Kuwait University
College of Computing Sciences and Engineering
Department of Information Science

COMP 3095 – Web Application Development with Java

Lab 2
Initializing the Project

Table of Contents

Laboratory #2 – Initilizing the Project

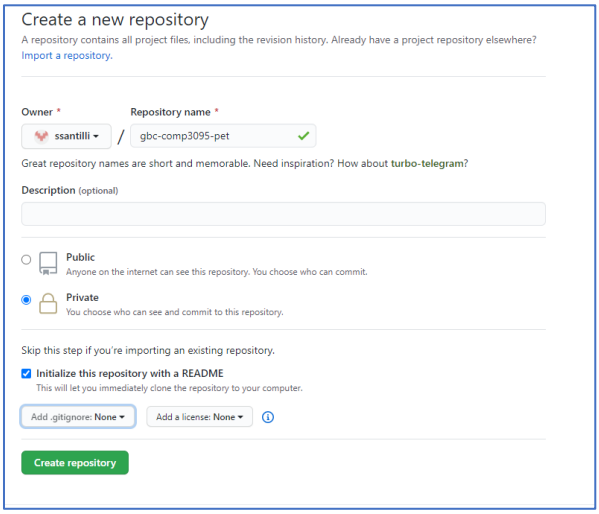
1. Laboratory Objective

The goal of this lab is to initialize the COMP3095-PET Clinic Project in GitHub (or GitLab etc..).

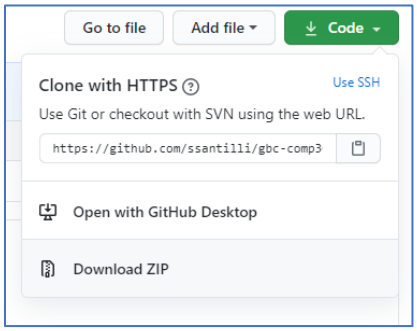
2. **Laboratory Learning Outcomes:** After conducting this laboratory students will be able to:
    a. Create a Git repository
    b. Configure Intellij to work with your Git Repository

3. Laboratory Instructions

a. Start by navigating to GitHub (create a free account if you like), and create new project repository.



b. Clone the project (using ssh or https).

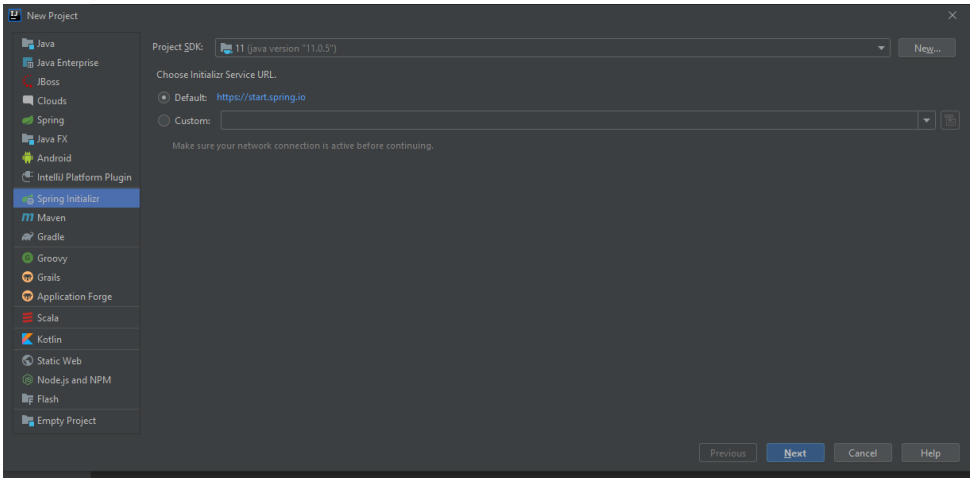


c. Issue git clone command to clone the project to your local directory of preference:

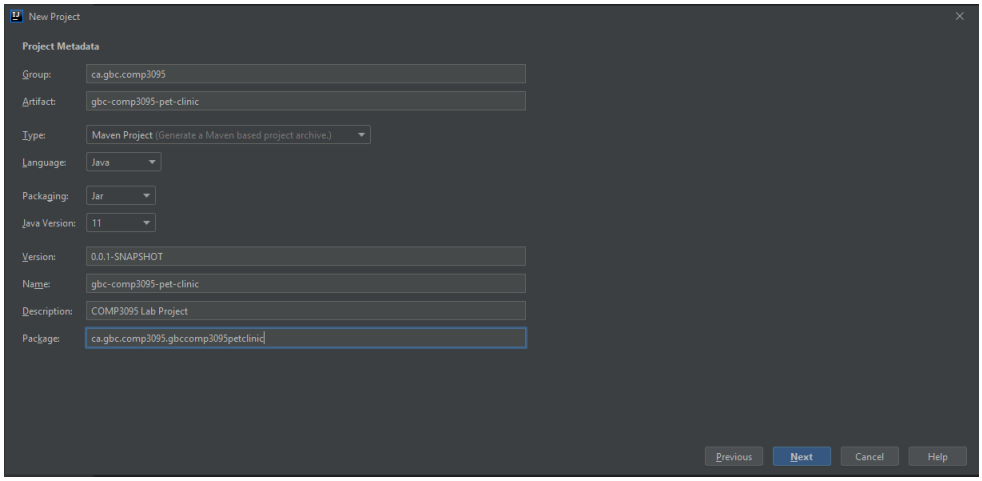
```
C:\MyProjects\java\comp3095\PetClinic
λ ls -l
total 0

C:\MyProjects\java\comp3095\PetClinic
λ git clone https://github.com/ssantilli/gbc-comp3095-pet-clinic.git
Cloning into 'gbc-comp3095-pet-clinic'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Unpacking objects: 100% (4/4), 867 bytes | 66.00 KiB/s, done.

C:\MyProjects\java\comp3095\PetClinic
λ ls -l
total 0
drwxr-xr-x 1 UTSCAD+santil13 2147484161 0 Aug 14 14:46 gbc-comp3095-pet-clinic/
```

```
C:\MyProjects\java\comp3095\PetClinic
λ cd gbc-comp3095-pet-clinic\

C:\MyProjects\java\comp3095\PetClinic\gbc-comp3095-pet-clinic (master -> origin)
λ ls -ltra
total 6
drwxr-xr-x 1 UTSCAD+santil13 2147484161   0 Aug 14 14:45 ../
-rw-r--r-- 1 UTSCAD+santil13 2147484161 301 Aug 14 14:46 .gitignore
-rw-r--r-- 1 UTSCAD+santil13 2147484161  57 Aug 14 14:46 README.md
drwxr-xr-x 1 UTSCAD+santil13 2147484161   0 Aug 14 14:46 ./
drwxr-xr-x 1 UTSCAD+santil13 2147484161   0 Aug 14 14:46 .git/
```

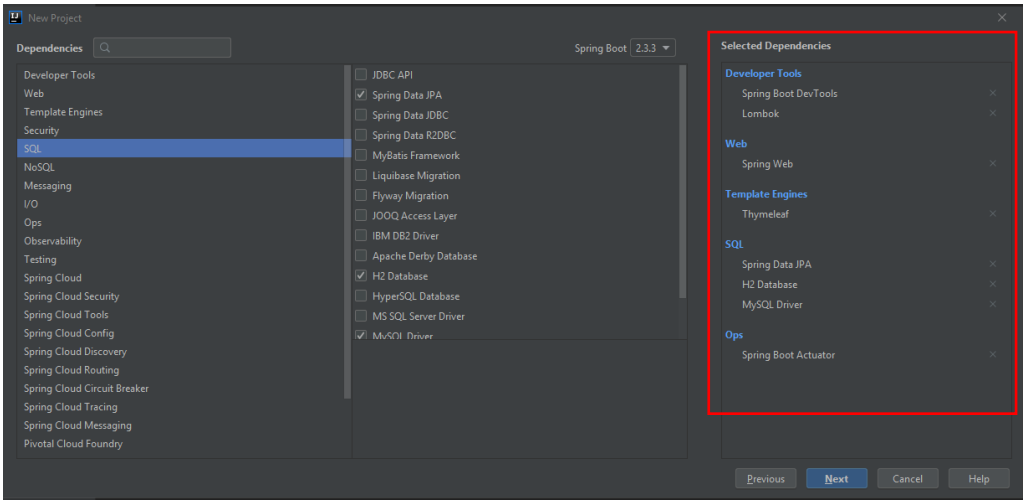
d. Create a new Project, pointing to the git pet project shell cloned. File → New Project



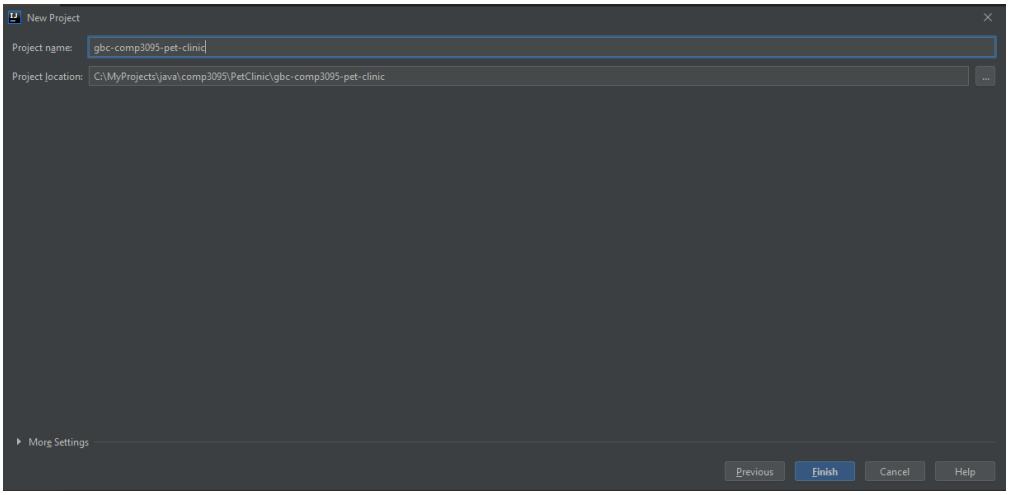
e. Click next, enter project details



f. The following screen represents similar options as presented in start.spring.io for addon/package module selection.
    1. Select a stable release of Spring (not a snapshot version)
    2. **Core**
        i. Spring Boot DevTools
        ii. Lombok
    3. **Web**
        i. Spring Web
    4. **Template Engines**
        i. Thymeleaf
    5. **SQL**
        i. Spring Data JPA
        ii. MySQL Driver
        iii. H2
    6. **Ops**
        i. Spring Boot Actuator

