

CAPÍTULO 1 – Ambiente e fundamentos do Node.js

Neste capítulo você vai ver:

- Como configurar o ambiente de desenvolvimento fazendo a instalação dos softwares Visual Studio Code e o Node.js.
- Irá escrever seus primeiros scripts e ver como executá-lo no node.js

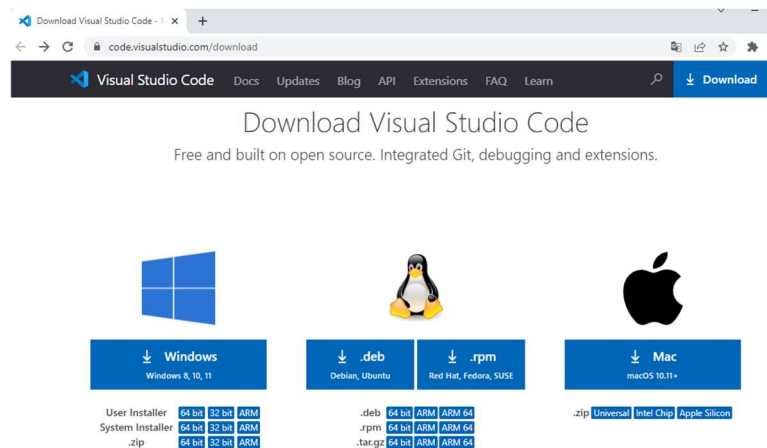
Atividade 1 – Download e instalação do editor de código fonte

Essa atividade tem como objetivo realizar o download e instalação do VS Code.

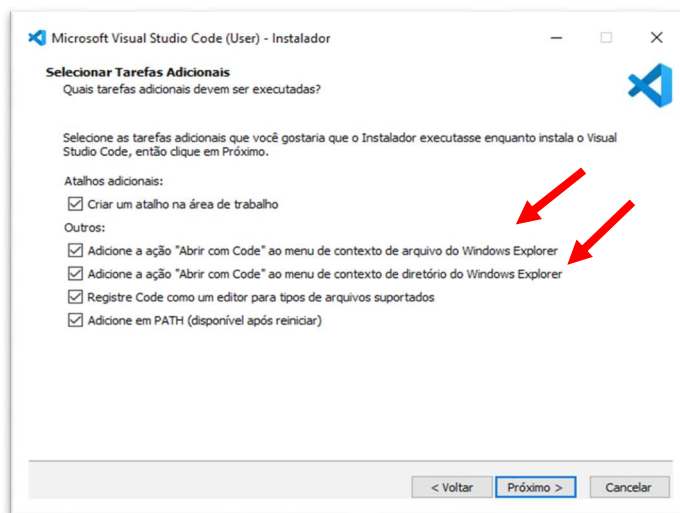
Obs.: Qualquer outro editor de código fonte JavaScript pode ser utilizado no desenvolvimento de aplicações Node.js, porém, o VS Code é o escolhido para o desenvolvimento deste curso.

Realizar o download e instalar o Visual Studio Code

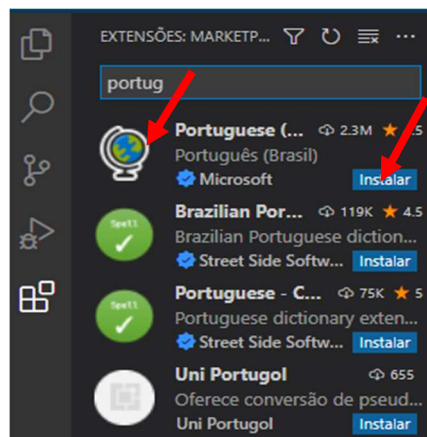
1. Acesse a url <https://code.visualstudio.com/download> e para realizar o download da versão mais atual, clique na opção compatível com o seu sistema operacional, por exemplo: Windows;



2. Após a conclusão do download, execute o instalador e siga os passos do assistente aceitando o contrato de licença, e clicando no botão que indique o passo seguinte. Para um manejo mais fácil dos projetos é indicado habilitar as duas opções indicadas na ilustração abaixo que por padrão aparecem desativadas.



