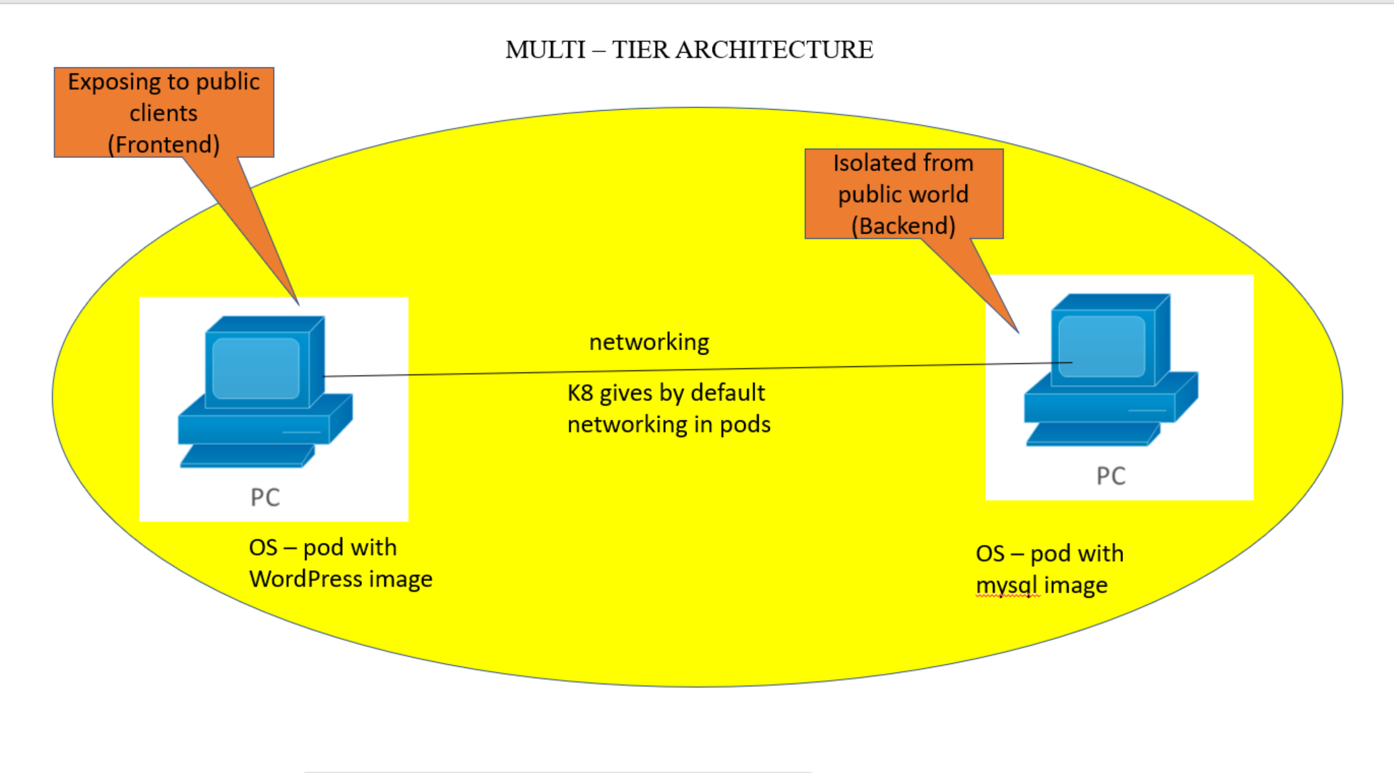
DEPLOY MULTINODE ARCHITECHTUTE IN K8

e.g FB as frontend interface managing with one OS, and for bacakend manaegin with one OS.

1. launching one pod in k8 and configuring Wordpress image as fronted.
2. launching another pod in K8 and configuring mysql database as backend.
3. Exposing the fronthend(NODE PORT in K8) server for public client.
4. keeping another OS isolated as the best secuty is avaoid the networking with public world.
5. Concept of ENVIRONMENT VARIABLES also known as Shell Variables.
6. login into the pod of database via CLI and then login in mysql to view and edit the tables.


