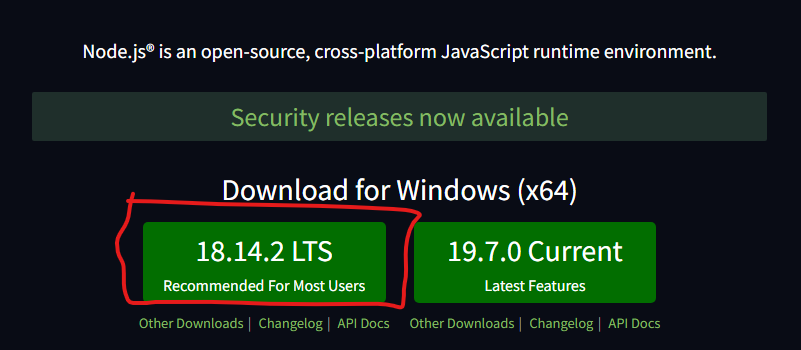
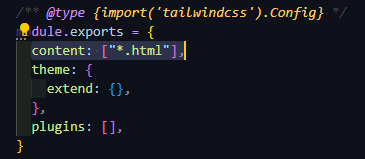
# **Установка Tailwind:**

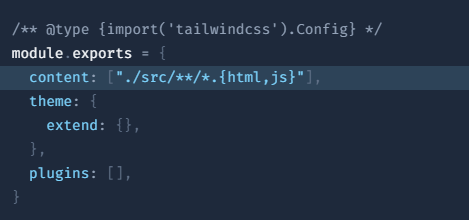
1. В проекте открыть терминал и вводит: npm init -y
2. Затем проверить установлено ли node.js на компьютере командой: node -v, если нет, то скачать и установить LTS версию:



1. Проверить npm: npm -v
2. Установить tailwindcss через npm: npm install -D tailwindcss
3. и создать свой tailwind.config.js файл: npx tailwindcss init
4. Добавить путь к файлам которые «работают» с tailwindCSS в файле tailwind.config.js file:



или же

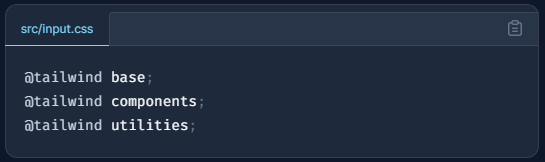


1. Создать файл «/css/input.css» и добавить туда @tailwind directives:

@tailwind base;

@tailwind components;

@tailwind utilities;



1. Надо запустить в терминале команду: npx tailwindcss -i ./css/input.css -o ./css/output.css –watch, чтоб «преобразовать команды выше в обычный css коды».
2. Чтоб не запускать эту команду каждый раз, в файле package.json вместо команды «test» запищим:

  "scripts": {

    "start": "npx tailwindcss -i ./css/input.css -o ./css/output.css --watch"

  },

В будущем вместо команды п.8 можно вводить: npm run start

1. Дальше, в файле «index.html» подключаем css файл «output.css» и всё!

**Плагин для работы с формами**

Он сбрасывает все стандартные стили элементов форм, и позволяет легко принимать к ним кастомные стили

1. Установка по команде: npm install -D @tailwindcss/forms
2. Затем подключать плагин в файле tailwind.config.js: require('@tailwindcss/forms'),



**Добавить свои цвета:**

1. В файле tailwind.config.js добавим свои объект с переменными:

   extend: {

      colors:{

        vanille: "var(--vanille)"

      }

    },



1. Затем в файле input.css: записываем значение для переменного:

*:root*{

    --vanille: #dfd6a7;

}