3. A instalação do VS Code é feita nativamente em inglês, mas se caso desejar pode-se instalar uma extensão par a ativação do idioma em português brasil, conforme indicado abaixo:



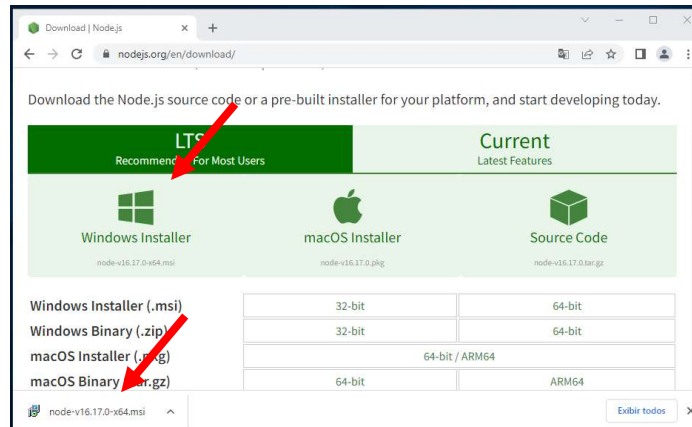
Atividade 2 – Download e instalação do Node.js

Essa atividade tem como objetivo realizar o download e instalação do Node.js e suas principais dependências.

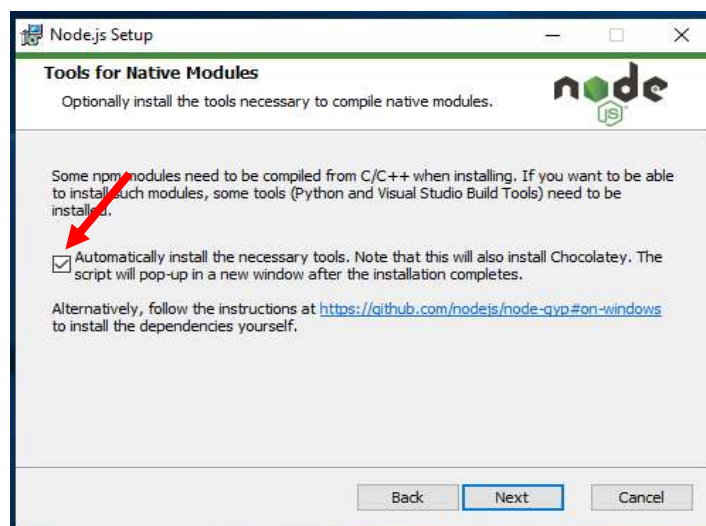
Obs.: Ao realizar esta atividade, procure utilizar sempre a última versão estável (LTS), a versão utilizada neste material é a 16.17.0, os passos podem sofrer pequenas alterações ao ser lançadas novas versões.

Realizar o download e instalar o Node.js

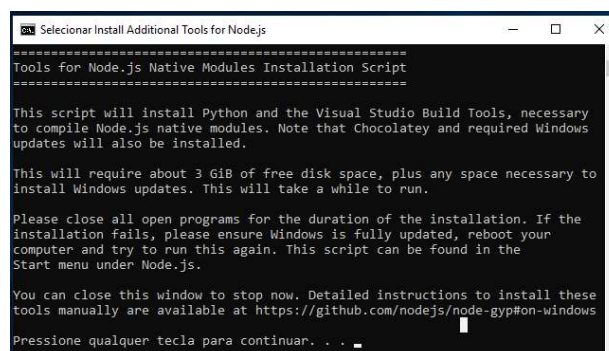
1. Acesse a url <https://nodejs.org/en/download/> e para realizar o download da versão mais atual, clique na opção compatível com o seu sistema operacional, por exemplo: Windows;
2. Após a conclusão do download execute o arquivo, conforme a ilustração abaixo:



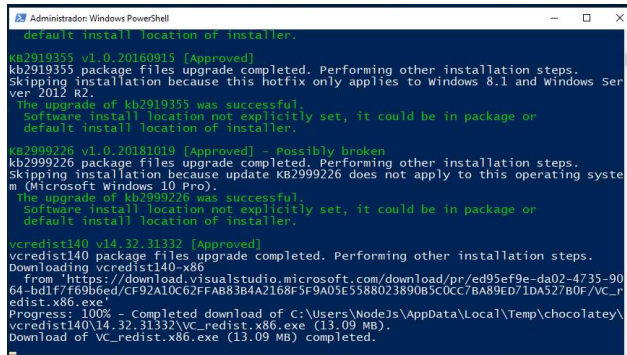
3. Siga os passos do assistente de instalação clicando no botão “Next”, até o passo “**Tools for Native Modules**” e ative a opção “**Automatically install the necessary tools**” para realizar a instalação de ferramentas necessárias para um bom funcionamento do Node.js.



4. Depois clique em próximo e por fim instalar após a conclusão clique em finish.
5. Uma tela do prompt se abrirá uma tela do prompt e iniciará a instalação do Chocolatey e várias bibliotecas e módulos nativos, pressione qualquer tecla para prosseguir com a instalação.

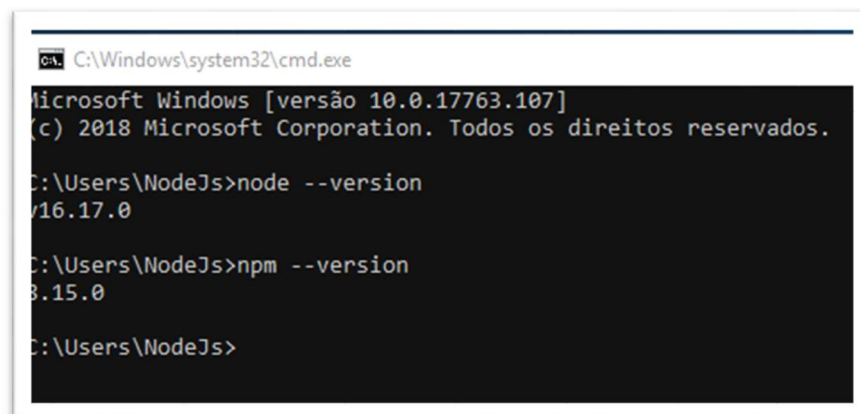


6. Após a janela se fechar, abrirá uma nova janela do Power Shell, onde poderá ser visto o progresso da instalação. Aguarde até a finalização, este processo pode ser bastante demorado, em muitos casos até mais de uma hora.



```
Administrator: Windows PowerShell
default install location of installer.
kb2919355 v1.0.20160915 [Approved]
kb2919355 package files upgrade completed. Performing other installation steps.
Skipping installation because this hotfix only applies to Windows 8.1 and Windows Server 2012 R2.
The upgrade of kb2919355 was successful.
Software install location not explicitly set, it could be in package or default install location of installer.
kb2999226 v1.0.20181019 [Approved] - Possibly broken
kb2999226 package files upgrade completed. Performing other installation steps.
Skipping installation because update KB2999226 does not apply to this operating system (Microsoft Windows 10 Pro).
The upgrade of kb2999226 was successful.
Software install location not explicitly set, it could be in package or default install location of installer.
vc_redist140 v14.32.31332 [Approved]
vc_redist140 package files upgrade completed. Performing other installation steps.
Downloading vc_redist140-x86
from 'https://download.visualstudio.microsoft.com/download/pr/ed95ef9e-da02-4735-9064-bd1f7f69b6ed/cf92a10c62ffab83b4a2168f5f9ad05e5588023890b5c0cc7ba89ed71da527b0f/vc_redist.x86.exe'
Progress: 100% - Completed download of C:\Users\NodeJs\AppData\Local\Temp\chocolatey\vc_redist140\14.32.31332\VC_redist.x86.exe (13.09 MB).
Download of VC_redist.x86.exe (13.09 MB) completed.
```

