

Serenity-

Serenity is bdd pattern.

You can write own step def.

They support rest assured, junit, cucumber etc and many other tools.

Go to this site and check more details-

[Welcome | Serenity BDD Users Manual \(serenity-bdd.github.io\)](http://serenity-bdd.github.io)

Serenity does not support testng.

Serenity nicely maps requirement documentation to the report.

First add in pom all the dependencies-

```

103
104      <!--dependencies for serenity report -->
105      <dependency>
106          <groupId>net.serenity-bdd</groupId>
107          <artifactId>serenity-core</artifactId>
108          <version>${serenity.version}</version>
109          <scope>test</scope>
110      </dependency>
111
112      <!--dependencies for serenity report -->
113      <dependency>
114          <groupId>net.serenity-bdd</groupId>
115          <artifactId>serenity-junit</artifactId>
116          <version>${serenity.version}</version>
117      </dependency>
118
119      <!--dependencies for serenity report -->
120      <dependency>
121          <groupId>net.serenity-bdd</groupId>
122          <artifactId>serenity-ensure</artifactId>
123          <version>${serenity.version}</version>
124      </dependency>
125
126      <!--dependencies for serenity report -->
127      <dependency>
128          <groupId>net.serenity-bdd</groupId>
129          <artifactId>serenity-cucumber6</artifactId>
130          <version>${serenity.cucumber.version}</version>
131      </dependency>
132

```

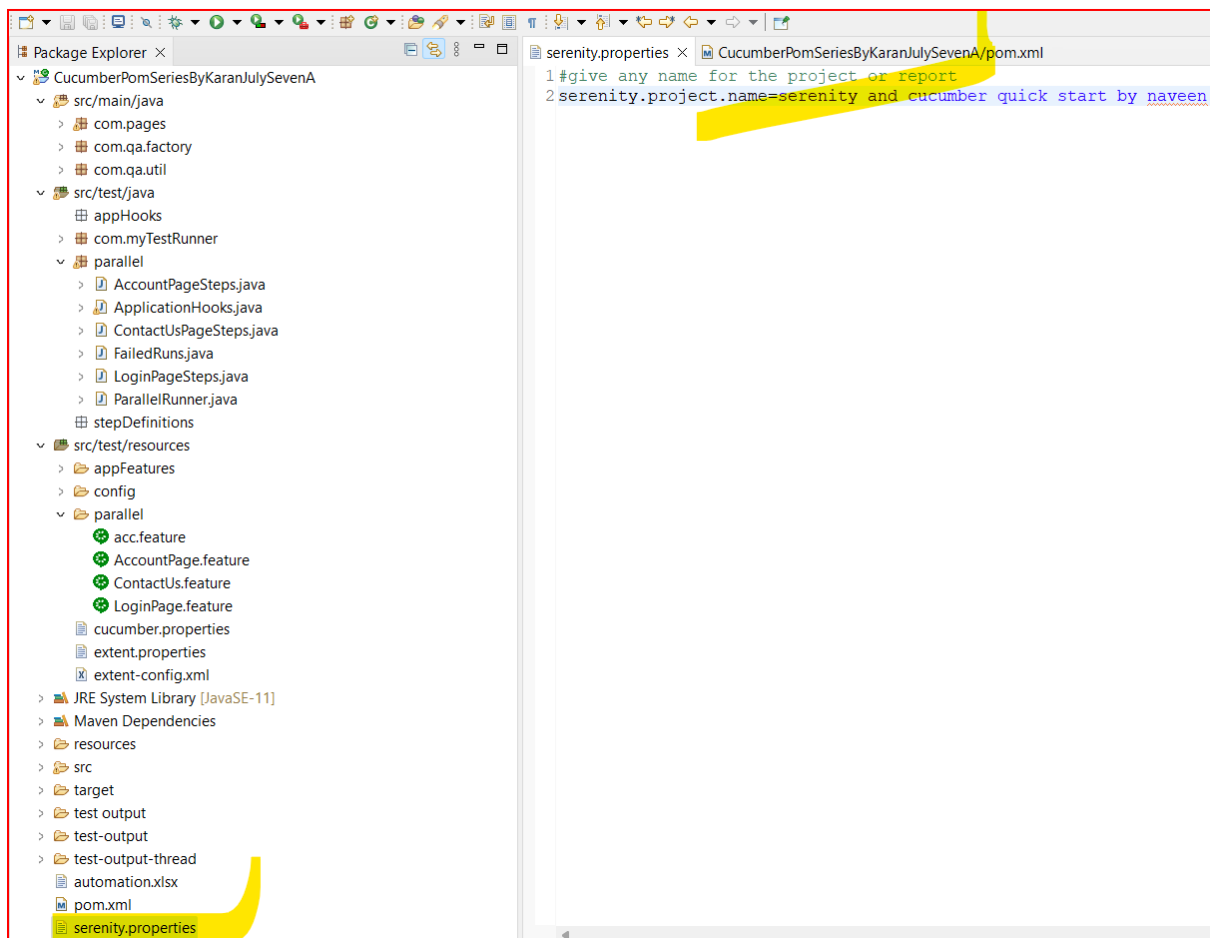
Add the versions in properties-

```

15
16<properties>
17    <!--all properties and the version numbers are mentioned within properties -->
18
19    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
20    <java.version>11</java.version>
21    <junit.version>4.13.1</junit.version>
22    <cucumber.version>6.9.0</cucumber.version>
23    <maven.compiler.version>3.8.1</maven.compiler.version>
24    <maven.surefire.version>2.22.2</maven.surefire.version>
25    <serenity.version>2.3.12</serenity.version>
26    <serenity.maven.version>2.3.12</serenity.maven.version>
27    <serenity.cucumber.version>2.3.12</serenity.cucumber.version>
28
29</properties>