MULTI – TIER ARCHITECTURE
Exposing to public clients (Frontend)
Isolated from public world (Backend)
networking
K8 gives by default networking in pods
PC
PC
OS – pod with WordPress image
OS – pod with mysql image

Commands

1. kubectl run mydb –image=mysql:5.7 - lauch pod with mysql image

2. kubectl logs mydb - a troubleshooting way to view the logs in case of environment variables.

3. kubectl exec -it myos1 -- bash - login into the pod via CLI

4. x=4 echo$x - set temporary env var. lost after logout

5. vi /root/.bashrc - dir to make a permanent env var, but lost after restart

6. kubectl run myos1 --image=vimal13/apache-webserver-php --env=x=10 - pass env var with command, lost only when pod is deleted

7. kubectl run mywp –image=wordpress:5.1.1-php7.3-apache - launch pod with wordpress image

8. kubectl run mydb --image=mysql:5.7 --env=MYSQL_ROOT_PASSWORD=redhat --env=MYSQL_DATABASE=wpdb --env=MYSQL_USER=akshay --env=MYSQL_PASSWORD=anil

9. mysql -u <username> -p<password> - login into mysql

10. SQL commands

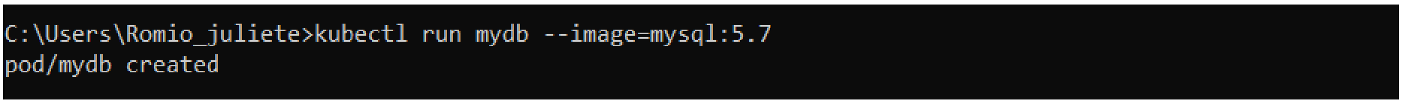
```
show databases;
use <databaseName>
show tables;
```

setting up frontend

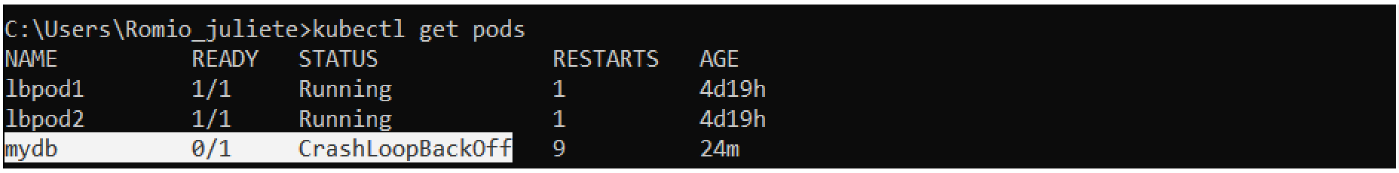
1.

kubectl run mydb –image=mysql:5.7

kubectl get pods



```
C:\Users\Romio_juliete>kubectl run mydb --image=mysql:5.7
pod/mydb created
```



```
C:\Users\Romio_juliete>kubectl get pods
NAME           READY   STATUS             RESTARTS   AGE
lbpod1         1/1     Running            1          4d19h
lbpod2         1/1     Running            1          4d19h
mydb           0/1     CrashLoopBackOff   9          24m
```

