Gherkin Best Practices:

- Use Keywords to maintain a standard and promote readability.

    Given: the precondition or setup, it must be first
    When: the test **Action** that must performed
    Then: the **Assertion** aka the expected outcome
    And, But: Wildcards

- Features and scenarios should always be independent.

    No feature or scenario should rely on one another.

- Keep your steps under 100 characters, but be as descriptive as possible.

- If multiple scenarios have the same setup, move that information into a *Background*.

- Link your requirements to the scenarios.

    Helps keep track of what scenarios are covering requirements.

    If a scenario fails during the test run, the requirements will be shown in the scenario step for reference.

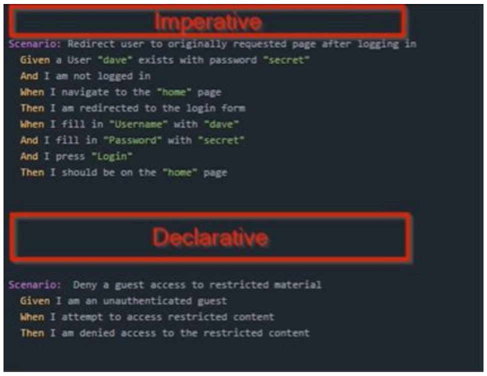
    Add them to the end of the specified scenario behind a colon, using the requirement ID. OR use a comment using a # above the scenario.

- Keep your data tables organized by formatting for Readability.

    Use cohesive spacing between table elements, format in grid.
    Use lowercase letters in tables.

- Write Declarative steps.

    Describe what a user does, not how they do it.
    It contains all the needed information and no more.
    It will be more easily readable.



Imperative

```
Scenario: Redirect user to originally requested page after logging in
  Given a User "dave" exists with password "secret"
  And I am not logged in
  When I navigate to the "home" page
  Then I am redirected to the login form
  When I fill in "Username" with "dave"
  And I fill in "Password" with "secret"
  And I press "Login"
  Then I should be on the "home" page
```

Declarative

```
Scenario:  Deny a guest access to restricted material
  Given I am an unauthenticated guest
  When I attempt to access restricted content
  Then I am denied access to the restricted content
```

- Write steps in a Modular fashion.

  Make your steps reusable, if possible.
  Use consistent language.
  Use ‘parameters’ to make steps more clear.
  Use lowercase letters wherever possible.

- Never use Microsoft Word to write your steps.

  Microsoft uses “Smart Quotes” which are not compatible with other applications.

- Use Comments where necessary for clarification.

  Keep comments at the top of the scenario or feature.
  Use #! to signify a comment that should be removed when the specified issue is remedied.
  Use # to signify a comment that should remain in place.