4.2 Annex B: Guideline and structure for clean code

In this guideline, a few tips to structure your code will be described to make your code easier to read, share, and verify. It is important to write the code as clean and structured as possible to make it simple, concise, and understandable. This means that it would be good to follow a set of conventions, standards, and practices that make it easy for others to read and follow.

The code file should ideally contain file-level documentation at the top containing information of the author, description, licence and date.

- Use a naming convention.
    - Use descriptive names for variables, functions, classes, and other identifiers. A well-chosen name can convey the purpose of the entity, making the code more understandable. Avoid single-letter variable names or cryptic abbreviations.
    - The R community doesn't have an official naming convention, you can choose your own, as long as it is consistent throughout your code.

- Always load packages at the beginning
    - This will save time and also show others what packages are used beforehand.

- Retrieving data

  Always use relative file paths, never use absolute file paths: having the file path to your local drive (where you store your data) will mean it will not run on another computer. If data is publicly available let others know where to download it, if it is in the working directory use relative file paths. If not publicly available state this in the script.

- Use indentation and spacing
    - Good usage of whitespace between operators, commas, and around flow statements enhances code readability and makes the codebase look clean and organized.

- Naming your variables, functions and files
    - Names that communicate their intended use in code are key components of clean programming. By reading a name alone, a developer should be able to readily understand what the function or variable does or represents in their application.
    - This is pretty straightforward, but it's probably the most common and maybe the easiest one to forget. Easily the most frustrating thing for another developer looking at your code is seeing a variable with a misleading name or, worse, named with a single letter.
    - File names should be meaningful. Avoid using special characters in file names - stick with numbers, letters, -, and _.

- Commenting
  - Use comments sparingly, and when you do, make them meaningful.
- Other tips
  - Avoid having one big code script rather structure your work into different files (e.g. data (data retrieval and preparation), model, output, report) and source them.

For more inspiration and examples of clean code please go here:
The tidyverse style guide: https://style.tidyverse.org/index.html