

npm & gerência de dependências



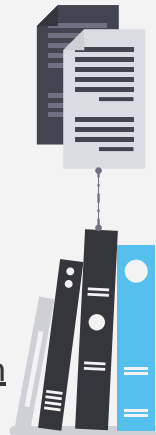
Definição

“npm is the world's largest software registry. Open source developers from every continent use npm to share and borrow packages, and many organizations use npm to manage private development as well.”

<https://docs.npmjs.com/about-npm>

“npm é o gerenciador de pacotes padrão para o ambiente de tempo de execução JavaScript Node.js. Ele consiste em um cliente de linha de comando, também chamado de npm, e um banco de dados online de pacotes públicos e privados pagos, chamado de registro npm. O registro é acessado por meio do cliente e os pacotes disponíveis podem ser navegados e pesquisados no site do npm.”

[https://pt.wikipedia.org/wiki/Npm_\(software\)](https://pt.wikipedia.org/wiki/Npm_(software))





Principais comandos - navegando no terminal

`pwd` (Print Working Directory): Exibe o caminho completo do diretório atual.

Exemplo: `pwd`

`ls` (List): Lista o conteúdo do diretório atual.

Opções Comuns:

`ls -l` : Lista detalhada com permissões, proprietários, tamanho e data de modificação.

`ls -a` : Inclui arquivos e diretórios ocultos.

`ls -lh` : Exibe tamanhos de arquivos em um formato legível para humanos.

Exemplo: `ls -la`

`cd` (Change Directory): Muda para outro diretório.

Exemplo:

Para mudar para o diretório home: `cd ~`

Para mudar para o diretório pai: `cd ..`

Para mudar para um diretório específico: `cd /caminho/para/diretorio`





Principais comandos - navegando no terminal

mkdir (Make Directory): Cria um novo diretório.

Exemplo: `mkdir nome_do_diretorio`

rmdir (Remove Directory): Remove um diretório vazio.

Exemplo: `rmdir nome_do_diretorio`

rm -r (Remove Recursive): Remove um diretório e seu conteúdo de forma recursiva.

Exemplo: `rm -r nome_do_diretorio`

cp (Copy): Copia arquivos ou diretórios.

Exemplo:

Para copiar um arquivo: `cp arquivo_origem arquivo_destino`

Para copiar um diretório de forma recursiva: `cp -r diretorio_origem diretorio_destino`

mv (Move): Move ou renomeia arquivos e diretórios.

Exemplo:

Para mover um arquivo: `mv arquivo_origem caminho_destino`

Para renomear um arquivo: `mv nome_antigo novo_nome`



Principais comandos npm

`npm init`: Inicializa um novo projeto Node.js, criando um arquivo `package.json` que contém informações sobre o projeto e suas dependências.

Exemplo: `npm init` ou `npm init -y` (aceita todas as configurações padrão automaticamente)

`npm install` (ou `npm i`): Instala todas as dependências listadas no arquivo `package.json` na pasta `node_modules`.

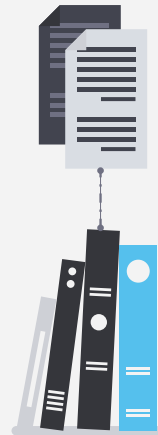
Exemplo: `npm install`

`npm install <package>` (ou `npm i <package>`): Instala um pacote específico e adiciona seu nome e versão ao arquivo `package.json` (na seção `dependencies`).

Exemplo: `npm install express`

`npm install --save-dev <package>` (ou `npm i -D <package>`): Instala um pacote específico e adiciona seu nome e versão ao arquivo `package.json` (na seção `devDependencies`).

Exemplo: `npm install --save-dev jest`



Principais comandos npm

`npm uninstall <package>` (ou `npm un <package>`): Remove um pacote instalado e atualiza o arquivo `package.json` removendo a referência ao pacote.

Exemplo: `npm uninstall express`

`npm update`: Atualiza todos os pacotes instalados para a versão mais recente que respeita as restrições de versão definidas no arquivo `package.json`.

Exemplo: `npm update`

`npm outdated`: Lista todos os pacotes que estão desatualizados em relação às versões mais recentes disponíveis.

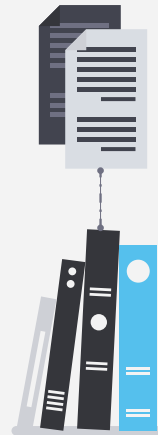
Exemplo: `npm outdated`

`npm list` (ou `npm ls`): Lista todos os pacotes instalados localmente no projeto e suas dependências.

Exemplo: `npm list`

`npm run <script>`: Executa um script definido na seção `scripts` do arquivo `package.json`.

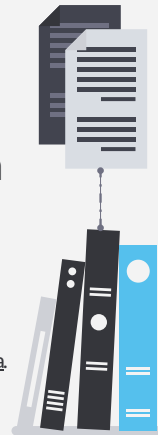
Exemplo: `npm run start` (supondo que haja um script chamado `start` no `package.json`)



Versionamento

“O versionamento consiste em estratégias para gerenciar as diferentes versões de um código, de um sistema ou de um modelo. É uma forma de administrar as mudanças que são feitas e de garantir mais segurança na transição de uma versão para outra.”

<https://blog.somostera.com/data-science/versionamento#:~:text=Trata%2Dse%20de%20um%20conjunto,juntos%2C%20a%20vers%C3%A3o%20do%20sistema>



Versionamento semântico

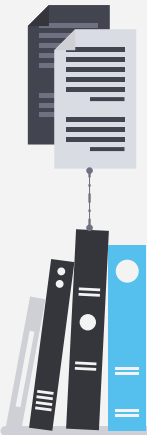
Dado um número de versão **MAJOR.MINOR.PATCH**, incremente a:

versão Maior(MAJOR): quando fizer mudanças incompatíveis na API,

versão Menor(MINOR): quando adicionar funcionalidades mantendo compatibilidade, e

versão de Correção(PATCH): quando corrigir falhas mantendo compatibilidade.

<https://semver.org/lang/pt-BR/>



Hands on