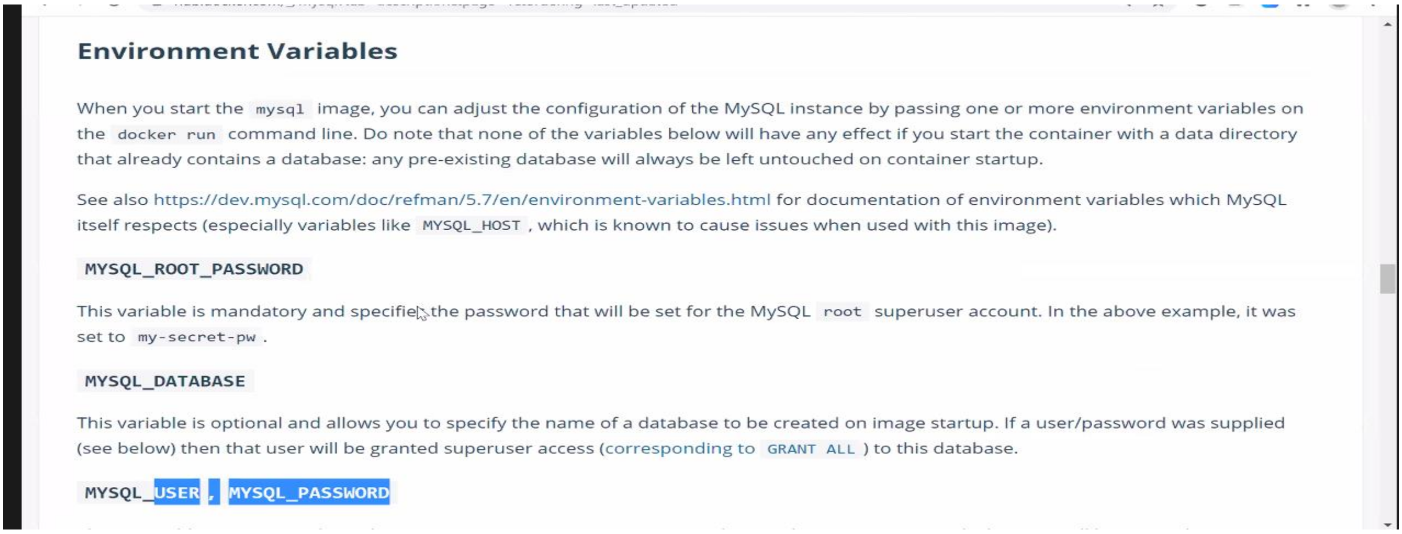
What is this error of “CrashLoopBackOff”?

Ans.

Whenever encounter the error while using an images for OS, check the log files, maybe the developer wants you to pass some variables, which is technically known as Environment Variables.

here these variables are (MYSQL_ROOT_PASSWORD, MYSQL_DATABASE, MYSQL_USER, MYSQL_PASSWORD).

let’s check the logs… kubectl logs mydb



Environment Variables

When you start the `mysql` image, you can adjust the configuration of the MySQL instance by passing one or more environment variables on the `docker run` command line. Do note that none of the variables below will have any effect if you start the container with a data directory that already contains a database: any pre-existing database will always be left untouched on container startup.

See also https://dev.mysql.com/doc/refman/5.7/en/environment-variables.html for documentation of environment variables which MySQL itself respects (especially variables like `MYSQL_HOST`, which is known to cause issues when used with this image).

`MYSQL_ROOT_PASSWORD`

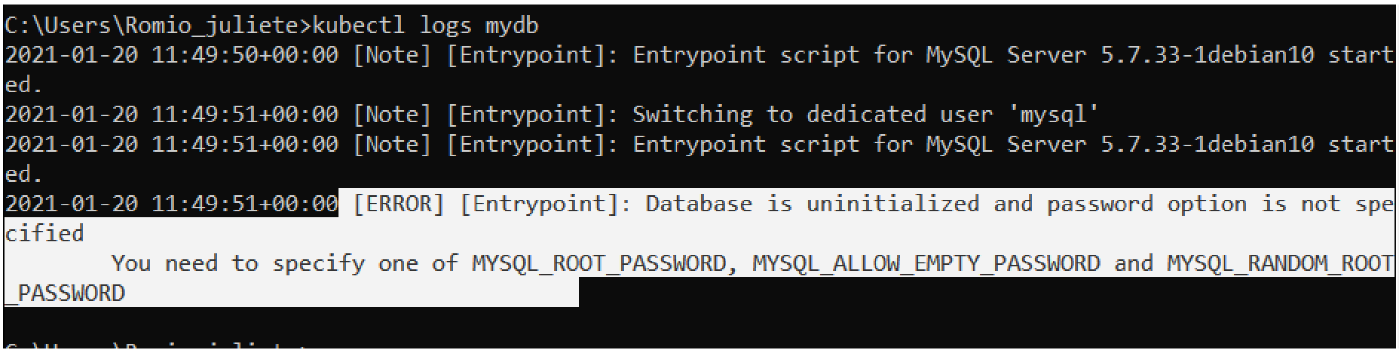
This variable is mandatory and specifies the password that will be set for the MySQL `root` superuser account. In the above example, it was set to `my-secret-pw`.

