

# Gherkin Cheat Sheet

## Gherkin Keywords

Feature
Scenario
Given, When, Then, And, But
*
Background
Scenario Outline
Examples

## Extra Keywords

# (Comments)
@ (tag)
(Tabular Data)
""" (Doc Strings)

## Feature (.feature) file structure

@tag @tag

**Feature:** <name of the feature>

*description*

*Background:*

*Given*

*And*

*When*

*And*

@tag @tag

**Scenario:** <name of the scenario>

**Given**

*And*

*But*

**When**

*And*

*But*

**Then**

*And*

*But*

Background step(s)

Additional contextual information. Run before each scenario

Describe an initial context

Describe actions

Describe observable expected outcome

When you have a complex business rule with severable variable inputs or outputs you might end up creating several scenarios that only differ by their values. We can simplify this with a Scenario Outline:

```
Scenario Outline: eating
  Given there are <start> cucumbers
  When I eat <eat> cucumbers
  Then I should have <left> cucumbers
```

Examples:

start	eat	left
12	5	7
20	5	15

The Scenario Outline will execute once for every row in the Examples table.

*optional*

Data tables are used for specifying a larger piece of data in tabular format

Given the following car models exists:

name	available cars
Sports car	3
Sedan	2
hatchback	4

In above example data is passed to Given step to create 2 records in table with 2 columns , *name* and *available\_cars*

When user fills the rent a car form:

Model	Sports car
Pickup Location	1st Avenue, NY

In above example data is passed to When step to fill a form with 2 fields, *Model* and *Pickup Location*

## Tips

- One feature per well named file. Please keep the features focused
- Give meaningful name to Features and Scenarios
- Make Scenarios Independent and Deterministic
- Cover Happy Paths, Error Paths and Edge Case scenarios. Give lots of clear and concrete examples
- Revisit, Refactor, and Improve Your Scenarios and Steps. Look for opportunities to generalize your steps and reuse them
- Declarative or Imperative, choose wisely
- Tags are a great way to organize your features and scenarios Using tags let you keep a feature's scenarios together structurally, but run them separately