Run the Test using GitHub and Jenkins (CI /CD)

Pre-requisites:

- GitHub Account
- Jenkins Account
- script/code available in github repository

Setup Github and Jenkins link:

- Create project in Jenkins
- Set the parameters

  Repository URL : <<your github repository URL>>

  Credentials: provide your github login credentials

  Branches to build: */main

- Build steps in Jenkins:

-> For Jenkins Running on **MAC OS**:

```
TIMESTAMP=$(date +%Y%m%d%H%M%S)
OUTPUT_DIR="report_$TIMESTAMP"
```

Note: Below is sample command, you need to change the paths in below command based on where your files are located.

```
/Users/vishwa/Softwares/JMeter/apache-jmeter-5.6.3/bin/jmeter.sh -n -t
"Test Scripts-Data-Result/TestScript/Test Plan - JpetStore Project.jmx" -l
"Test Scripts-Data-Result/TestResult/$OUTPUT_DIR/testresult.csv" -e -
o "Test Scripts-Data-Result/TestReport/$OUTPUT_DIR"
```

Explanation:
1. **TIMESTAMP=$(date +%Y%m%d%H%M%S)**
This command generates a timestamp in the format `YYYYMMDDHHMMSS` using the `date` function.

- The `+%Y%m%d%H%M%S` format gives the year, month, day, hour, minute, and second in sequence.

- It stores this value in the `TIMESTAMP` variable. This helps in ensuring that each test run or report gets a unique directory name, which is particularly useful for automation scenarios like performance testing where multiple runs may occur.

2. **OUTPUT_DIR="report_$TIMESTAMP**

This command creates a directory name, `report_$TIMESTAMP`, by concatenating the string `report_` with the value of the `TIMESTAMP` variable.

- This directory will be used to store the test results and reports for this specific run.

- It ensures your test reports are stored separately for each run, making it easier to analyze and organize your performance testing data, especially when running multiple tests on tools like JMeter.

3. `**/Users/vishwa/Softwares/JMeter/apache-jmeter-5.6.3/bin/jmeter.sh -n -t "Test Scripts-Data-Result/TestScript/Test Plan - JpetStore Project.jmx" -l "Test Scripts-Data-Result/TestResult/$OUTPUT_DIR/testresult.csv" -e -o "Test Scripts-Data-Result/TestReport/$OUTPUT_DIR"**`
This is the JMeter command that runs the actual test script. Here's what it does step by step:

- `**/Users/vishwa/Softwares/JMeter/apache-jmeter-5.6.3/bin/jmeter.sh**`: Runs JMeter in non-GUI mode, specifically using the shell script (`.sh`) to execute it.

- `**-n**`: Runs JMeter in non-GUI mode, which is ideal for automation or running tests from Jenkins.

- `**-t "Test Scripts-Data-Result/TestScript/Test Plan - JpetStore Project.jmx"**`: Specifies the location of the JMeter test plan (`.jmx` file) to execute, in this case, a test plan for the JPetStore project.
- `**-l "Test Scripts-Data-Result/TestResult/$OUTPUT_DIR/testresult.csv"**`: Logs the results of the test to a CSV file, stored in the directory named by the `OUTPUT_DIR` (which includes the timestamp).
- `**-e -o "Test Scripts-Data-Result/TestReport/$OUTPUT_DIR"**`: Generates an HTML report of the test run and stores it in the specified output directory.

-> For Jenkins running on **Windows OS**:

To get a timestamp in the format YYYYMMDDHHMMSS using a Windows batch command:

```
FOR /F "tokens=2 delims==" %%A IN ('WMIC OS GET LocalDateTime /VALUE') DO SET TIMESTAMP=%%A
SET TIMESTAMP=%TIMESTAMP:~0,8%%TIMESTAMP:~8,6%
```

Note:

1. FOR /F "tokens=2 delims==" %%A IN ('WMIC OS GET LocalDateTime /VALUE') DO SET TIMESTAMP=%%A: This extracts the current date and time using the WMIC command.
2. SET TIMESTAMP=%TIMESTAMP:~0,8%%TIMESTAMP:~8,6%: Reformats the timestamp to the YYYYMMDDHHMMSS format by extracting the first 8 characters (date) and the next 6 characters (time).

You can use %TIMESTAMP% in your batch script afterward to refer to the formatted timestamp.

For the Windows batch equivalent of OUTPUT_DIR="report_$TIMESTAMP"

```
SET OUTPUT_DIR=report_%TIMESTAMP%
```

Note:

- SET OUTPUT_DIR=report_%TIMESTAMP%: This sets the OUTPUT_DIR variable by concatenating the string "report_" with the value of the %TIMESTAMP% variable.

Now %OUTPUT_DIR% will hold the value "report_YYYYMMDDHHMMSS".

The corresponding Windows batch command :

Note: This is sample command, you need to change the paths in below command based on where your files are located.

```
C:\Users\vishwa\Softwares\JMeter\apache-jmeter-
5.6.3\bin\jmeter.bat -n -t "Test Scripts-Data-Result\TestScript\Test
Plan - JpetStore Project.jmx" -l "Test Scripts-Data-
Result\TestResult\%OUTPUT_DIR%\testresult.csv" -e -o "Test
Scripts-Data-Result\TestReport\%OUTPUT_DIR%"
```

Note:

1. JMeter path:
   - In Windows, the JMeter executable file is `jmeter.bat`, located at `C:\Users\vishwa\Softwares\JMeter\apache-jmeter-5.6.3\bin\jmeter.bat` (the forward slashes `/` from Unix are replaced by backslashes `\` in Windows).

2. Non-GUI Mode (`-n`):
   - The `-n` option runs JMeter in non-GUI mode, the same as in Unix.

3.Test Plan File (`-t`):
   - The path to the JMeter test plan file is `"Test Scripts-Data-Result\TestScript\Test Plan - JpetStore Project.jmx"`. This is the location of your JMX test script.

4. Log Results (`-l`):

- The path where JMeter will save the test results is `"Test Scripts-Data-Result\TestResult\%OUTPUT_DIR%\testresult.csv"`. The `%OUTPUT_DIR%` variable will be replaced by the directory name generated by your earlier timestamp command.

5. Generate Report (`-e -o`):
   - The `-e` option enables reporting, and `-o` specifies the path where the HTML report will be saved, which in this case is `"Test Scripts-Data-Result\TestReport\%OUTPUT_DIR%"`.

In this command, the `OUTPUT_DIR` is a Windows variable created earlier, containing the timestamp, and it is referenced using `%OUTPUT_DIR%` as per Windows batch scripting syntax.

- Build the project
- Verify the console log
- Verify the test result
- Verify the test report