7. Após algum tempo, a instalação finalizará e o power shell será encerrado.
8. Para executar testes digite CMD na pesquisa do Windows e acesse o prompt de comandos.
9. Depois digite os seguintes comandos:
- ```
node -- version
npm -- version
```
10. Se os comandos retornarem suas respectivas versões tudo foi instalado devidamente



```
C:\Windows\system32\cmd.exe
Microsoft Windows [versão 10.0.17763.107]
(c) 2018 Microsoft Corporation. Todos os direitos reservados.

C:\Users\NodeJs>node --version
v16.17.0

C:\Users\NodeJs>npm --version
8.15.0

C:\Users\NodeJs>
```

### **Atividade 3 – Escrever seu primeiro script e executa-lo no Node.js**

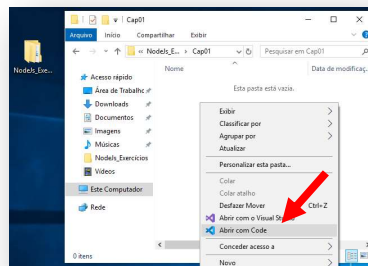
Essa atividade tem como objetivo:

- Escrever o primeiro script e executá-lo no Node.js;
- Importar e utilizar um modulo do Node.js;
- Observar na prática o comportamento da programação assíncrona.

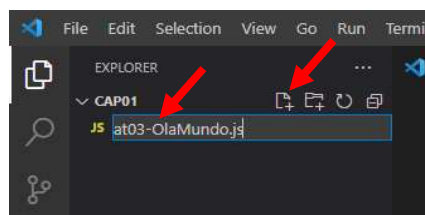
## Escrever o primeiro script e executar no Node.js

1. Crie na área de trabalho uma pasta (ou diretório) chamada “NodeJs\_Exercicios” e crie dentro dela uma subpasta chamada “Cap01”;
2. Abra a pasta e em um lugar em branco clique com o botão direito e escolha a opção “**abrir com Code**” para carregar a pasta no VS Code como diretório de um projeto.

**Obs:** Outras formas também podem ser utilizadas, tais como abrir o VS Code e arrastar a pasta, além de utilizar o menu do VS Code nas opções “File/Open Folder...” e escolher a pasta.



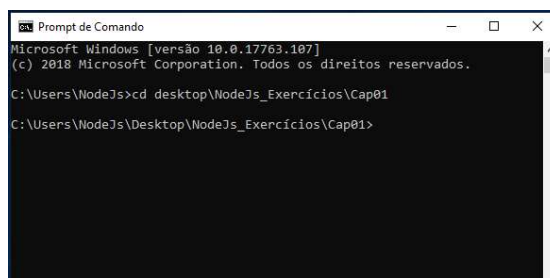
3. Crie um arquivo com o nome “at03-OlaMundo.js” para escrever o primeiro script;



4. Insira o seguinte código JavaScript no arquivo:

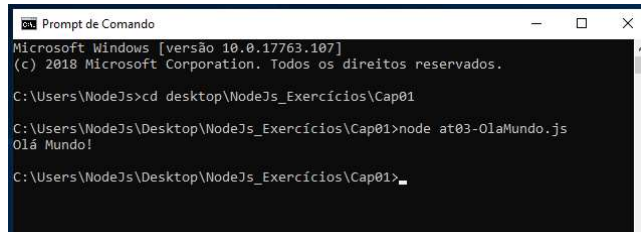
```
console.log('Olá Mundo!')
```

5. Existem algumas formas diferentes de executar um script no node.js, uma delas é utilizar o CMD. Digite CMD na pesquisa do Windows e abra o prompt de comandos;
6. Acesse a pasta do projeto utilizando o comando “**cd desktop\NodeJs\_Exercicios\Cap01**”, conforme exemplo abaixo:



**Obs.:** Caso tenha utilizado outra pasta, adapte o comando “**cd <caminho\_da\_pasta>**”.

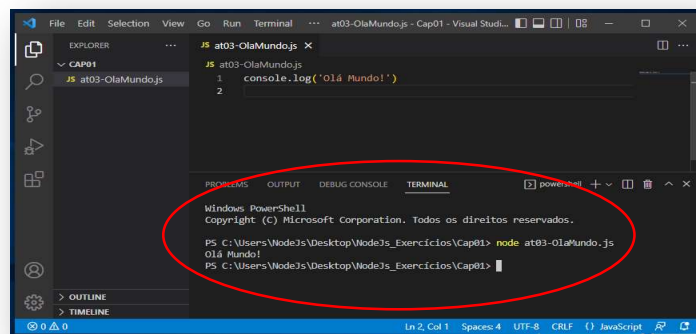
7. Para executar um script no node.js basta digitar o comando “**node <nome\_do\_arquivo>**”. No caso do nosso exemplo o comando apropriado é “**node at03-OlaMundo.js**” e o script será executado, conforme pode ser visto na imagem abaixo:



```
Prompt de Comando
Microsoft Windows [versão 10.0.17763.107]
(c) 2018 Microsoft Corporation. Todos os direitos reservados.

C:\Users\NodeJs>cd desktop\NodeJs_Exercicios\Cap01
C:\Users\NodeJs\Desktop\NodeJs_Exercicios\Cap01>node at03-OlaMundo.js
Olá Mundo!
C:\Users\NodeJs\Desktop\NodeJs_Exercicios\Cap01>
```

8. A execução do script também pode acontecer à partir do próprio VS Code. Voltando ao VS Code, clique no meu Terminal/New Terminal.
9. Note que o terminal já está apontando para o diretório raiz do projeto, de forma que só dar o comando “node at03-OlaMundo”.



## **Atividade 4 – Incrementar o código do primeiro script**

Essa atividade tem como objetivo:

- Elevar a complexidade do primeiro script;
- Ler entradas a partir do console;
- Importar um modulo interno do Node.js;
- Observar na prática o comportamento da programação assíncrona.

Comandos:

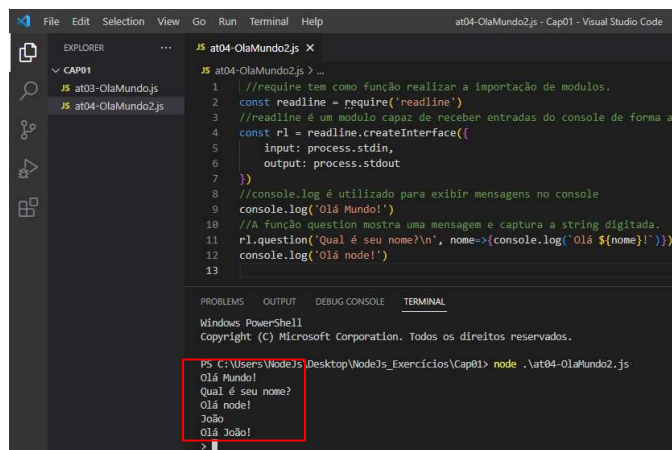
- Declaração de constantes - const
- Importação de um modulo – require('nome\_modulo')
- Uso do modulo readline

## Incrementando o código

1. Crie novo arquivo no diretório do projeto com o nome de “at04-OláMundo2.js”;
2. Insira o seguinte código no arquivo. Cada linha contém um comentário explicativo.

```
//require tem como função realizar a importação de módulos.
const readline = require('readline')
//readline é um modulo capaz de receber entradas do console de forma assíncrona
const rl = readline.createInterface({
 input: process.stdin,
 output: process.stdout
})
//console.log é utilizado para exibir mensagens no console
console.log('Olá Mundo!')
//A função question mostra uma mensagem e captura a Sting digitada.
rl.question('Qual é seu nome?\n', nome=>{
 console.log(`Olá ${nome}!`)
 rl.close()
})
console.log('Olá node!')
```

3. Pressione Ctrl+C para finalizar a execução.
4. Note a ordem de execução do código observando as saídas do console:



5. Perceba que a saída “Olá node!” veio antes da saída “Olá João”, isto porque como tratava-se de um código assíncrono e dependia de uma entrada do usuário.

## Exercícios extras

Estas atividades extras têm como objetivo revisar e aprimorar fundamentos da linguagem JavaScript que serão utilizados nas atividades deste curso.

1. **Pesquise** quais são as **estruturas de repetição** que podem ser utilizadas no JavaScript;
2. **Pesquise e implemente** pelo menos um exemplo das estruturas: **for** - **while** - **do while**
3. Pesquise a diferença entre **break** e **continue**;