`MYSQL_DATABASE`

This variable is optional and allows you to specify the name of a database to be created on image startup. If a user/password was supplied (see below) then that user will be granted superuser access (corresponding to `GRANT ALL`) to this database.

`MYSQL_USER`, `MYSQL_PASSWORD`

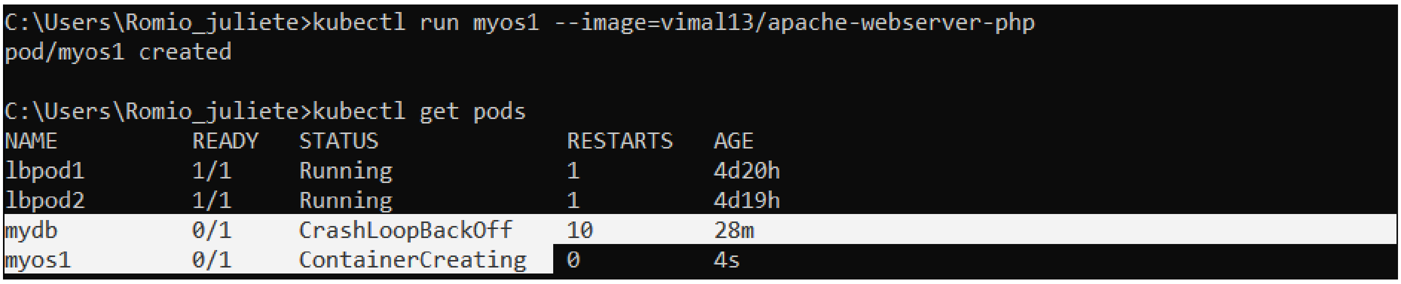
kubectl logs mydb



```
C:\Users\Romio_juliete>kubectl logs mydb
2021-01-20 11:49:50+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 5.7.33-1debian10 start
ed.
2021-01-20 11:49:51+00:00 [Note] [Entrypoint]: Switching to dedicated user 'mysql'
2021-01-20 11:49:51+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 5.7.33-1debian10 start
ed.
2021-01-20 11:49:51+00:00 [ERROR] [Entrypoint]: Database is uninitialized and password option is not spe
cified
        You need to specify one of MYSQL_ROOT_PASSWORD, MYSQL_ALLOW_EMPTY_PASSWORD and MYSQL_RANDOM_ROOT
_PASSWORD

C:\Users\Romio_juliete>
```

let’s launch a pod.

