=========================

Readiness and liveness

=========================

Readiness is used to determine when the pods is ready to accept the traffic

liveness is used to check whether the pod is live or not

====================

Configmap and secrets

====================

For every application multiple environment will be available for testing purpose

a) DEV (development)

b) SIT (system integration testing)

c) UAT (user acceptance testing)

d) PILOT pre-production env

DEV env used by development team for code integration testing

SIT env used by testing team for functional testing

UAT env used by client-side team for acceptance testing

PILOT env used for pre-production testing

Once application testing is completed in all above environments then it will be deployed into PRODUCTION environment (Live Environment).

Note: For every environment, some properties will be different

1) Database properties

2) SMTP properties

### We should not hardcode properties in our source code ###

### We need to make our application loosely coupled to deploy in any environment ###

=> Config Map & Secret concepts are used to avoid hard coded properties in the application.

=> Config Map & Secret allows us to de-couple application properties from Docker images so that our application can be deployed into any environment without making any changes for our Docker image.

=> Config Map is used to store data in key-value (non-confidential)

=> Secret is used to store confidential data in key - value format (ex: pwd)

### Note: ConfigMap & Secrets will make docker images as portable. ###

=========================================

Application properties (hard coded value)

=========================================

spring:

datasource:

driver-class-name: com.mysql.cj.jdbc.Driver

url: jdbc:mysql://mysqldb:3306/sbms

username: root

password: root

jpa:

hibernate:

ddl-auto: update

show-sql: true

========================================

Application properties (Environment variables)

=========================================

spring:

datasource:

driver-class-name: ${DB\_DRIVER}

url: ${DB\_URL}

username: ${DB\_USERNAME}

password: ${DB\_PASSWORD}

jpa:

hibernate:

ddl-auto: update

show-sql: true

=================================================================================

Application properties (Environment variables) - if you want to run it locally

=================================================================================

spring:

datasource:

driver-class-name: ${DB\_DRIVER: com.mysql.cj.jdbc.Driver}

url: ${DB\_URL: jdbc:mysql://mysqldb:3306/sbms}

username: ${DB\_USERNAME: root}

password: ${DB\_PASSWORD: root}

jpa:

hibernate:

ddl-auto: update

show-sql: true

==================

Configmap manifest

==================

# vim config.yml

---

apiVersion: v1

kind: ConfigMap

metadata:

name: vinod-config-map

labels:

storage:vinod-db-config-map

data:

DB\_DRIVER: com.mysql.cj.jdbc.Driver

DB\_URL: jdbc:mysql://mysqldb:3306/sbms

DB\_USERNAME: root

...

==================

secret manifest

==================

# vim secret.yml

---

apiVersion: v1

kind: Secret

metadata:

name: vinod-secret

labels:

storage:vinod-scret

data:

DB\_PASSWORD: cm9vdA==

type: Opaque

...