



# FIAP

Shift



Aula 04

# NPM & TailwindCSS

Parte 3



# O QUE O MATERIAL COBRE

1

AMBIENTE E BIBLIOTECAS NPM

2

INTRODUÇÃO AO FRAMEWORK TAILWINDCSS

3

INSTALAÇÃO E CONFIGURAÇÃO DO TAILWINDCSS

4

COMO USAR O TAILWINDCSS

5

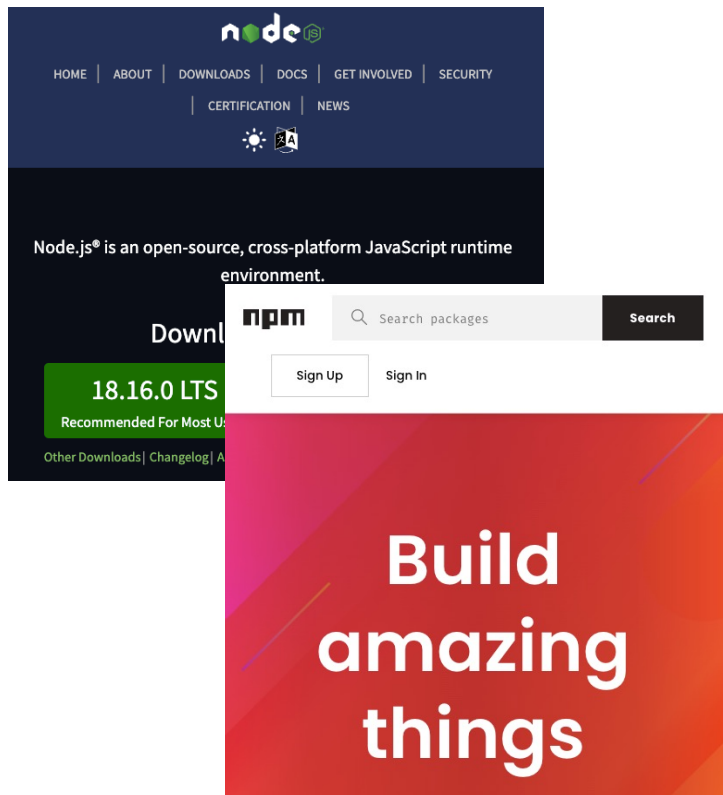
SUGESTÕES DE EXERCÍCIOS

1

# Ambiente e Bibliotecas

## **NPM**

# NPM: Node Package Manager



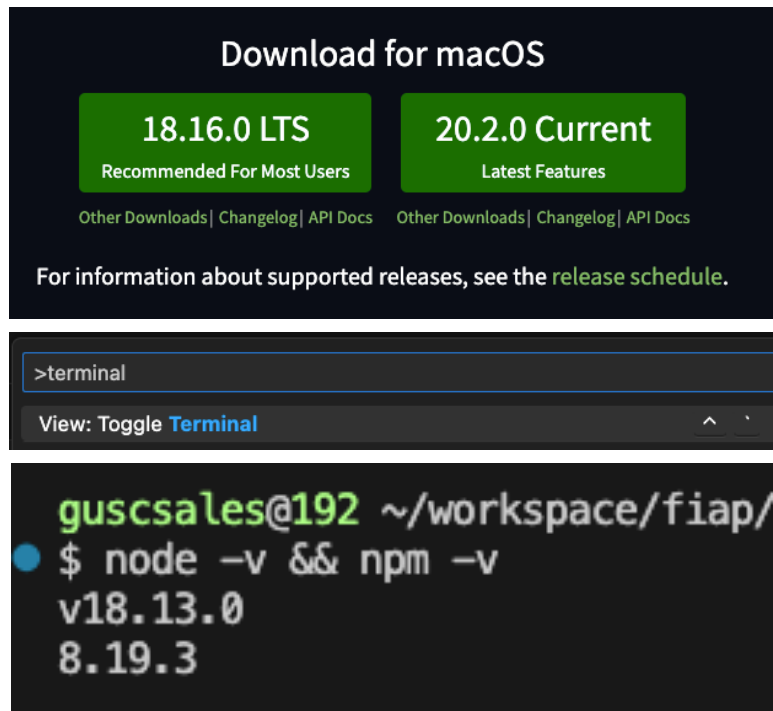
O **NPM** é o gerenciador de pacotes padrão para o ambiente Node.js, que possibilita rodar Javascript do lado do servidor.

