Foi mencionado na introdução, que não queremos associar a linguagem JavaScript ao browser, apensar de ser o formato em que ela é mais utilizada. Uma das ferramentas que usaremos para interpretar a linguagem e desenvolver nosso programa será o NodeJs.

Ao acessarmos o navegador conseguimos utilizar o console para digitar pequenos códigos e realizar alguns comandos, pois existe uma auxiliador para interpretação do JavaScript.

Como não utilizaremos o browser neste curso, precisaremos instalar o NodeJs na versão 13.5.0 Current. Uma vez que a instalação for concluída, abriremos nosso PowerShell ou terminal do computador.

Ao escreveremos o comando:

node -v

Teremos como retorno a versão do NodeJs instalada em nosso computador para fazermos verificação. Limparemos a tela com o comando clear e então escreveremos node para abrir o interpretador no PowerShell.

Dessa forma,conseguimos realizar operações simples e escrever nosso código, como por exemplo:

const idade = 26

const é um termo reservado do JavaScript, e ao longo do curso entenderemos mais sobre isso e como conseguimos salvar valores dentro de palavras. Também conseguiremos realizar operações matemáticas, como 2+2 e obter o resultado correto.

Podemos escrever também o console.log() com uma mensagem dentro para exibir na tela:

console.log("Meu primeiro programa JS")

Teremos como retorno o texto e a palavra "undefined", e também estudaremos esse termo ao longo do curso.

O JavaScrip não precisa de um ambiente muito carregado para desenvolvermos, e se fecharmos o PowerShell todo nosso código desaparece, isto e, não é salvo. Em ambiente de produção, nossos arquivos são salvos, editados e testados. O PowerShell e o interpreatador - chamado Repl - é uma ferramenta interessante para testarmos pequenas operações.

O que usaremos para de fato criar arquivos será o Visual Studio Code, que também precisará ser instalado na máquina. Trata-se de u editor gratuito que possui muitas funcionalidades interessantes.

No Visual Studio, criaremos um novo arquivo que chamaremos de 1-meuPrimeiroPrograma.js, já que se trata de um arquivo JavaScript, utilizamos a extensão .js.

Nosso arquivo ainda está completamente vazio, poderemos escrever nosso código a partir de agora, e começaremos com o uma declaração simples:

console.log("Meu primeiro programa");

Toda linguagem tem seu nível de formalidade e sua sintaxe, perceba que a inserção ";" toda a vez que terminarmos uma instrução é opcional no caso do JavaScript, mas é uma regra em outras linguagens.

Nosso código, diferentemente do PowerShell, está salvo em arquivo e pode ser aberto posteriormente. Podemos acessar o resultado do programa que escrevemos no PorwerShell, para isso navegaremos até a pasta em que se encontra nosso projeto: escrevermos o comando cd e então digitaremos o endereço da pasta:

cd "D:\HD Estudio Preto\Documentos\Bugan\1617-jsintro\projeto-atual"

Conseguiremos então visualizar o que há dentro da pasta, que será 1-meuPrimeiroPrograma.js. Para de fato vermos o resultado do programa, primeiramente limparemos a tela por meio do comando clear.

Feito isso, escreveremos o comando node, mais caminho do arquivo em que está nosso programa.

node .\1-meuPrimeiroPrograma.js

Ao acionarmos a tecla "Enter" teremos impresso na tela "Meu primeiro programa" exatamente como escrevemos no Visual Studio Code.

Este é o fluxo que seguiremos em nosso curso, ora no Visual Studio ora no PowerShell para testar e executar nosso código.