

🔗 gcg-cg

Material da disciplina de Computação Gráfica que utiliza o OpenTK como renderizador.

🔗 OpenTK / SDK do .NET Core / VSCode

Como executar aplicações utilizando o OpenTK no Visual Studio Code

Pré-requisitos _____

1. ter o SDK do .NET Core (<https://www.microsoft.com/net/download>)
2. ter o Visual Studio Code (<https://code.visualstudio.com/>)
3. instalar as extensões do VSCode (diponíveis no site ou pelo próprio editor - <https://code.visualstudio.com/docs/editor/extension-gallery>):
 - o C# (<https://marketplace.visualstudio.com/items?itemName=ms-vscode.csharp>)
 - o NuGet Package Manager (<https://marketplace.visualstudio.com/items?itemName=jmrog.vscode-nuget-package-manager>)

🔗 Passo a passo

Para os passos a seguir é possível utilizar o prompt do Windows (cmd) ou o terminal do VSCode.

Crie uma nova pasta que será o diretório do projeto OpenTK no VSCode e navegue até ela. Nesse exemplo o nome da pasta será 'OlaMundo':

```
$ mkdir OlaMundo; cd OlaMundo
```

Em seguida crie um Console Application nessa pasta:

```
$ dotnet new console
```

Nesse ponto um novo arquivo Program.cs contendo um método main é criado automaticamente. Para executar o projeto digite:

```
$ dotnet run
```

Se o projeto foi criado corretamente a mensagem 'Hello World' aparecer no terminal.

Agora para incluir o OpenTK e os outros pacotes necessários no projeto digite:

```
$ dotnet add package OpenTK --version 3.0.1
$ dotnet add package Microsoft.Win32.SystemEvents --version 4.5.0
$ dotnet add package System.Drawing.Common --version 4.5.0
```

Esses comandos estão disponíveis, respectivamente, em:

(<https://www.nuget.org/packages/OpenTK/3.0.1>)

(<https://www.nuget.org/packages/Microsoft.Win32.SystemEvents/>)

(<https://www.nuget.org/packages/System.Drawing.Common/>)

Nesse ponto, para testar se o OpenTK está funcionando, acrescente as duas linhas de código a seguir no método main e re-execute o projeto (não se esqueça de adicionar a linha 'using OpenTK;' no cabeçalho da classe):

```
using OpenTK;
...
GameWindow window = new GameWindow(600, 600);
window.Run(1.0/60.0);
```

Uma nova janela em branco deverá aparecer após a execução.

Caso ocorra algum erro de 'undefined command' tente executar o comando 'dotnet restore' no terminal para recarregar o projeto.

OBS: Caso apareça algum erro do tipo: 'System.IO.FileNotFoundException: Could not load file or assembly...' simplesmente pesquise o nome do arquivo que está faltando no site <https://www.nuget.org/> e execute a versão do comando .NET CLI no diretório do projeto pelo terminal.

Para executar projetos OpenTK no Linux _____

Se estiver usando o Linux e não funcionou a execução do projeto usando os passos acima, mude no arquivo do projeto (extensão .csproj) da linha:

```
<PackageReference Include="OpenTK" Version="3.0.1" />
```

para:

```
<PackageReference Include="OpenTK.NETCore" Version="*" />
```

Para criar Class library _____

Se caso for preciso criar uma biblioteca - Class library (não esqueça de criar uma nova pasta para este projeto):

```
$ dotnet new classlib
```

Nesse ponto um novo arquivo Class1.cs contendo a definição de uma classe é criada automaticamente.
