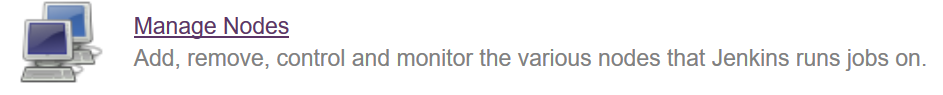
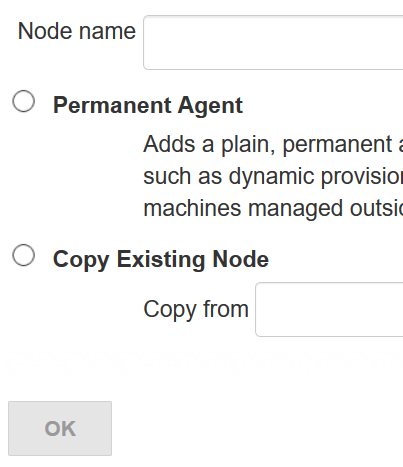
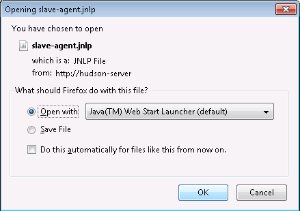
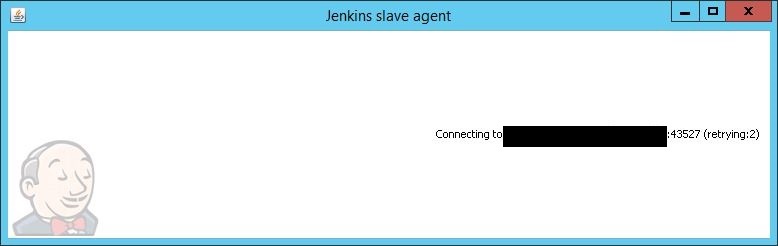
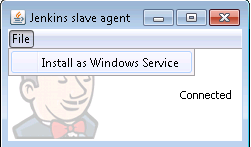
Why we need to configure Jenkins in Master slave:

1. For avoiding single point for failure.
2. For supporting multiple technologies(for Window’s- Jenkins window, for Linux- Jenkins Linux- for mac-> Linux mac)
3. Note: If we are not implementing in master slave then we need to enhance server capacity.
4. On your master machine go to **Manage Jenkins** > **Manage Nodes**.  
   
5. **New Node**
   1. **Enter Node Name**
   2. **Select Permanent Agent**
   3. Press **OK**.  
      
6. Fill out the following:
   1. Set a **number of executors**
      1. (one or more) as needed.
   2. Set a **Remote FS Root**
      1. a home directory for the master on the agent machine.
      2. For a *Windows agent*, use something like: "C:\Jenkins\"
   3. Select the appropriate **Usage** setting:
      1. For an additional worker: *Utilize this node as much as possible*
      2. For specialized jobs: *Leave this machine for tied jobs only*
   4. **Launch Method**:
      1. An easy way to control a Windows agent is by using *Launch agent via Java Web Start*  (Recommended for Windows)
      2. TODO: add steps for other methods.
   5. **Availability**
      1. *Keep this agent online as much as possible*
      2. TODO: add details for each option.
   6. Press **OK**.  
      
7. Now you need to connect your agent machine to the master using the following steps.
   1. Open a browser on the **agent machine** and go to the **Jenkins master server** url (<http://yourjenkinsmaster:8080>).
   2. Go to **Manage Jenkins** > **Manage Nodes**,
      1. Click on the newly created agent machine. You will need to login as someone that has the "Connect" Agent permission if you have configured global security.
   3. Click on the **Launch** button to launch agent from browser on agent.  
      
   4. Run the program.  
        
      If Windows asks you to choose a program, use $JAVA\_HOME\bin\javaws.exe (the Java Web Start Launcher).
   5. If you encounter connection issue, then you could enlarge the popup windows to see the master **port used** and check your network configuration (firewall, port forward, ...).   
        
      Note that Jenkins chooses a random, high-number port. If you want Jenkins to use a static port, go to "Manage Jenkins" > "Configure Global Security" and choose the port number in the "TCP port for JNLP agents" box.
   6. If the port is open, the agent still can't connect, and your Jenkins instance is served securely over SSL/HTTPS, download and install the [Java Cryptography Extension (JCE) Unlimited Strength Jurisdiction Policy Files 8](http://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html). Take the .jar files and save them to $JAVA\_HOME\lib\security. Try the above steps again.
   7. Now you should see the agent machine connected under **Nodes**.



1. If you want the service to run on start-up of the agent machine do the following (Windows only directions):
   1. In the agent program running on your agent machine,
   2. click **File** --> **Install as Windows Service.**  
        
      Note that this feature requires ".Net Framework 3.5"  
      
   3. **Start**, type Services and Select the **Services** program.
   4. Find **Jenkins Agent** in the list, Double click to open.
   5. Select **Startup type** --> **Automatic**.
   6. Go to the **Log On** tab, change the **Log on as** to a user of your choice (Special user account Jenkins recommended).
   7. Make sure that auto login is set for the agent machine for the user account, then the VM (or physical computer) should connect and be available when needed.

TODO: Add steps for connecting via ssh

In Jenkins sunny is a better day-> cloud means issue

Function testing: done by tester

Unit testing: done by developer unit test.

Code quality check: SonarQube