Com o NPM é possível baixar e instalar blocos de códigos (ou pacotes) feitos por outras pessoas espalhadas no mundo, experimentando na prática o conceito de open source.

**Site NodeJS:** <https://nodejs.org/>

**Site NPM:** <https://www.npmjs.com/>

# NPM: Instalando o Node & NPM



Para fazer a instalação do NPM é muito simples, basta baixar o Node.js (recomendo na sua versão LTS que é a estável) e seguir o passo a passo da instalação.

Logo após abra seu VSCode, aperte **“F1”** e digite a palavra **“terminal”**.

Clique na opção **“View: Toggle terminal”**.

Dentro do terminal digite **“node -v && npm -v”** e de enter, se aparecerem as versões quer dizer que tudo foi instalado corretamente.

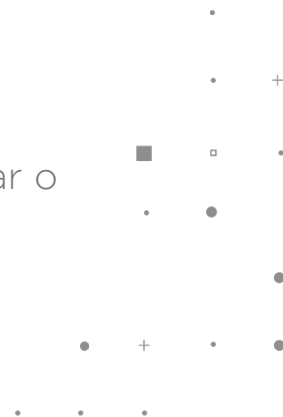
# NPM: package.json

```
npm init -y
```

O arquivo **package.json** é fundamental para os projetos feitos com Javascript.

Com ele podemos controlar os metadados do projeto, como nome, descrição, versão e outros, mas principalmente os pacotes de terceiros que são utilizados como dependências de projeto ou de desenvolvimento.

Para criar um package.json basta rodar o commando **“npm init -y”** no terminal dentro da pasta do projeto.



# NPM: package.json

O arquivo será criado com algumas propriedades:

- **name** – o nome do projeto
- **version** – a versão atual do projeto
- **description** – uma breve descrição sobre o projeto
- **main** – o arquivo de entrada (apenas para pacotes)
- **scripts** – comandos personalizados para rodarem no terminal
- **keywords** – palavras-chave para SEO no npm
- **author** – o autor do projeto ou pacote
- **license** – tipo de licença

```
{
  "name": "nome-do-projeto",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \\\"Error: no test specified\\\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}
```



# NPM: instalando pacotes novos

```
npm install remixicon
npm install --save-dev tailwindcss

# ou

npm i remixicon
npm i -D tailwindcss
```

```
{
  "name": "nome-do-projeto",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \\\"Error: no test specified\\\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "dependencies": {
    "remixicon": "^3.3.0"
  },
  "devDependencies": {
    "tailwindcss": "^3.3.2"
  }
}
```

Existem dois principais tipos de dependências: as **comuns de projeto (dependencies)** utilizadas no navegador e as de desenvolvimento

**(devDependencies)**, focadas em ajudar no processo de criação do projeto. Essas dependências ficam alocadas na pasta chamada **“node\_modules”**.

Para instalar utilize:

- **npm install <nome\_do\_pacote>**
- **npm install --save-dev <nome\_do\_pacote>**

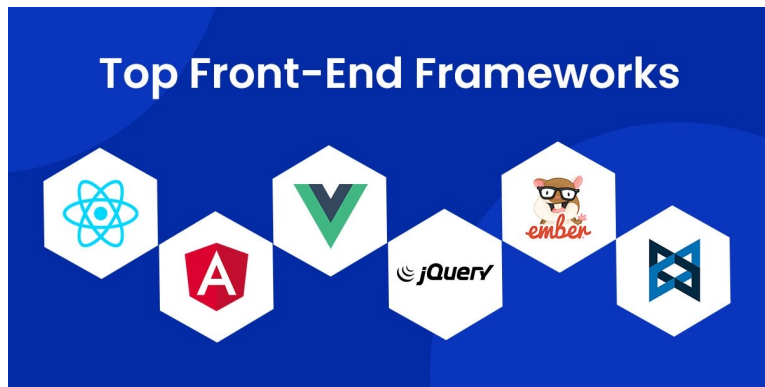
Para instalar pacotes que já estão dentro das dependências utilize apenas:

- **npm install**

2

# Introdução ao framework **TailwindCSS**

# Antes, o que são frameworks?



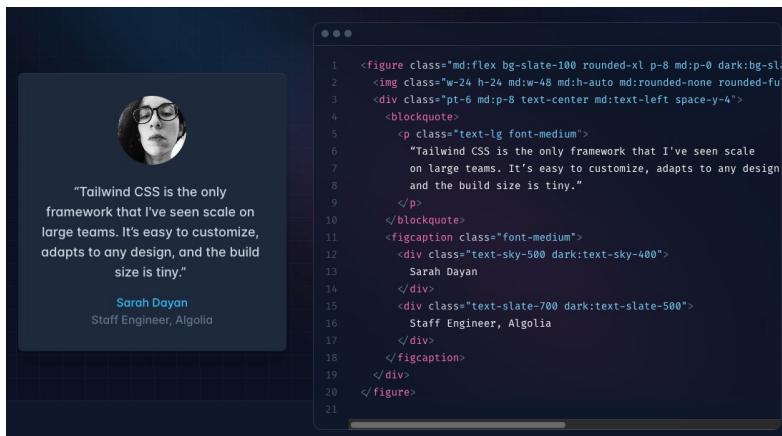
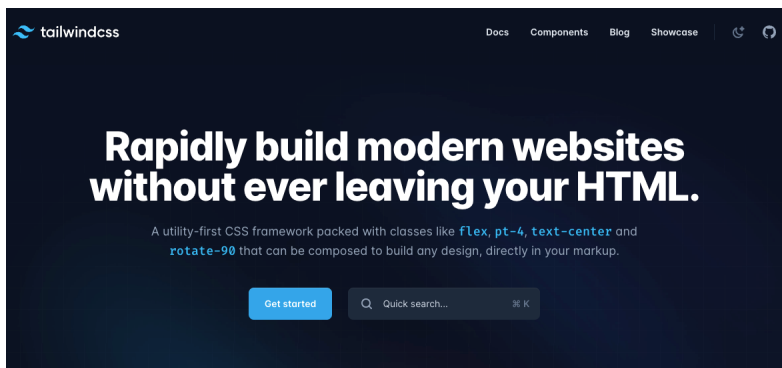
Frameworks são conjuntos de ferramentas, bibliotecas e padrões de código que facilitam a criação de projetos, seja web ou qualquer outro tipo.

Existem muitos frameworks famosos na internet como o React, Next.js e o

**TailwindCSS** (esses são apenas alguns exemplos de vários).

Com esses frameworks ganhamos eficiência no desenvolvimento, padrões bem definidos e ajuda da comunidade de programação.

# TailwindCSS: crie sites sem sair do seu HTML



O **TailwindCSS** é um dos mais famosos frameworks de CSS do mundo. Ele fornece várias classes utilitárias pré definidas que abstraem o CSS, sendo assim possível montar todo o site utilizando essas classes.

O framework conta com um sistema de cores, espaçamentos, grid, breakpoints já definidos e totalmente customizáveis.

Ele é muito conhecido por gerar produtividade e flexibilidade para os projetos.

# TailwindCSS: vantagens e desvantagens

## Vantagens

Produtividade

Personalização

Responsividade

Manutenção

Componentização

## Desvantagens

Curva de aprendizado

Complexidade no HTML

**Dica: sempre preste atenção nas vantagens e desvantagens de cada framework, não existe “bala de prata” 😊**

3

# Instalação e configuração do **TailwindCSS**

# TailwindCSS: instalação via NPM

```
npm install -D tailwindcss  
npx tailwindcss init
```

É possível instalar através do **Tailwind CLI**, instalando o pacote **tailwindcss** como dependência de desenvolvimento e logo após executar o comando **tailwindcss init** no terminal.

