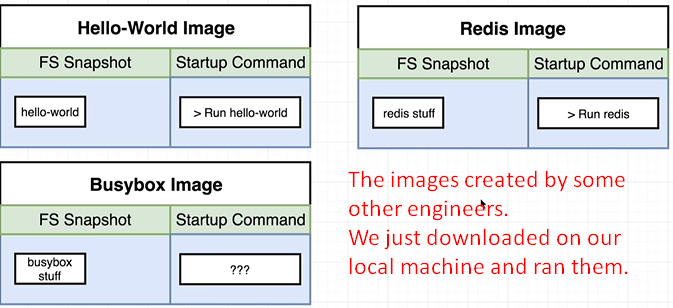
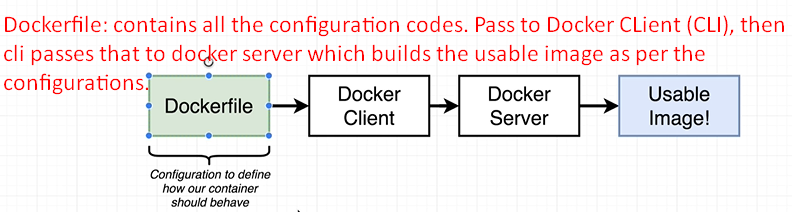
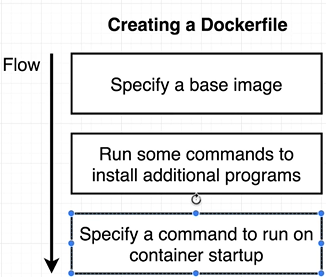
1. Till now, we were making use of images created by some other engineers such as “hello-world”, “busybox”, “redis”.
2. We downloaded them on our local machine and created container out of those images.  
   
3. **Agenda**: How to build our own **custom images** so that we can run our **own applications** inside of our **own customized container**.
4. The process is very straightforward.
5. ** Steps**:
   1. **Dockerfile:**
      1. Create docker file.
      2. Simple text file.
      3. Having a couple of lines of configuration.
      4. Configuration decides how our container behaves. More specifically what different program it’s going to contain and what it does when it starts up as a container.
      5. This file contains all the complexities. It’s just about learning a couples of new commands that’s all.
   2. **Docker Client**:
      1. The docker CLI.
      2. We pass the docker file to docker client.
      3. Docker client provides the docker file to the docker server.
   3. **Docker Server**:
      1. Docker Server does heavy lifting for us.
      2. It takes the docker file then takes a look at configuration lines inside of it then builds a usable image that can be used to start up a new container.
6.  s  
   **Specify a base Image**: Inside every docker file, we’re going to specify a base image.  
   **Run some commands to install additional programs**: Then add some additional configuration to run commands to add in some dependencies or some more software that we need to successfully create and execute our container.  
   **Specify a command to run on container startup:** Finally, we will specify a startup command for the image.   
   So anytime we take that image and creates container out of it, it will be the command that is executed to essentially boot up or start the container.