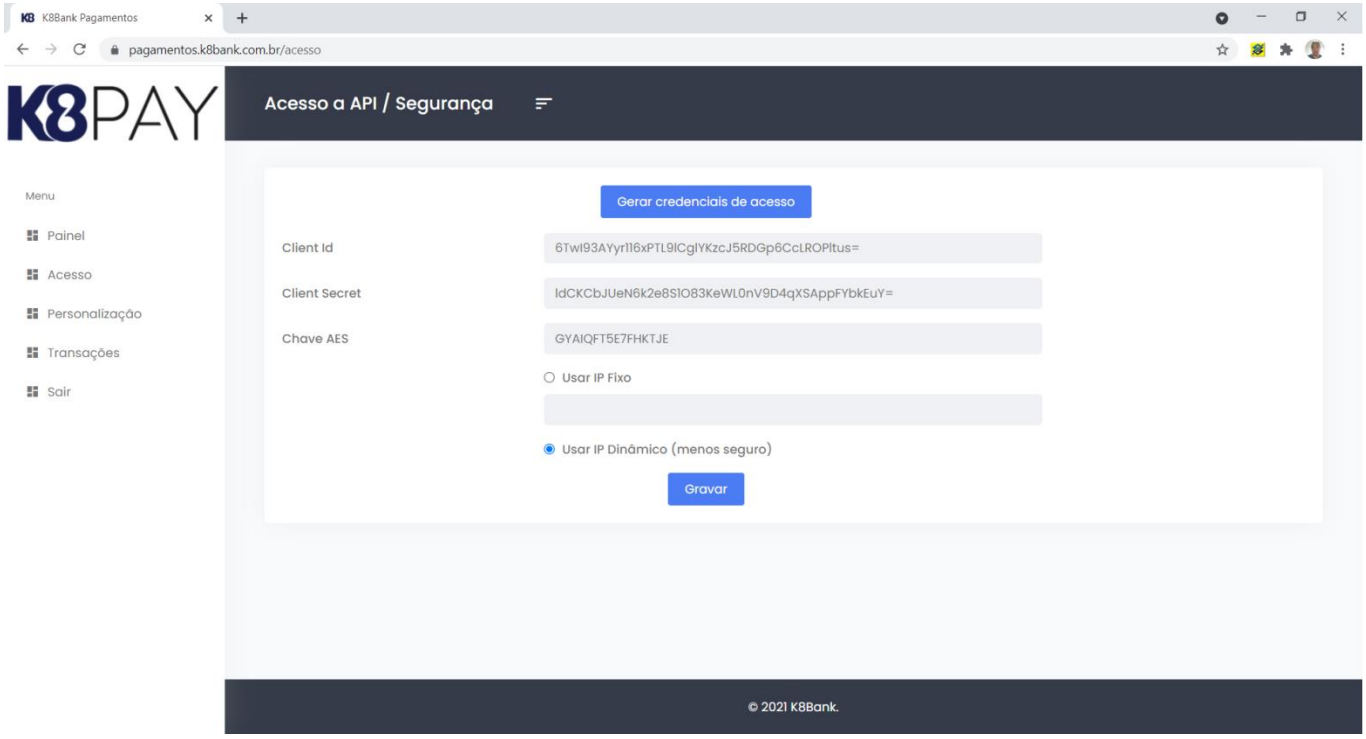
Se for desejado compilar os arquivos .ts (Typescript) para geração dos .js (Javascript) respectivos, utilizar o comando npm run build, como indicado no arquivo “build.bat”.

3.4.3. Parametrização

Abrir o arquivo index.ts e notar os valores dos parâmetros url, clientId, clientSecret e aesKey:



