1. Define a class called `vehicle` with the following properties and methods:
    - Properties:
        - `wheels` (int)
        - `fuel` (double): initially 0 liter
        - `milage` (double): initially 0 mile
    - Methods:
        - `wheels` (): returns the number of wheels
        - `fuel` (): returns available fuel
        - `milage` (): returns total milage run
        - `petrol` (): add fuel according to the driver’s need
        - `drive` ()
2. Implement 3 subclasses of `vehicle`:
    - `sedan`: override
        - `drive` (): runs for 5 miles and spends 2L of fuel if fuel is available
    - `motorcycle`: override
        - `drive` (): runs for 1.5 miles and spends 0.5L of fuel if fuel is available
    - `SUV`: override
        - `drive` (): runs for 4 miles and spends 2.5L of fuel if fuel is available
3. Define a class called `color` with the following properties:
    - Properties:
        - `name` (string)
    - Methods:
        - `showcolor` ()
4. Implement 3 subclasses of `color`:
    - `red`, `blue`, and `green`
5. Create instances of `vehicle` (e.g., `sedan`, `motorcycle`, `SUV`) of different `color` (e.g., `red`, `blue`, `green`) by interacting with necessary UI components from home activity. Add UI components to interact with `drive` () and `petrol` ().

6. Develop unit tests for each of the `vehicle` subclasses and the `color` subclasses to ensure that their methods function correctly.
7. Design UI tests to ensure that the user interface of the application behaves as expected when interacting with shapes and colors.
8. Make sure to add CI feature while doing this project.