# **(Chapter 16) A Comment on Comments**

▼ What are Three Things I learned today

1. **Before** – I've had experience previously, so I added header comments to my project so that I could simply explain how my code functions to anyone who was interested.

**After** - I’ve learned that the information in my header comments should enable any programmer to utilize my code without having to understand it, but the information in my in-line comments should help the following developer improve or extend it.

2. **Before** - I've previously made mistakes or swapped the comments for the two functions in my code's comments.

**After**  - I’ve learned that I must add pertinent comments to the code that describe what the code is meant to do.

3. **Before** – I've tried commenting on every feature and may have gone overboard, but thus far it's been successful.

**After -** I’ve learned on commenting may sometimes go too far and make sure my code is not obscured but is clarified by my comments.

# **(Chapter 17) Comment Only What the Code Cannot Say**

▼ What are Three Things I learned today

1**. Before** – I've had experience adding all comments to the functions I used in my code.

**After** - I’ve learned that each comment should provide the reader with something of value; otherwise, it is unnecessary and should be deleted or modified.

2. **Before** – I've used git hub before, but I have no idea what to write in the commit message. I only put “first commit”, “second commit” and etc.

**After** - I now understand the use of comments in version control systems, which, when combined with version control systems, may aid in tracking changes and comprehending the history of a project.

3. **Before** - I've had this happen before on a project where I neglected to include comments, and when I ran into problems during testing, I had to spend time reading my code to figure out where the problems were or what caused them.

**After**  - I've learnt that comments have an influence on code maintenance since they can aid future maintainers in understanding the code and making the appropriate modifications, which will facilitate and speed up maintenance.