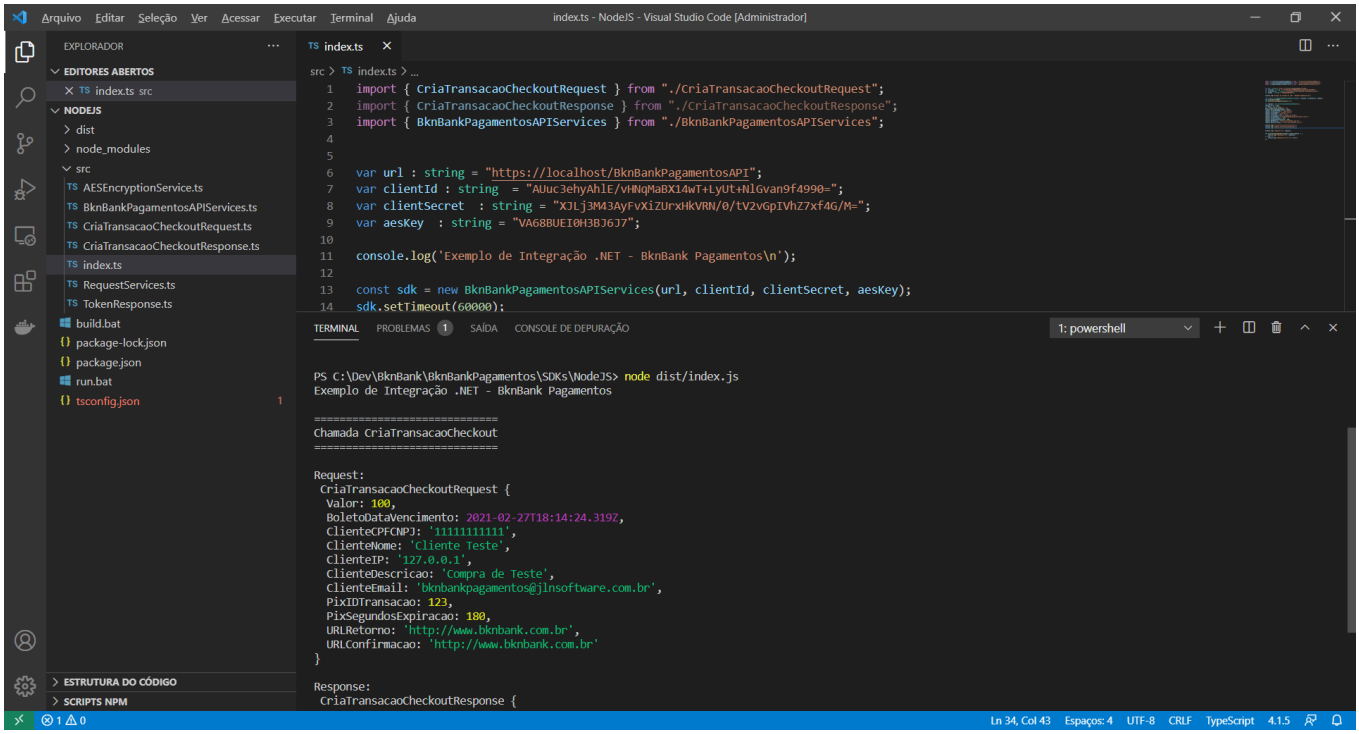
Para obter os valores a serem utilizados, logar na interface do K8Bank Pagamentos e selecionar a opção de menu Acesso:

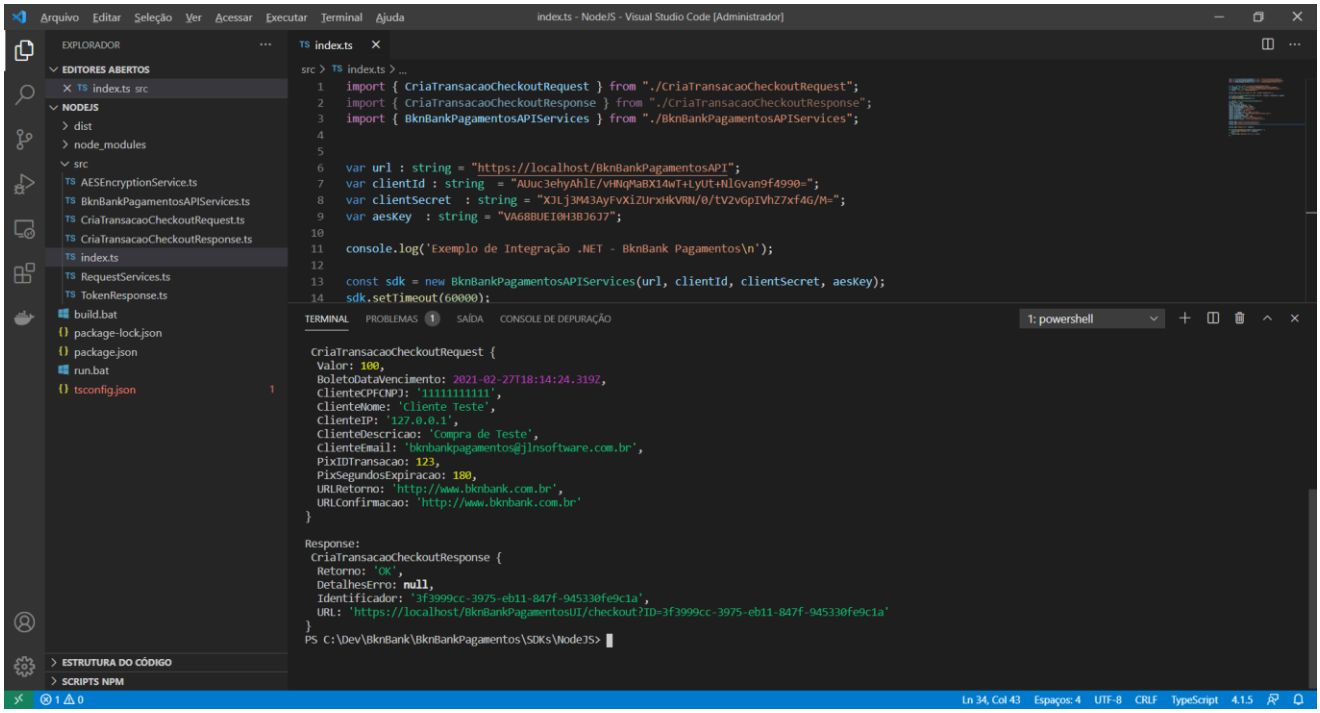


Notar os parâmetros ClientId, Client Secret e Chave AES. Notar ainda que é possível especificar o uso de um IP fixo ou de IP dinâmico. Ajustar ainda a URL de produção do sistema.

3.4.4. Execução

Para executar a SDK em Node.js, abrir o terminal e executar o comando node dist/index.js:





Notar a saída da execução do programa, invocando o método CriaTransacaoCheckout, enviando os parâmetros e obtendo a respectiva resposta.