kubectl run myos1 --image=vimal13/apache-webserver-php

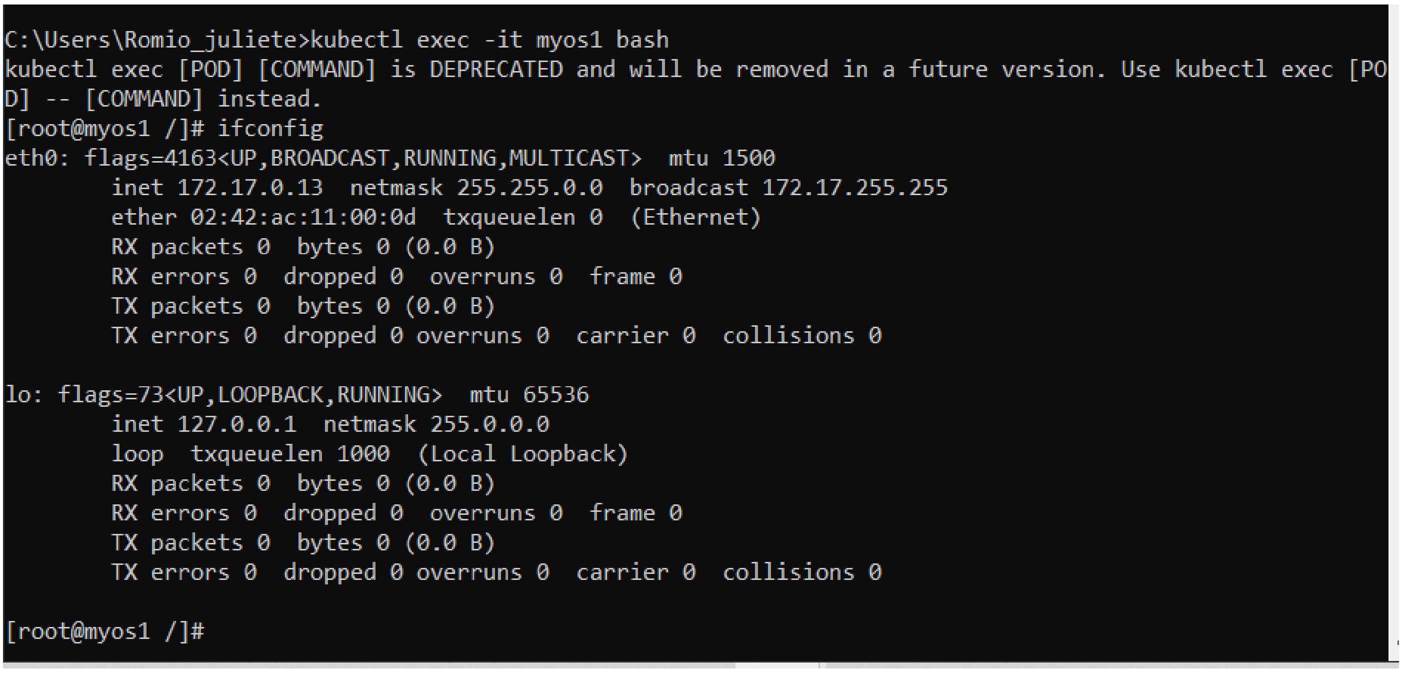


```
C:\Users\Romio_juliete>kubectl run myos1 --image=vimal13/apache-webserver-php
pod/myos1 created

C:\Users\Romio_juliete>kubectl get pods
NAME          READY   STATUS              RESTARTS   AGE
lbpod1        1/1     Running             1          4d20h
lbpod2        1/1     Running             1          4d19h
mydb          0/1     CrashLoopBackOff    10         28m
myos1         0/1     ContainerCreating   0          4s
```

2.

login into pod via CLI.