O arquivo **tailwind.config.js** será criado na raiz do projeto.

# TailwindCSS: arquivo de configuração

```
/** @type {import('tailwindcss').Config} */
module.exports = {
  content: ['src/**/*.html', 'src/**/*.css', 'src/**/*.js'],
  jit: true,
  theme: {
    extend: {},
  },
  plugins: [],
};
```

Com o arquivo de configuração é possível personalizar o Tailwind em diversas partes utilizando ao **extends**, especificar quais arquivos fazem parte do conteúdo com o **content** e também ativar o **JIT (Just In Time)** no qual apenas as classes utilizadas no código serão compiladas no CSS final.



# TailwindCSS: diretivas principais e normalização

```
@tailwind base;  
@tailwind components;  
@tailwind utilities;
```

É importante importar o arquivo base e de utilidades do tailwind no arquivo global de CSS. Os componentes também podem ser importados, mas é opcional.



# TailwindCSS: comando de execução

```
{
  "name": "projeto-tailwindcss",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "watch-tailwind": "tailwindcss -i
./src/css/tailwind.css -o
./public/css/tailwind.css --watch"
  },
  "keywords": [],
  "author": "",
  "license": "ISC",
  "devDependencies": {
    "tailwindcss": "^3.3.2"
  }
}
```

```
npm run watch-tailwind
```

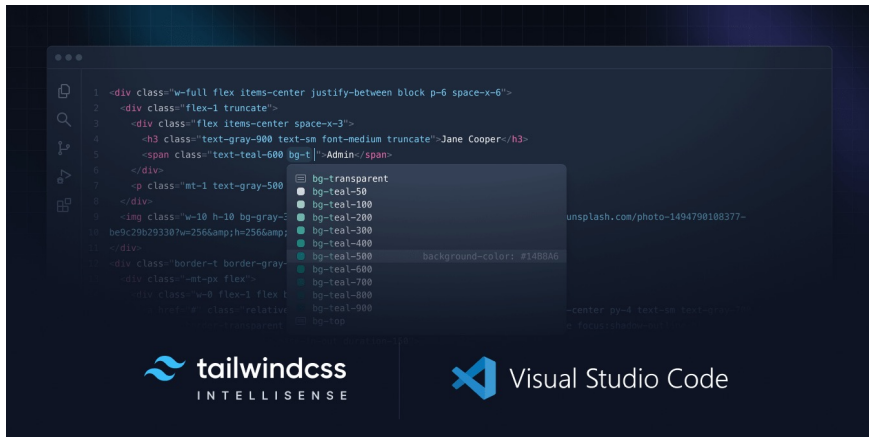
Para executar o tailwind basta criar um comando no **package.json** com o arquivo de input e o arquivo de output.

O commando será **watch-tailwind** com **tailwindcss -i ./src/css/tailwind.css -o ./public/css/tailwind.css -watch**

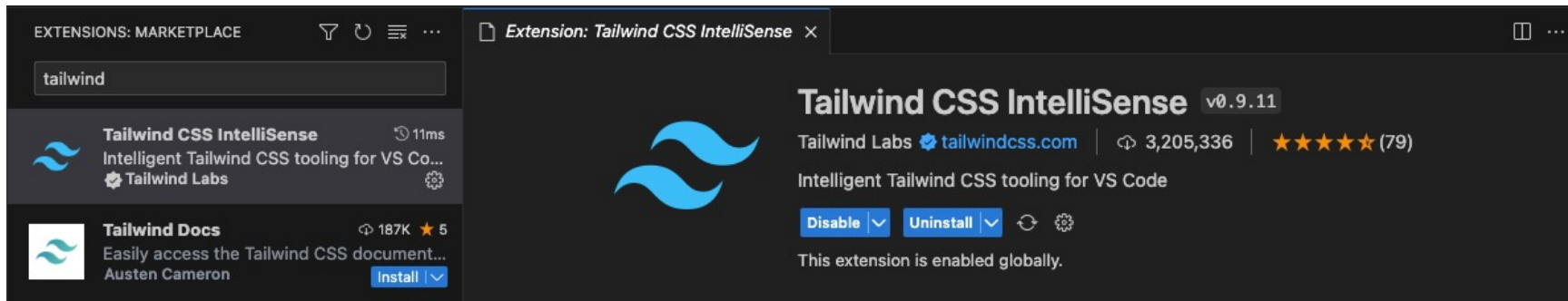
Importe o arquivo

“/public/css/tailwind.css” no seu HTML.