3.5. Linguagem Python

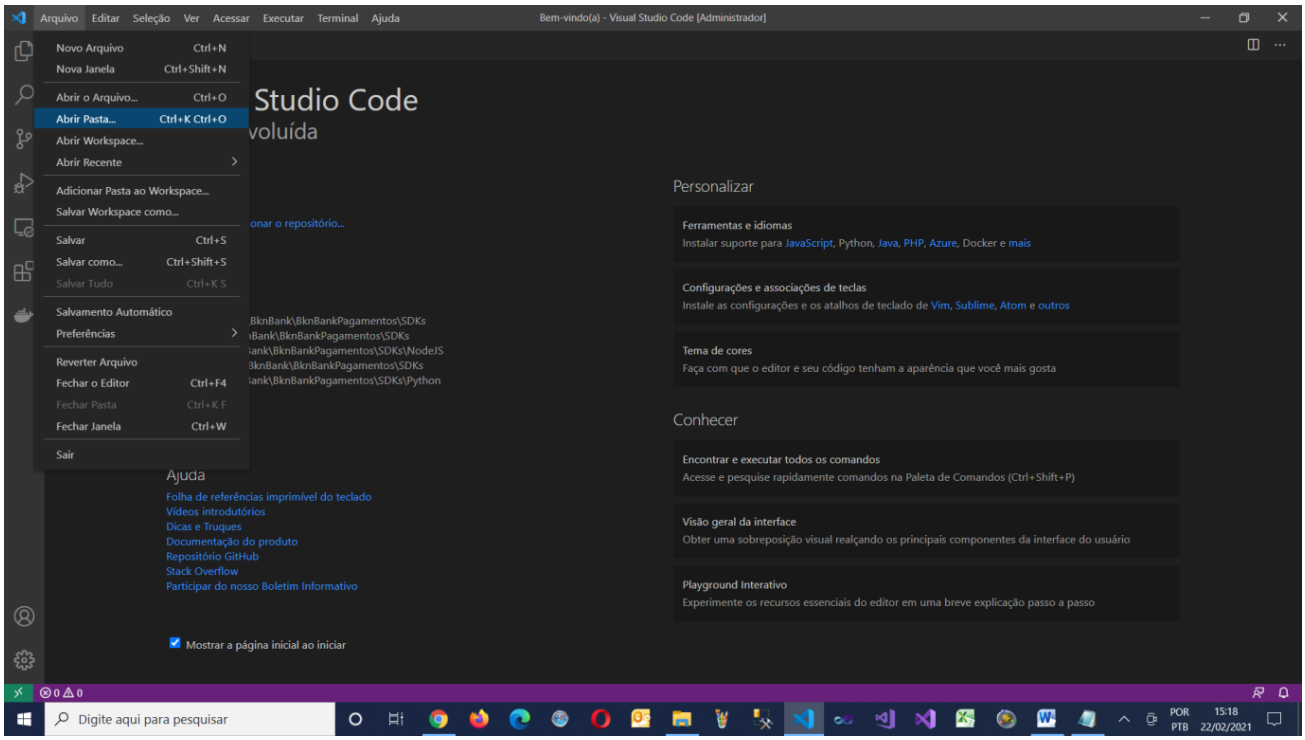
3.5.1. Pré Requisitos

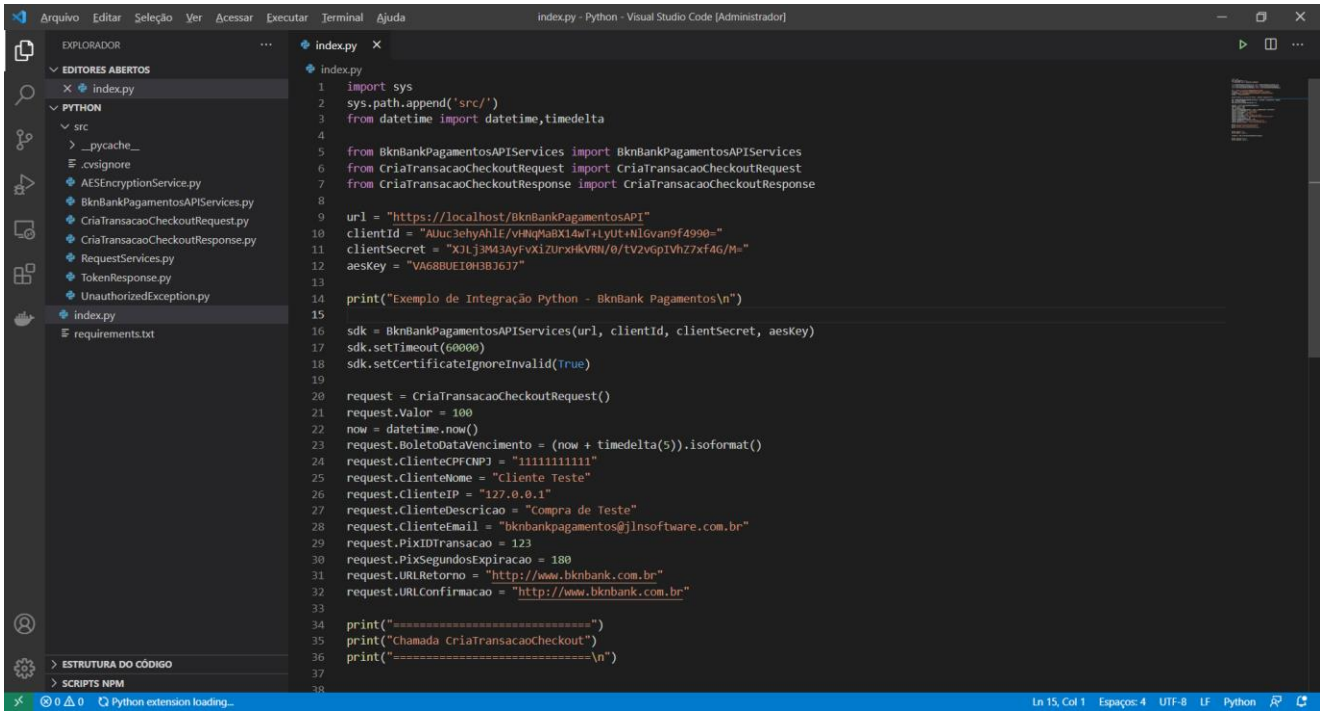
Para utilização da SDK para Python, é necessário ter os seguintes softwares instalados:

- Python 3.8.3 o superior
    - https://www.python.org/downloads/
    - Este instalador já adiciona os executáveis do Python na variável de ambiente Path e o gerenciador de pacotes Pip.
- Visual Studio Code
    - https://code.visualstudio.com/download
    - Efetuar o download do instalador e realizar a instalação.