kubectl exec -it myos1 bash

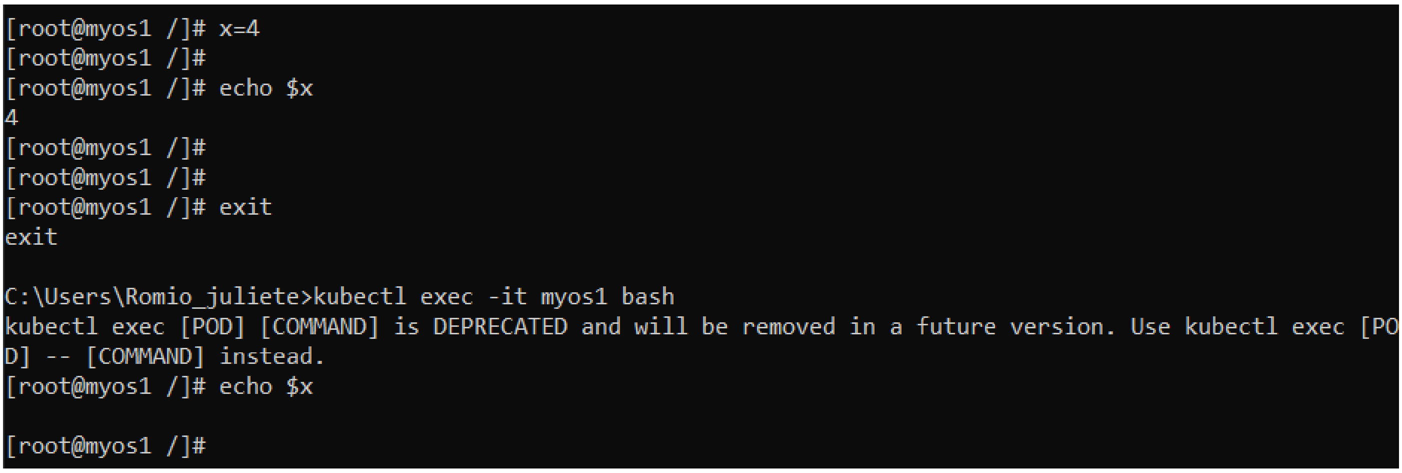


```
C:\Users\Romio_juliete>kubectl exec -it myos1 bash
kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec [PO
D] -- [COMMAND] instead.
[root@myos1 /]# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 172.17.0.13  netmask 255.255.0.0  broadcast 172.17.255.255
        ether 02:42:ac:11:00:0d  txqueuelen 0  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[root@myos1 /]#
```

Making temporary Env Variables

temporary because- > it get lost after logout from the pod

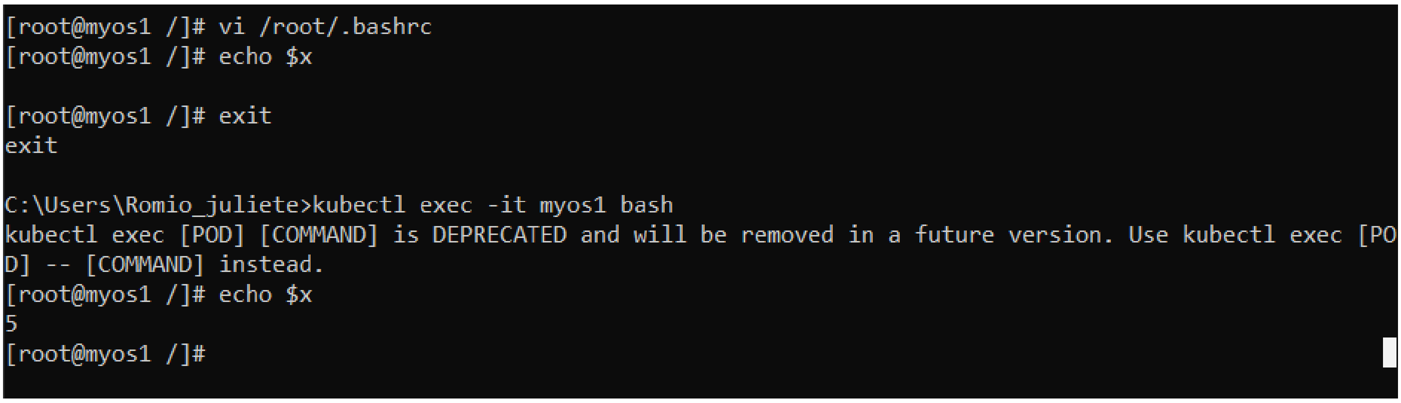
let’s verify – login again and type echo$x



```
[root@myos1 /]# x=4
[root@myos1 /]#
[root@myos1 /]# echo $x
4
[root@myos1 /]#
[root@myos1 /]#
[root@myos1 /]# exit
exit

C:\Users\Romio_juliete>kubectl exec -it myos1 bash
kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec [PO
D] -- [COMMAND] instead.
[root@myos1 /]# echo $x

[root@myos1 /]#
```