```

Create this property file at project level-



We add surefire plugin to run test cases using maven.

We add the serenity runner file path inside the surefire plugin-

```

139
140     <plugin>
141       <groupId>org.apache.maven.plugins</groupId>
142       <artifactId>maven-compiler-plugin</artifactId>
143       <version>${maven.compiler.version}</version>
144     <configuration>
145       <encoding>UTF-8</encoding>
146       <source>${java.version}</source>
147       <target>${java.version}</target>
148     </configuration>
149   </plugin>
150   <plugin>
151     <groupId>org.apache.maven.plugins</groupId>
152     <artifactId>maven-surefire-plugin</artifactId>
153     <version>${maven.surefire.version}</version>
154   <configuration>
155     <!-- to run with jenkins and to run using maven use the below includes -->
156     <includes>
157       <include>*/SerenityTestRunner.java</include>
158     </includes>
159     <parallel>methods</parallel>
160     <!--<threadCount>4</threadCount>-->
161     <useUnlimitedThreads>true</useUnlimitedThreads>
162   </configuration>
163 </plugin>
164

```

Add one more plugin for serenity report-

This will generate good serenity reports using serenity core. Goal is to aggregate all the features into one file and prepare an aggregate report.

```

203<!-- using serenity core it will generate serenity report by clubbing all the feature files. it will create aggregate
204 report. hence we are giving aggregate. -->
205
206<plugin>
207  <groupId>net.serenity-bdd.maven.plugins</groupId>
208  <artifactId>serenity-maven-plugin</artifactId>
209  <version>${serenity.maven.version}</version>
210  <configuration>
211    <tags>${tags}</tags>
212  </configuration>
213  <dependencies>
214    <dependency>
215      <groupId>net.serenity-bdd</groupId>
216      <artifactId>serenity-core</artifactId>
217      <version>${serenity.version}</version>
218    </dependency>
219  </dependencies>
220  <executions>
221    <execution>
222      <id>serenity-reports</id>
223      <phase>post-integration-test</phase>
224      <goals>
225        <goal>aggregate</goal>
226      </goals>
227    </execution>
228  </executions>
229 </plugin>
230

```

Once all settings done, force update the project always.

Run from cmd-

```
E:\Naveen Java Training\Cucumber\CucumberPomSeriesByKaranJulySevenA>mvn clean verify|
```

Go to the project location and run the above command.

