CLASSNOTES: DAY 18-2

Today's schedule:

#1- dryRun

#2- tags

#3- html reports

#4- Background

#5- Hooks

#6- Parameterization

#7- dataTables

#8- scenario outlines

#9- cucumber rerun

#10-report generation

#11-parallel testing

- finish the task
- shortcuts for pom design from intellij ultimate
- browserutils all methods
- cucumber rerun
- generate a new type of report: maven cucumber reporting
- parallel testing
- review

- What is the difference between data table and scenario outline?
- Data tables allow us to pass collection type into a step in a feature file.

- Scenario outline is completely different. It is similar approach to Parameterization. But instead of passing the test data inside of the step, we provide a test data as a table under our Scenario outline.

- We cannot create SCENARIO OUTLINE/TEMPLATE without providing an "Examples" table.

- But, we can use "data tables" without having to use "examples"

- Scenario Outline and Scenario Template are the same thing. Just alternative keyword.

RERUN:

- CUCUMBER RERUN IS A PLUGIN COMING FROM CUCUMBER

- This plugin allows us to keep track of the FAILED scenarios.

- All we have to do is use the plugin, and provide where it should be creating the .txt file.

HOW TO IMPLEMENT A RERUN PLUGIN?

#1-

syntax: add appropriate plugin under "plugins" in our runner class

```
plugin = {

            "html:target/cucumber-report.html",

            "rerun:target/rerun.txt"

    },
```

"rerun: --> plugin name

target/ --> where I want to store my rerun.txt

rerun.txt" --> file name and file type. I can pass any file name here.

#2- CREATE A NEW TEST RUNNER CLASS.

- This runner class will be pointing directly to the rerun.txt file that is generated by our plugin.

- It will only have one job.

- When we run it, it will execute the "failed" scenarios that are stored in the rerun.txt

```
@RunWith(Cucumber.class)

@CucumberOptions(

    glue = "com/cydeo/step_definitions",

    features = "@target/rerun.txt"

)

public class FailedTestRunner {

}
```

- Why do we need RERUN and FailedTestRunner?

    - Mostly to save time when tests fails.

    - If we have hundreds of tests, and only few of them are failing, we don't want to execute all of the hundreds of the tests again.

    - Thats why we can just execute the failed tests that are automatically stored under the rerun.txt using our FailedTestRunner.

- FailedTestRunner cannot change or edit the rerun.txt.

- Therefore even if the tests are passing, the lines will still in the rerun.txt, which is normal.

- We can delete rerun.txt by going Maven --> clean

#1- close project intellij and open again

#2- reload the project

#3- file > invalidate cache

file > restart intellij

#4- go to ~/.m2 file > go in the repository > select all folder > delete

then maven --> reload