

## ► TRABALHANDO COM PACOTES *(um pacote é um conjunto de módulos. Onde poder ser usado para varios projetos, precisando apenas importar o pacote para o projeto em uso)*

é possível criar e instalar pacotes num sistema sem precisar saber como foi feito. Basta saber quais métodos tem é suas funções contidas no pacote. *(E saber o que é possível fazer com o pacote)*

▫ Os pacotes podem ser: pacotes pessoais (desenvolvido por mim mesmo) , pacotes criados por terceiros,

### UTILIZANDO PACOTES PESSOAIS:

#### Criando um pacote

*(na pasta do projeto 1)*

- **Deve já possuir todos métodos** *(onde os métodos são os objetos e suas funções. Podendo ser em um arquivo ou varios arquivos que em cada um contém um metodo com objeto funções)*

##### No terminal do Vscode ou cmdr

- Npm init

Sera realizado algumas perguntas:

- package name: (node) definir Nome
  - version: (1.0.0) digita uma versão
  - description: digita a descrição
  - entry point: ( \_\_\_\_.js) digita o nome.js  
*É usado para definir qual o arquivo principal que unir todos os outros métodos*
  - test command:
  - git repository:
  - keywords:
  - author:
  - license: (ISC)
- Confirmar a criação do pacote

**Foi criado o pacote**

Basta olhar nos arquivos do projeto, terá um arquivo **[package.json]**

#### Instalando um pacote

*(na pasta do projeto 2)*

**Pegar todo o endereço da pasta local do pacote do projeto 1 para usar na instalação do pacote no projeto 2.** Exemplo: (C:\Users\amori\Desktop\node.js\)

**(ou caso o pacote esteja em um repositório deve pegar o endereço para instalar)**

**O recomendado ao instalar um pacote em outro projeto é que esse projeto também seja um pacote para isso deve ser criado um pacote para o projeto 2 a ser instalado o pacote do projeto 1**

*Criar o pacote do projeto 2 (mesmo sem ter nada dele ainda. Apenas o arquivo script.js)*

##### No terminal do Vscode ou cmdr

- Npm init

Sera realizado algumas perguntas:

- package name: (node) definir Nome
  - version: (1.0.0) digita uma versão
  - description: digita a descrição
  - entry point: ( \_\_\_\_.js) digita o nome.js  
*É usado para definir qual o arquivo principal que unir todos os outros métodos*
  - test command:
  - git repository:
  - keywords:
  - author:
  - license: (ISC)
- Confirmar a criação do pacote

**Foi criado o pacote**

Basta olhar nos arquivos do projeto, terá um arquivo **[package.json]**

*Instalar o pacote do projeto 1 no projeto 2*

##### No terminal do Vscode ou cmdr

- Npm install "endereço onde está o pacote (local ou remoto)"

*Exemplo de um endereço local:*  
npm install "C:\Users\amori\Desktop\projeto 2 (para usar o pacote)\node.js"

#### Foi instalado o pacote

- Basta olhar no arquivo `package.json` da pasta do projeto 2. terá no final uma dependência que é o pacote instalado.
- E foi criado um arquivo `package-lock.json` dentro da pasta do projeto 2.
- e foi criado uma pasta `node_modules` que contém uma cópia do pacote do projeto 1.

## Utilizando o pacote

*Chamar o pacote para ser usado:*

```
let nomeDaVariavel= require ('nome da pasta do pacote a ser usado');
```

*Chamar uma função de um modulo:*

```
nomeDaVariavel.nome Do Modulo Que Esta A Função(colocar um valor caso a função que está no modulo estiver esperando algum valor de entrada);
```

*Ex1: mostra um resultado encontrado por uma função. Recebendo um valor como parâmetro direto pelo código:*

```
console.log(nomeDaVariavel.nomeDoModuloQueEstaAFunção(40);
```

`pasta calculadora. metodo que está a função ()`

*Ex2: mostra o resultado de uma função. Quando foi digitado valores no terminal*  
`console.log(pasta calculadora. metodo que está a função ())`

*a função vai ser chamada pelo terminal junto com os valores*

```
node index.js nomeDaFunção valor1 valor2
```