

create script folder root script/file.js (any name)

1) npm install glob --save-dev

```
const fs = require("fs");
const path = require("path");

function sortCssFile(file) {
  const content = fs.readFileSync(file, "utf-8");
  const lines = content.split("\n");

  let sortedContent = [];
  let stack = [];

  function flushBlock(block) {
    if (block.lines.length > 0) {
      // ✅ Sort ascending by line length
      block.lines.sort((a, b) => a.trim().length - b.trim().length);
      block.sortedContent.push(...block.lines);
      block.lines = [];
    }
  }

  lines.forEach((line) => {
    const trimmedLine = line.trim();

    if (trimmedLine.endsWith("{")) {
      // New block
      const newBlock = { sortedContent: [line], lines: [] };
      stack.push(newBlock);
    } else if (trimmedLine.endsWith("}")) {
      // Close block
      const finishedBlock = stack.pop();
      flushBlock(finishedBlock);
      finishedBlock.sortedContent.push(line);

      if (stack.length > 0) {
        // Nested block (media query, keyframes etc.)
        stack[stack.length - 1].sortedContent.push(...finishedBlock.sortedContent);
      } else {
        // Top-level block
        sortedContent.push(...finishedBlock.sortedContent);
      }
    } else {
      if (stack.length > 0) {
        // Inside a block
        if (trimmedLine.includes(":") && trimmedLine.endsWith(";")) {
          stack[stack.length - 1].lines.push(line);
        } else {
          stack[stack.length - 1].sortedContent.push(line);
        }
      } else {
        // Outside any block
        sortedContent.push(line);
      }
    }
  });

  const finalContent = sortedContent.join("\n");
  fs.writeFileSync(file, finalContent, "utf-8");
  console.log(`✅ Sorted CSS in ${file}`);
}
```

```
// ---- main ----
const fileArg = process.argv[2];
if (!fileArg) {
  console.error("❌ Please provide a file path. Example: node scripts/sort-
css.js src/styles/Home.module.css");
  process.exit(1);
}

const resolvedPath = path.resolve(fileArg);
sortCssFile(resolvedPath);
```