Portlet-ext.properties

**Direct theme deploy -** gulp.watch(['\*.scss', '\*.ftl'], gulp.series('deploy'));

**Port kill command**

* Tomcat server: ps aux | grep tomcat
* Docker server: ps aux | grep docker
* Kill –9 id

**Server starts :** got to root folder >server>tomcat>bin>**./catalina.sh run**(it’s start server and also print logs)

Docker installation : <https://docs.docker.com/engine/install/ubuntu/#prerequisites>

* $ sudo apt-get update
* $ sudo apt-get install \  
   ca-certificates \  
   curl \  
   gnupg \  
   lsb-release
* sudo mkdir -m 0755 -p /etc/apt/keyrings
* $ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
* echo \ "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg] <https://download.docker.com/linux/ubuntu> \ $(lsb\_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
* sudo apt-get update

How Docker works

* Sudo docker run hello-world
* Docker client connected to our docker file. Docker file created docker image or container
* **Container** means docker image add runtime to container keeping program file from all other application by lightweight virtual machine. Docker start from inside the isolated file system. (gujarati docker image create kare 6e. Image ma system sathe koi leva deva nahi because ae machine ma alag thi storage create kare 6e ane run time ae ae isolated storge no use kare 6e je already registered 6e.) One word container is sandboxed process on our machine that isolated from all other process.
* **Container image** while running a container it uses isolated file system. Image contains container filesystem. It must contain by run application –all depedancy,configuration,binaries. It has also contains configurations like – env variable , default command to run.
* STARTING WITH DOCKER - <https://learn.liferay.com/dxp/latest/en/getting-started/starting-with-a-docker-image.html>
  + **docker pull liferay/portal:7.4.3.55-ga55**
  + **docker run -it -m 8g -p 8080:8080 liferay/portal:7.4.3.55-ga55**
    - -it means stop docker container and start this one
    - -p means **leftside:custom port** and **rightside:default liferay server port**
    - **-m 8g** memory allocated as 8 giga bytes
  + **docker container ls –a** till created docker container
  + **docker ps –q** current docker running gives an id
  + Remove unwanted docker by **sudo docker rm [container-name]** 
    - contaner-name from **docker container ls –a** check names field
  + Logs – **docker logs –f [container-name]**
  + Start docker container **- docker start [container-name]**
  + Rename docker – **docker rename [current-container] [new-container]**
* Docker database name in properties file
  + jdbc:mysql://db:3306/liferay\_docker\_mysql\_1
  + Means the name db get from docker file
  + Why we use 3306 because we use same network in docker file on bottom u can see so it take default mysql port 3306 liferya and db are connected by one network so we define db name and port default of mysql in port-ext.properties

**PROPERTIES**

* PORTAL
* SYSTEM

1. PORTAL PROPERTIES
   1. Some configuration update from site configuration but others are updated from properties file
   2. Why use?
      1. We can manage version controller
      2. Reuseable this file to other liferay project
      3. Database connection
      4. Connect to home path
   3. It’s initialized as either an environment variable or in volumes on docker
   4. Portal properties set by system settings and configuration file are stored in the database
   5. include-and-override=portal-developer.properties at top of portal-ext.properties and it enable all developer properties
   6. Everything can be configured of Liferay installation is configurable in docker container.
   7. Can be over written in either portal.properties file or docker env variable
   8. **Environment variable** can be override in portal-ext.properties in only linux
      1. Start with LIFERAY\_
      2. It shout be uppercase, digit, underscore(\_)but not start with digit
      3. Two way to define environment
         1. **charPool constant**
            1. In portal properties = setup.database.driverClassName[mysql]=com.mysql.jdbc.Driver
            2. Environment= LIFERAY\_SETUP\_PERIOD\_DATABASE\_PERIOD\_DRIVER\_UPPERCASEC\_LASS\_UPPERCASEN\_AME\_OPENBRACKET\_MYSQL\_CLOSEBRACKET\_=com.mysql.jdbc.Driver
         2. **Unicode constant**
            1. LIFERAY\_SETUP\_46\_DATABASE\_46\_DRIVER\_67\_LASS\_78\_AME\_91\_MYSQL\_93\_=com.mysql.jdbc.Driver
2. SYSTEM PROPERTIES
   1. Can be overridden in system-ext.properties
3. SYSTEM SETTINGS (control menu>system settings)
   1. Four categories: platform, security, contents and data, others
   2. It's overrides in widget settings like
      1. System settings >blogs>widget template>disable enable commnets save this
      2. Site menu>content and data> blogs >create test blogs >save
      3. Add this on to home page as blogs
      4. Click on configuration at right top
      5. Now u can see that it’s override the value
4. **SYSTEM SETTINGS** VS **INSTANCE SETTINGS**
   1. **Virtual instace** meaning liferay has multiple portals where each portals have different data, configurations and unique domain name so is a separate liferay based implementation.
   2. Instance have similar to system but the configuration and virtul instance scop are different
      1. System settings the instance configuration we reset to default values.(updated configuration values override)
      2. Instance setting it’s removes the custom configuration defualt value and add current value which in virtual scope like (vs code configuration value takes it’s not overrird the vs value)
   3. System settings use for global configuration
   4. Instance setting use for virtual host based configuration (applies on only one virtual instance of liferay)
5. Import – Export Configurations
   1. Name should be .config
   2. Export these configurations from sites and import in [liferay\_home]/server/volumes/liferay/osgi/configs(replace folder)
6. INSTANCE SETTINGS
   1. Structure - Adding VI, configure VI, integrating user to liferay
   2. Location – control panel>system>virtual instance