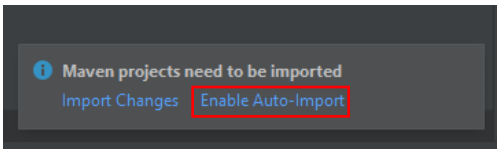


g. Click Next, validate project directory to point to git cloned project directory.

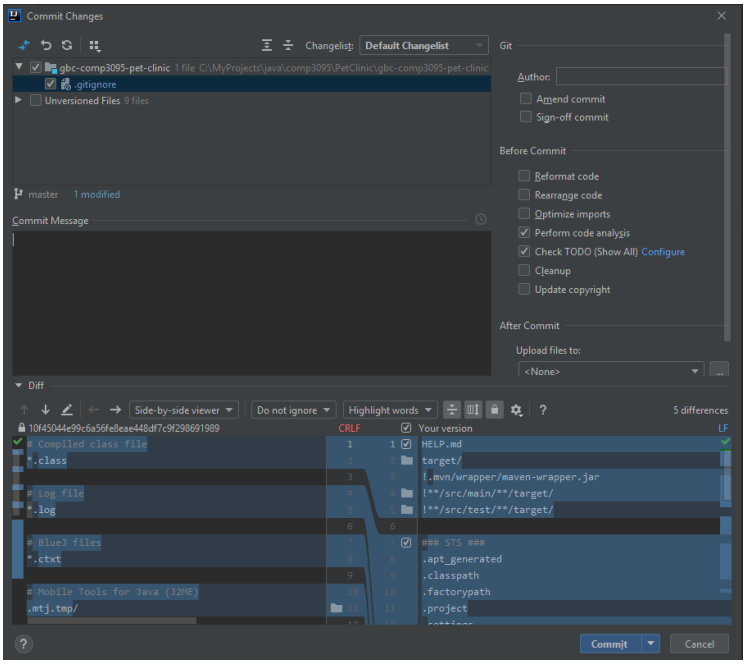


h. Click finish, behind the scenes intellij will call-out to start.spring.io, create and index the project.

i. Be sure to enable auto-import if presented.

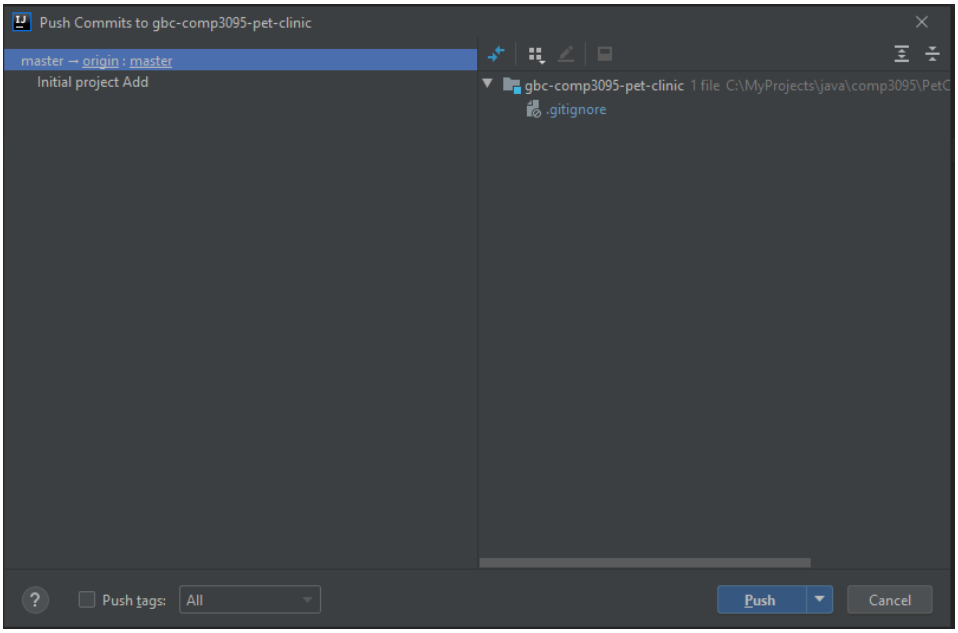


j. Commit project directory to Git, **Git→Commit directory**



k. Add a commit message (good practice)
   a. Tips top remember.
      i. **Commit** → only commits to local repository
      ii. **Commit and Push** → Commits to local repository and pushes to remote git repository.

l. Confirm Push



m. Validate project has been commited and pushed