Build success-

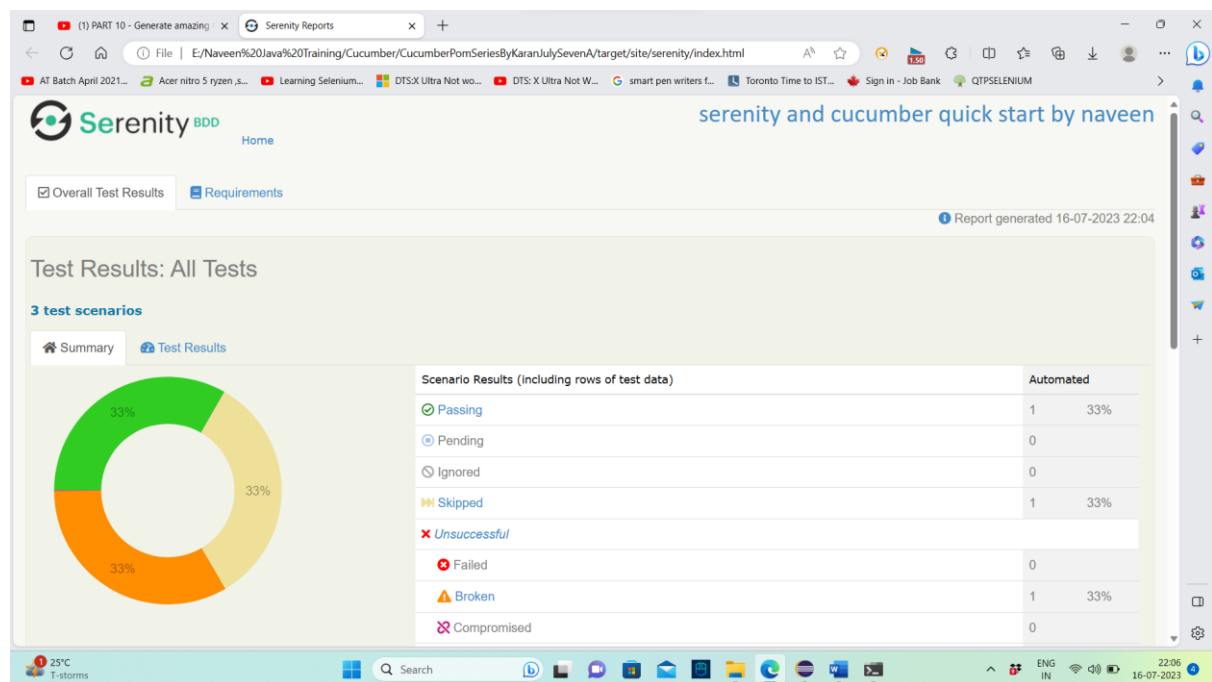
```

mSeriesByKaranJulySevenA/target/site/serenity/
INFO] -----
INFO] SERENITY TESTS : ERROR
INFO] -----
INFO] | Tests executed      | 3
INFO] | Tests passed        | 1
INFO] | Tests failed        | 0
INFO] | Tests with errors    | 1
INFO] | Tests compromised    | 0
INFO] | Tests pending       | 0
INFO] | Tests ignored/skipped | 1
INFO] |-----|
INFO] | Total Duration      | 18s 006ms
INFO] | Fastest test took   | 1s 169ms
INFO] | Slowest test took    | 14s 399ms
INFO] |-----|
INFO] SERENITY REPORTS
INFO] - Full Report: file:///E:/Naveen%20Java%20Training/Cucumber/CucumberPomSeriesByKaranJulySevenA/target/site/serenity/index.html
INFO] -----
INFO] BUILD SUCCESS
INFO] -----
INFO] Total time: 40.940 s
INFO] Finished at: 2023-07-16T22:04:56+02:00
INFO] -----

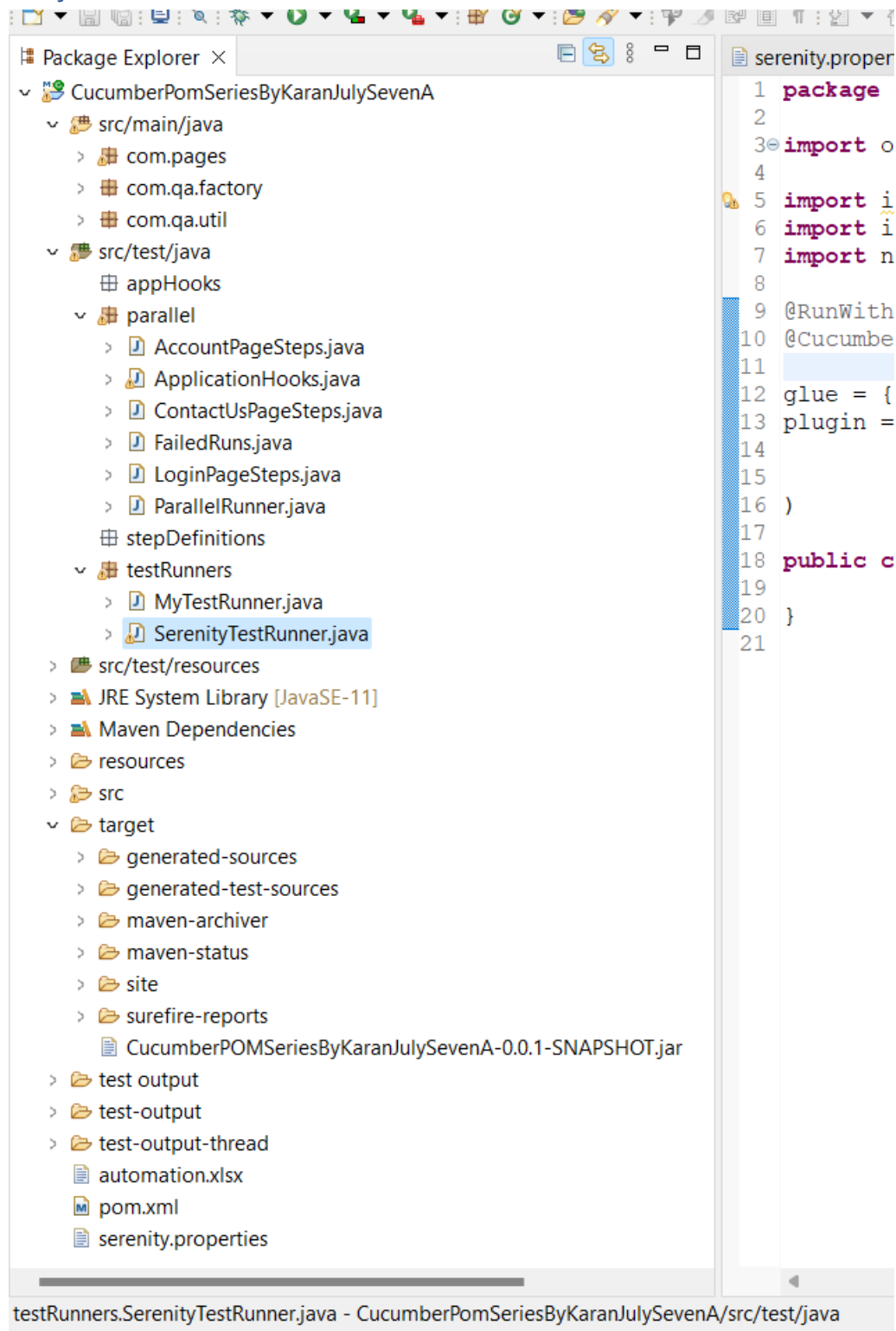
```

Inside target – site – serenity folder, lots of files generated. We don't want to see this. We copy the entire file path and open in browser.

Click on the serenity reports link in cmd and play around with the report-



Project structure-



Codes from this chapter-

Serenity properties file-

```

#give any name for the project or report
serenity.project.name=serenity and cucumber quick start by Naveen

```

Serenity runner-

```
package testRunners;

import org.junit.runner.RunWith;

import io.cucumber.junit.Cucumber;
import io.cucumber.junit.CucumberOptions;
import net.serenitybdd.cucumber.CucumberWithSerenity;

@RunWith(CucumberWithSerenity.class) // this is the only change when we
want serenity report
@CucumberOptions(
    features = { "src/test/resources/parallel/LoginPage.feature"
},
    glue = { "parallel" },
    plugin = { "pretty" }

)

public class SerenityTestRunner {

}
```