# TailwindCSS: extensão para VSCode



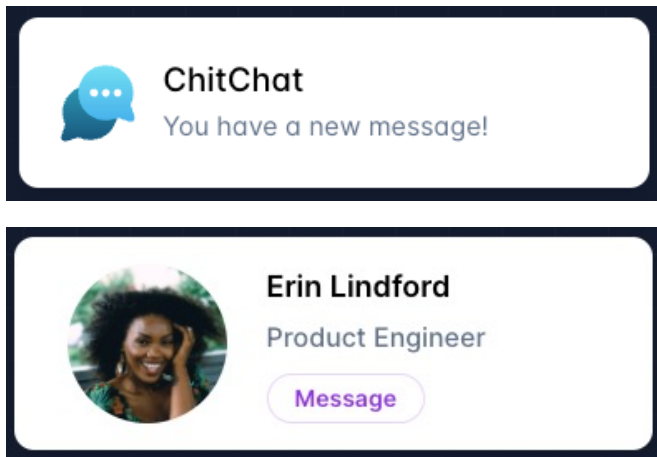
Existe uma extensão para VSCode que compila todas as classes do TailwindCSS, inclusive as personalizada.



4

# Como utilizar o **TailwindCSS**

# TailwindCSS: fundamentos



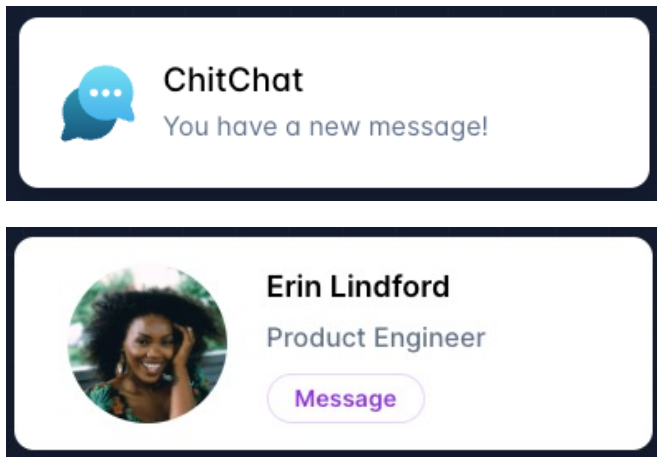
Podemos fazer esse mesmo componente utilizando as classes tradicionais de CSS e regras para essas classes.

Já no Tailwind tudo isso pode ser abstraído com as classes fundamental do Tailwind.

Imagine que todas as propriedades de CSS foram abstraídas em classes e você pode combinar todas essas classes.



# TailwindCSS: fundamentos



Por exemplo, **padding: 24px;** no CSS pode ser utilizado como **p-6** em formato de classe.



# TailwindCSS: classes de espaçamento

Name	Size	Pixels	
0	0px	0px	
px	1px	1px	
0.5	0.125rem	2px	
1	0.25rem	4px	
1.5	0.375rem	6px	
2	0.5rem	8px	
2.5	0.625rem	10px	
3	0.75rem	12px	
3.5	0.875rem	14px	
4	1rem	16px	
5	1.25rem	20px	
6	1.5rem	24px	
7	1.75rem	28px	
8	2rem	32px	

- **m** - margem geral
- **mx/my** - margem horizontal / vertical
- **mt/mb/ml/mr** - margem de cada lado
- **p** - padding geral
- **px/py** - padding horizontal / vertical
- **pt/pb/pl/pr** - padding de cada lado



# TailwindCSS: classes de largura e altura



- **w** - largura
- **h** - altura
- **min-w/min-h** - largura / altura mínima
- **max-w/max-h** - largura / altura máxima





