1.Install Gauge

For Mac
We would recommend using brew for a very simple straightforward installation using this command.
$ brew install gauge

For Windows
https://gauge.org/get-started.html

2.Install core plugins

$ gauge install java
$ gauge install spectacle

3.Install IntelliJ

Download the IntelliJ software Community edition from https://www.jetbrains.com/idea/download/

4.Gauge plugin for IntelliJ

1. Open the Setting/Preferences dialog from IntelliJ Menu
2. In the left-hand pane, select Plugins
3. On the Plugins page, click on Install JetBrains plugin or Browse repositories button
4. In the dialog that opens, search for Gauge
5. Select, install and restart IntelliJ

5a.Create New Gauge Project in IntelliJ

1. File -> New Project.
2. Choose 'Gauge'
3. Choose the project location and java sdk
4. Finish
5. Download selenium-java 2.5.3 from http://selenium-release.storage.googleapis.com/index.html?path=2.53/. Unzip the download and save all the jar files at one location in your system e.g. D:\selenium_jars (this folder will contain all the jars from the selenium folder)
6. To add the above selenium libraries to the IntelliJ project follow the steps given below
    a. Right click on the project name and select ‘Open Module Settings’ option
    b. In the dialog box that open select ‘Library’ section
    c. Select the folder from Step 5 where all the jar files have been stored
    d. Click on Apply, then click on OK

OR

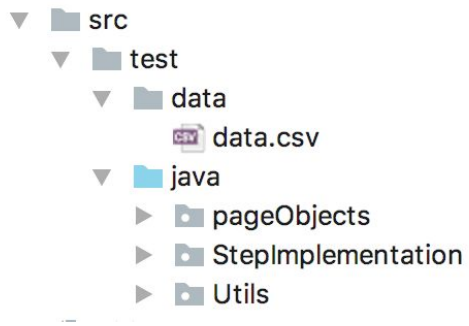
5b.Create New Project in IntelliJ (Gauge with Maven)

1. File -> New Project
2. Choose Maven
3. Select Create from Archetype
4. Select the gauge archetype - com.thoughtworks.gauge.maven
5. If the com.thoughtworks.gauge.maven archetype is not added select Add Archetype
    a. Enter GroupId: com.thoughtworks.gauge.maven
    b. Enter ArtifactId: gauge-archetype-java
    c. Enter Version: 1.0.1 or the latest version number
6. Enter the groupId and artifactId for your project.
7. Enter Project Name and finish
8. Enable auto-import for the project

Add the following dependencies in the pom.xml file that is created

```
<dependencies>
  <dependency>
    <groupId>com.thoughtworks.gauge</groupId>
    <artifactId>gauge-java</artifactId>
    <version>0.6.0</version>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>4.12</version>
    <scope>test</scope>
  </dependency>
  <dependency>
    <groupId>org.seleniumhq.selenium</groupId>
    <artifactId>selenium-java</artifactId>
    <version>3.6.0</version>
  </dependency>
</dependencies>
```

7.Create the following folder structure



8.Context and Tear down

Context
Delete project
==========
These are context steps

* User is logged in as "mike"
* Navigate to the project page
Delete single project
---
* Delete the "example" project
* Ensure "example" project has been deleted

Teardown

These are teardown steps

* Logout user "mike"
* Delete user "mike"

9.Params and Special params

Simple Params
* Create a “gauge-java” project
* Write “100” line specification

Table Params
```
    |id| name    |
    |--|---------|
    |1 |vishnu   |
    |2 |prateek  |
    |3 |navaneeth|
```
Scenario
---

* Say "hello" to <name>

Special params
* Step that takes a table <table:data.csv>
* Check if the following users exist <table : /Users/john/work/users.csv>

Sample csv file:

```
Id,Name
1,The Way to Go On
2,Ivo Jay Balbaert
```

10.**Concept**

Create a .cpt file under specs directory with the concept definition.
The concept definition contains the 2 parts:

- Concept header
- Steps

Login as user <username> and <password>

* Enter log in details <username> and password <password>
* Submit the form
* User should be logged in with message "MY ACCOUNT"

11.**Tags**

Login specification
==========
Tags: login, admin, user-abc

Successful login scenario

Tags: login-success, admin

12.**Execution (Create project using step 5a to execute gauge command line commands)**

$ gauge run specs/login_test.spec or **$ gauge specs/**

$ gauge run specs/login_test.spec or **$ gauge specs/**

Single scenario execution
$ gauge run specs/login_test.spec:3

Table Driven Execution
$ gauge run --table-rows "1" specs/hello.spec

$ gauge run --table-rows "1-3" specs/hello.spec

Tagged Execution
$ gauge run --tags "search & admin" SPEC_FILE_NAME

Parallel Execution
$ gauge run --parallel specs or $ gauge run --p specs

$ gauge run --parallel -n=4 specs

$ gauge run -n=4 --strategy="lazy" specs or gauge run -n=4 specs

$ gauge -n=4 --strategy="eager" specs