

Jenkins, Git and GitHub (Continued)

- Explain how the Jenkins, Git and GitHub are going to work together - view [here](#)
- Configure the GitHub to work with Jenkins
 - Create a new repository in GitHub
 - Go to a folder in your machine where want to store the GitHub repository code
 - Open Git Bash from inside the above navigated folder
 - Type **pwd** command to confirm whether you are at the right path
 - Take the clone of the existing repository on GitHub using the below command:
 - Example: **git clone** <https://github.com/seleniumbyarun/ProjectXYZ.git>
 - Cloning will begin and an empty repo folder with a .git folder from GitHub will be copied to your local machine
 - Type the below two commands to provide a name and email ids to the activities that you are going to perform on GitHub
 - **git config --global user.name "Arun Motoori"**
 - **git config --global user.email "arun.motoori@gmail.com"**
 - Go inside the cloned folder on your local machine by using the below command
 - **cd ProjectABC**
 - Type **pwd** to confirm you are in the correct path
 - Type **git status** command and you will get nothing to commit
 - Now copy the Project files you want to upload to GitHub central repository
 - Again type **git status** command and you will this time get the details that can be committed
 - Type **git add .** to include all the files for committing
 - Commit the changes to confirm by using the command **git commit -m "MyFirstProjectCommit"**
 - Push the changes to the server by using the command **git push -u origin master**
 - After the above process is done, refresh the GitHub Project Repo page to see the Copied files in the repository
 - Go to Jenkins and create a new project
 - Configure the project by selecting the 'Git' option under Source Code Management and paste the URL of the GitHub Project
 - Example URL: <https://github.com/seleniumbyarun/ProjectABC>
 - Under Build section, select 'Invoke ANT' option and just provide the 'build.xml' into Build File field and 'run' target into the 'Targets' field and Save the Jenkins Project configurations.
 - Go the Dashboard project and build the Project using 'Build Now' option
 - View the results in the Console of Jenkins Project.
 - Install TestNG Results Plugin and configure the existing Jenkins Project by selecting 'Post Build' configuration option to 'Publish TestNG Reports'
 - Build the project again and view the generated TestNG results