Clean Code Development (CCD) Cheat Sheet

1. Meaningful Naming Conventions

- Use descriptive function and variable names that explain their purpose.
- Avoid single-letter variable names except for loop indices.
- Example:
  Bad: x = 5
  Good: max_retries = 5

2. Function Length & Single Responsibility

- Keep functions small; ideally under 20 lines.
- Each function should do ONE thing only.
- Example:
  Bad: def process_data(): # Does too many things
  Good: def clean_data(): # Only cleans data
        def format_data(): # Only formats data

3. Comments vs. Self-Documenting Code

- Avoid unnecessary comments; write clear code instead.
- Use comments only for complex logic.
- Example:
  Bad: # Increment counter
        counter += 1
  Good: counter += 1 # This does not need a comment

4. Error Handling & Exception Safety

- Always handle errors gracefully to prevent crashes.
- Example:

```
try:
    file = open('data.json')
except FileNotFoundError:
    print('Error: File not found.')
```

5. Code Formatting & Linting

- Follow PEP-8 for Python code formatting.
- Use linters like Flake8 or Black.
- Example:

  Bad: def add(a,b): return a+b

  Good: def add(a, b):

        return a + b

6. DRY (Do not Repeat Yourself)

- Avoid duplicate code by using functions and classes.
- Example:

  Bad: print(user.name)

      print(user.email)

  Good: def print_user_info(user):

        print(user.name)

        print(user.email)

7. KISS (Keep It Simple, Stupid) & YAGNI (You Are not Gonna Need It)

- Do not overcomplicate code; keep it simple.
- Avoid premature optimizations and unnecessary features.
- Example:

  Bad: Implementing an entire class for one small function.

  Good: Using a simple function instead.

8. Writing Readable Conditionals

- Simplify conditionals for readability.
- Example:

  Bad: if user.age > 18 and user.verified and not user.banned:

  Good: if is_valid_user(user):