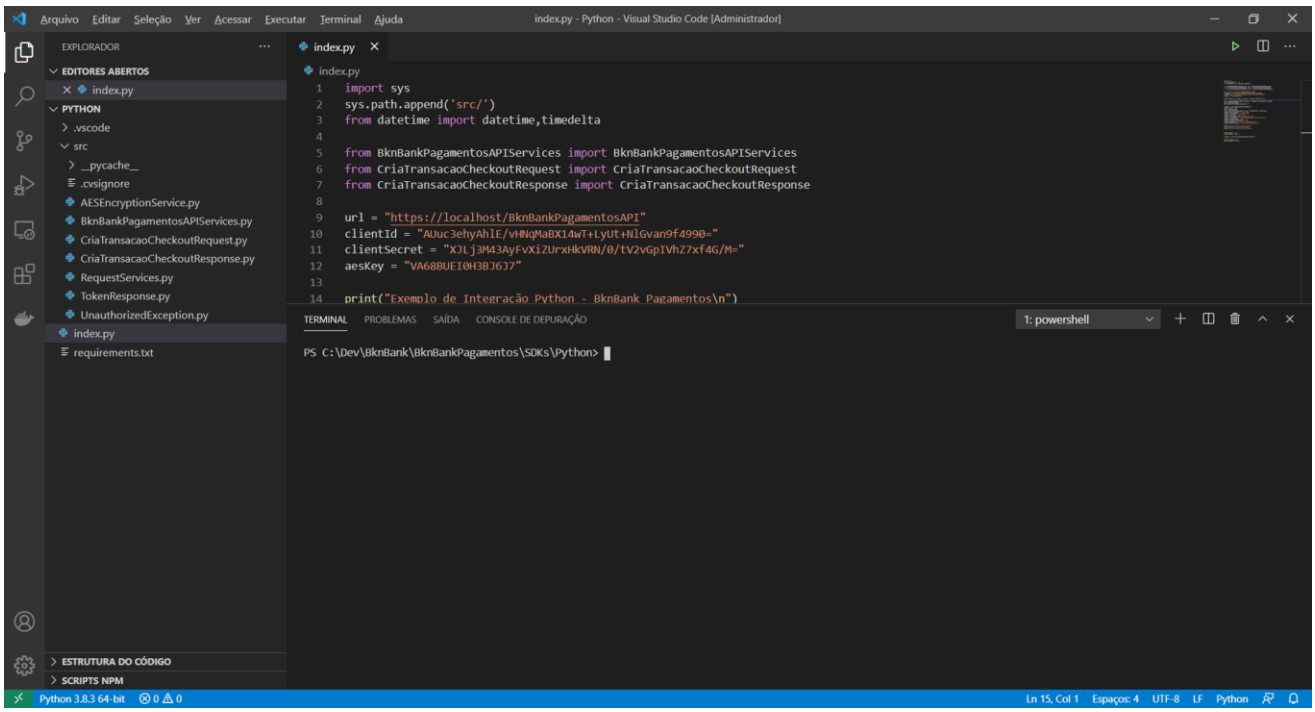
3.5.2. Instalação/Compilação

No programa Visual Studio Code, clicar em Arquivo , Abrir Pasta…, e abrir a pasta Python abaixo do SDK:

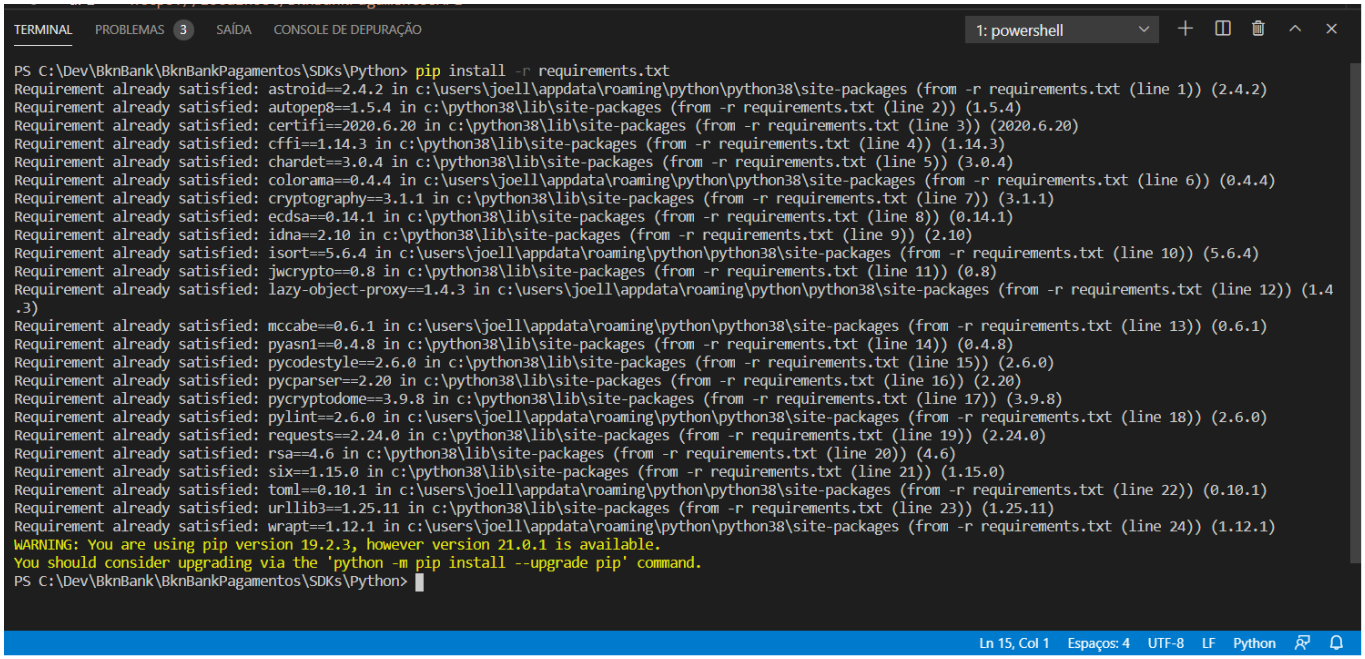




Instalar os pacotes necessários com o comando **pip install -r requirements.txt**, abrindo um terminal (menu Ver->Terminal do Visual Studio Code):

