**LOUIE JAY A. CENIZA APRIL 4, 2024**

**PACKING INFORMATION INTO NAMES (CHAPTER 2)**

**Choose Specific Words.** As the name suggests, it's crucial to choose your words wisely when naming variables or adding comments. Precision is key, names and comments should accurately reflect the intended purpose or functionality of the code. Additionally, consistency is important; if you name something, ensure that its name aligns with its function or purpose throughout the codebase.

**Avoid Generic Names Like tmp and retval.** It refers to the naming of your variables or others, as always you need to find a word that best describes what it is, not just random words just to say it has a use on it.

**Prefer Concrete Names over Abstract Names.** Concrete naming focuses on choosing names that are specific, descriptive, and tangible, while abstract naming relies on more general and ambiguous terms. Concrete names enhance the clarity, readability, and maintainability of code, whereas abstract names can lead to confusion and misunderstanding.

**Attaching Extra Information to a Name.** it's important to attach extra information to a name, when necessary, especially to specify the type of data it represents. Adding this extra information helps clarify the purpose and usage of the variable, making the code more understandable and reducing the risk of errors.

**How Long Should a Name Be?.** In this lesson When determining the ideal length for names in code, several considerations come into play. Shorter names may suffice for elements with a narrower scope, such as local variables, where context is limited and clarity can still be maintained.

**Use Name Formatting to Convey Meaning.** In this lesson it discusses how formatting conventions, such as capitalization, underscores, or other patterns, can be used to convey additional meaning or context in names within code. It emphasizes the importance of consistent and meaningful formatting to enhance readability and comprehension.