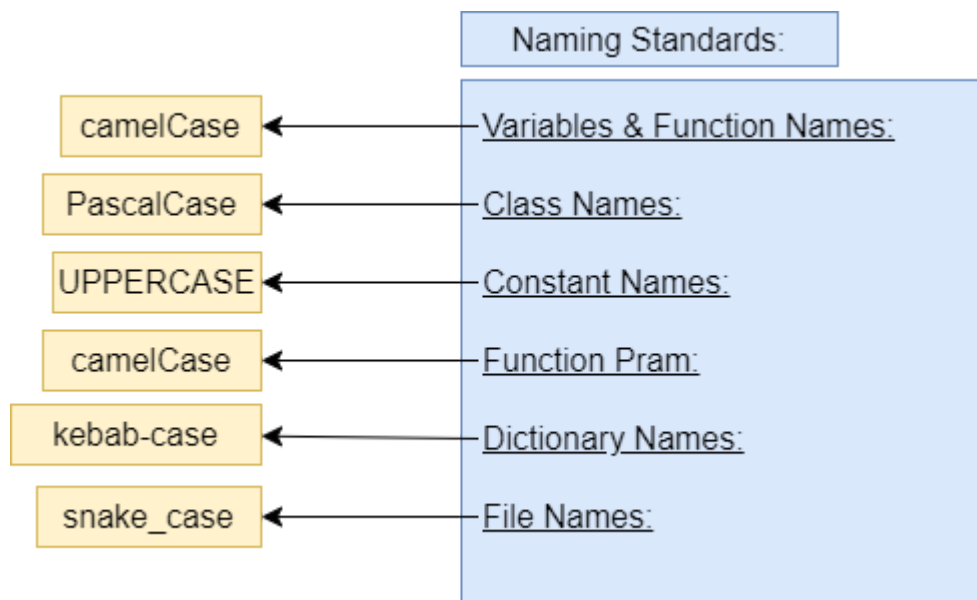# Coding Standards Diagram

- Naming conventions
    - Variable and function names have to be meaningful so that anyone can understand the reason of using it
    - Variables should be named using camel case lettering starting with small letters (e.g localVariable)
    - Constants should use capital letters only (e.g CONSTANTVAR)
    - Avoid the use of digits in variable names
    - Names of functions and function parameters should be written in camel case lettering starting with small letters
    - Name of a function must clearly and briefly describe the reason of using the function
    - File names should be in snake case

| | Naming Standards: |
|---|---|
| camelCase | Variables & Function Names: |
| PascalCase | Class Names: |
| UPPERCASE | Constant Names: |
| camelCase | Function Pram: |
| kebab-case | Dictionary Names: |
| snake_case | File Names: |

- Indentation
    - Use tab indentation
    - Each nested block should be properly indented and spaced

- ○ All braces should start from a new line and the code following the end of braces should also start from a new line
- Header files
  - ○ Each module has it's own corresponding header file that describes its interface. (i.e no mixing of interfaces into one header file)
  - ○ Header files will not contain any code, just declarations
  - ○ Header files will not be used to initialise data
- Subroutine headers
  - ○ Each subroutine in the file should have a description before the beginning of its declaration
- Comments
  - ○ All comments should comply to Doxygen commenting syntax
  - ○ Comments should be presented in paragraph form prior to a block of code
  - ○ Indent block comments on the same level as the block being described
  - ○ Although block comments will be the primary form of commenting, use inline comments where appropriate
  - ○ An inline comment shall accompany each definition of a variable
- Git standards
  - ○ No pushing directly to the master branch
  - ○ There will be branches for each design pattern
  - ○ Code will be merged into the master branch at a later stage
  - ○ Use clear and concise commit messages

Git Standards

Branch Naming Convention ⟶ Feature Name

Commit Messages ⟶ Explain the commit and changes(e.g., "Add Feature")

Pull requests ⟶ Include Tests and Documentation

Code Review ⟶ Peer review with Comments

Merging Strategy ⟶ Merge with rebase(merging branches together before Merge to Master)