# TailwindCSS: classes de cores



O TailwindCSS conta com uma bonita e grande gama de cores, cada cor tem um nome e podemos utilizar juntamente com a classe referente a propriedade CSS, por exemplo:

- **text-nome\_da\_cor** - largura
- **bg-nome\_da\_cor** - altura
- **border-nome\_da\_cor** - largura / altura mínima

Tabela de cores completa:

<https://tailwindcss.com/docs/customizing-colors>

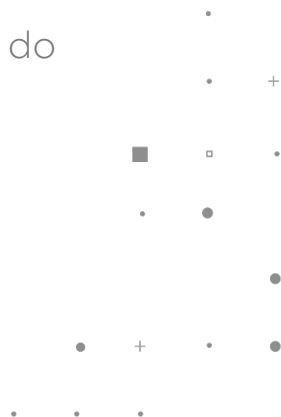
# TailwindCSS: tipografia

Class	Properties
<code>text-xs</code>	<code>font-size: 0.75rem; /* 12px */</code> <code>line-height: 1rem; /* 16px */</code>
<code>text-sm</code>	<code>font-size: 0.875rem; /* 14px */</code> <code>line-height: 1.25rem; /* 20px */</code>
<code>text-base</code>	<code>font-size: 1rem; /* 16px */</code> <code>line-height: 1.5rem; /* 24px */</code>
<code>text-lg</code>	<code>font-size: 1.125rem; /* 18px */</code> <code>line-height: 1.75rem; /* 28px */</code>
<code>text-xl</code>	<code>font-size: 1.25rem;</code> <code>line-height: 1.75rem;</code>

Class	Properties
<code>font-sans</code>	<code>font-family: ui-sans-serif,</code>
<code>font-serif</code>	<code>font-family: ui-serif, Geo</code>
<code>font-mono</code>	<code>font-family: ui-monospace,</code>

Class	Properties
<code>font-thin</code>	<code>font-weight: 100;</code>
<code>font-extralight</code>	<code>font-weight: 200;</code>
<code>font-light</code>	<code>font-weight: 300;</code>
<code>font-normal</code>	<code>font-weight: 400;</code>
<code>font-medium</code>	<code>font-weight: 500;</code>
<code>font-semibold</code>	<code>font-weight: 600;</code>
<code>font-bold</code>	<code>font-weight: 700;</code>
<code>font-extrabold</code>	<code>font-weight: 800;</code>
<code>font-black</code>	<code>font-weight: 900;</code>


- **text-tamanho** – define o tamanho da fonte
- **font-expressura** – define a expressura da fonte
- **text-alinhamento** – alinhamento do texto



# TailwindCSS: layout e posicionamento

**Grid**

Use ``grid`` to create a grid container.



```
<div class="grid gap-4 grid-cols-3 grid-rows-3">
  <!-- ... -->
</div>
```

- **flex** – define o container como flex
- **grid** – define o container como grid
- **grid-cols-número** – define o número de colunas
- **gap-numero** – define o espaçamento entre colunas
- **justify-center** – igual ao justify-content: center;
- **items-center** – igual ao align-items: center;

## TailwindCSS: valores arbitrários

```
<div class="m-[10px] p-[20px]">
  Olá mundo!
</div>
```

É possível utilizar valores customizados para praticamente todas as classes do tailwind, basta colocar o nome da classe e o valor entre **colchetes ([ ])**, por exemplo:

- **m-[10px]**
- **p-[20px]**



5

# Sugestões de **Exercícios**

# Exercícios para treinar NPM e TailwindCSS

Com o site criado nas últimas aulas

- Atualize seu site para utilizar o remix icons pelo NPM;
- Refatore o site para utilizar TailwindCSS e não mais CSS comum.

Bons estudos! 😊





# OBRIGADO

@guscsales - gsales.io

FIAP

Copyright © 2023 | Professor Gustavo Campos Sales

Todos os direitos reservados. Reprodução ou divulgação total ou parcial deste documento, é expressamente proibido sem consentimento formal, por escrito, do professor/autor.