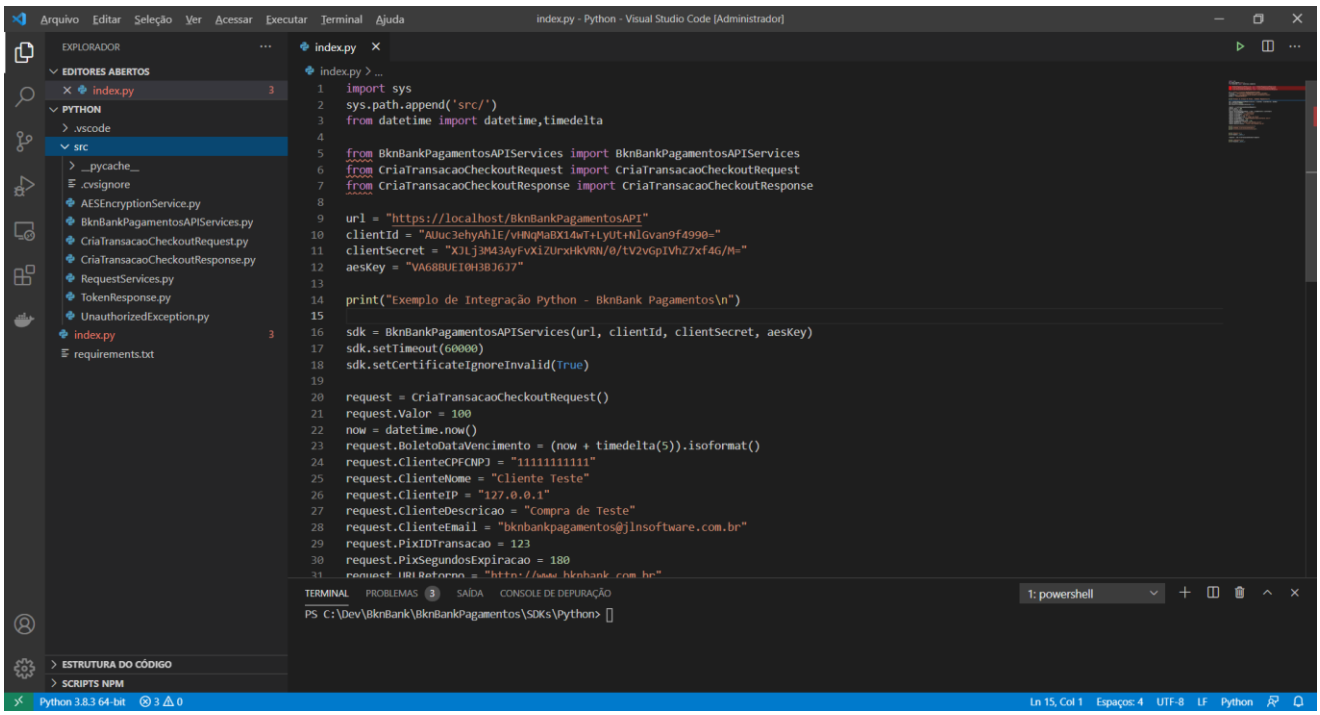


Que instala os pré-requisitos:

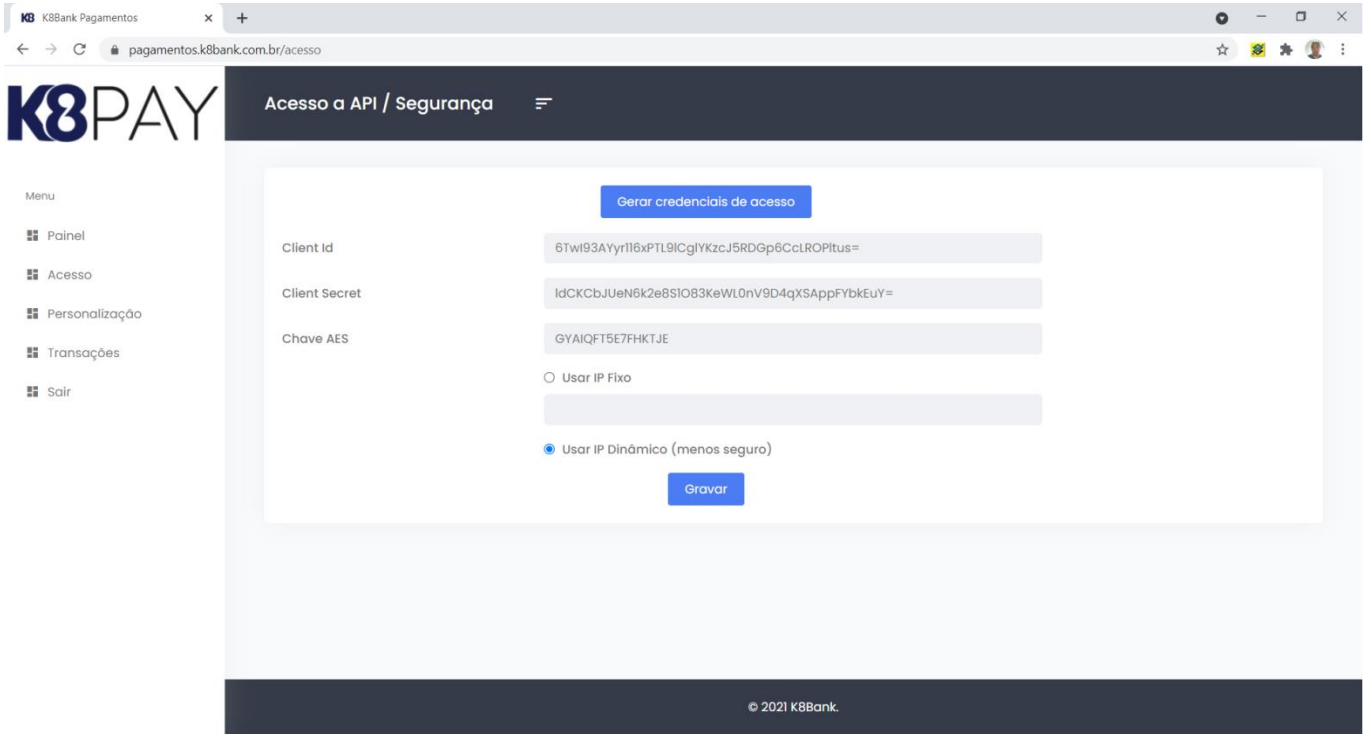


3.5.3. Parametrização

Abrir o arquivo index.py e notar os valores dos parâmetros url, clientId, clientSecret e aesKey:



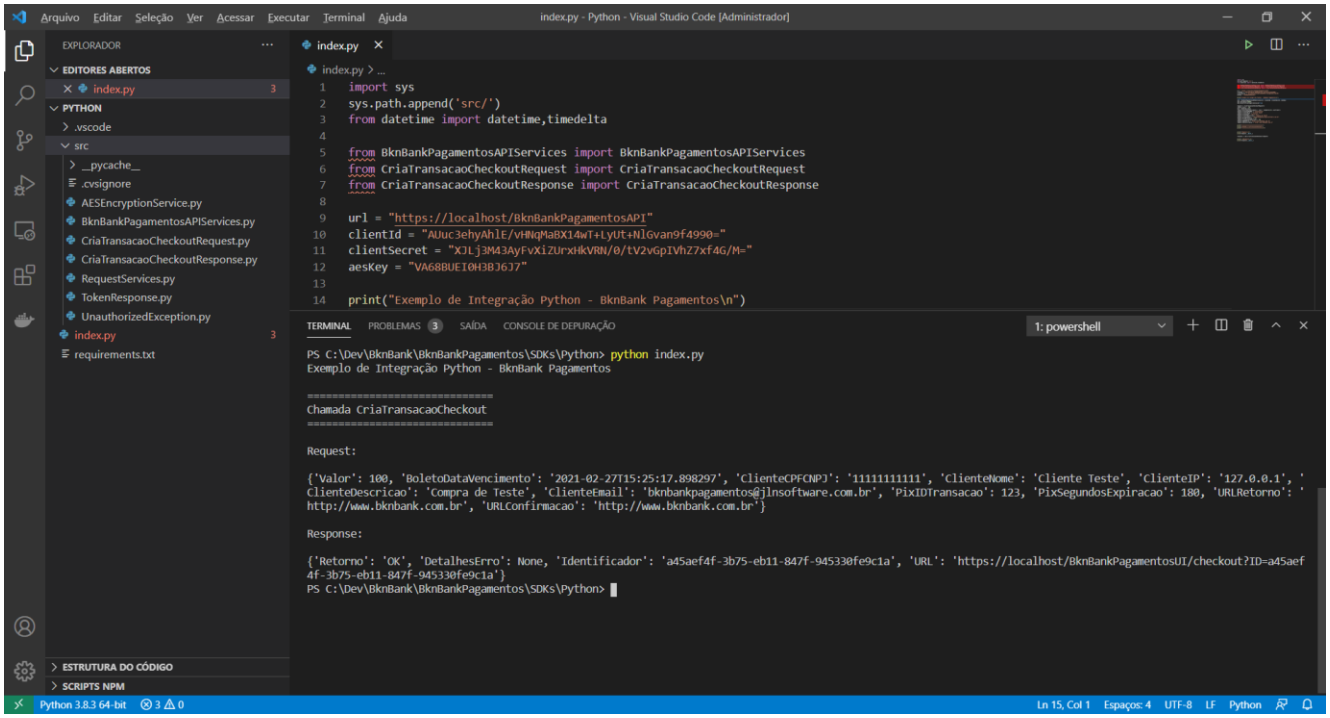
Para obter os valores a serem utilizados, logar na interface do K8Bank Pagamentos e selecionar a opção de menu Acesso:



Notar os parâmetros ClientId, Client Secret e Chave AES. Notar ainda que é possível especificar o uso de um IP fixo ou de IP dinâmico. Ajustar ainda a URL de produção do sistema.

3.5.4. Execução

Para executar a SDK em Python, abrir o terminal e executar o comando python index.py:



Notar a saída da execução do programa, invocando o método CriaTransacaoCheckout, enviando os parâmetros e obtendo a respectiva resposta.