1. **Meaningful Variable Names:**

Use descriptive names for variables that convey their purpose.

2. **Functions Should Do One Thing:**

Functions should have a single responsibility and perform only one logical operation.

3. **Comments are a Last Resort:**

Strive to write self-explanatory code. Use comments sparingly and only when necessary.

4. **Avoid Magic Numbers:**

Replace hardcoded numeric values with named constants to improve code readability.

5. **Consistent Formatting:**

Maintain a consistent code style and formatting throughout the project.

6. **Keep Functions Small:**

Aim for short, focused functions that are easy to understand and test.

7. **Avoid Deep Nesting:**

Limit the depth of nested structures to enhance code readability.

8. **Error Handling without Magic Values:**

Use exceptions for error handling instead of returning magic values.

9. **Unit Tests for Every Function:**

Write unit tests to ensure each function behaves as expected.

10. **Refactor Regularly:**

Keep code clean by refactoring regularly. If you see a better way to structure your code, don't hesitate to improve it.