ways to make it permanent. open /root/.bashrc file and write there – exit – then again login…

vi /root/.bashrc

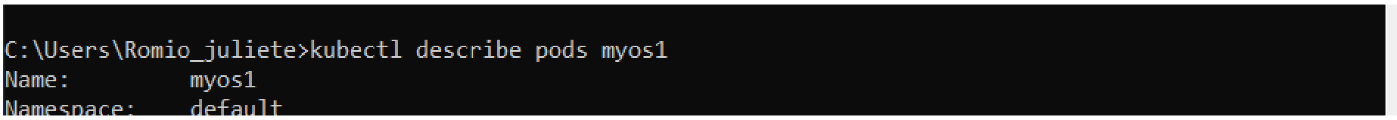


```
[root@myos1 /]# vi /root/.bashrc
[root@myos1 /]# echo $x

[root@myos1 /]# exit
exit

C:\Users\Romio_juliete>kubectl exec -it myos1 bash
kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec [PO
D] -- [COMMAND] instead.
[root@myos1 /]# echo $x
5
[root@myos1 /]#
```

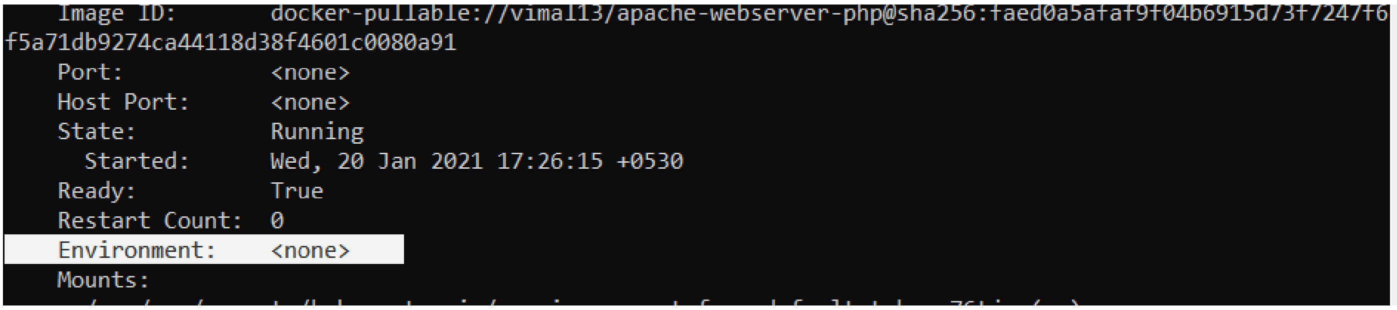
but this way works, only till the pod is running…

lets verify by describing the pod myos1.

kubectl describe pods myos1



```
C:\Users\Romio_juliete>kubectl describe pods myos1
Name:         myos1
Namespace:    default
```



```
    Image ID:       docker-pullable://vimal13/apache-webserver-php@sha256:faed0a5afaf9f04b6915d73f7247f6
f5a71db9274ca44118d38f4601c0080a91
    Port:           <none>
    Host Port:      <none>
    State:          Running
      Started:      Wed, 20 Jan 2021 17:26:15 +0530
    Ready:          True
    Restart Count:  0
    Environment:    <none>
    Mounts:
```

THEREFORE, the only way remains to set Env Var. while launching the OS.

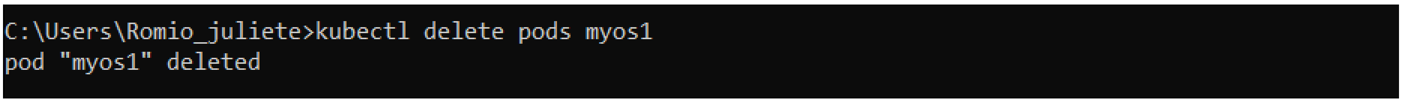
Deleting myos1 - > creating a new one and setting the Env Var simultaneously-> check environment->verify by login into CLI of POD

kubectl delete pods myos1

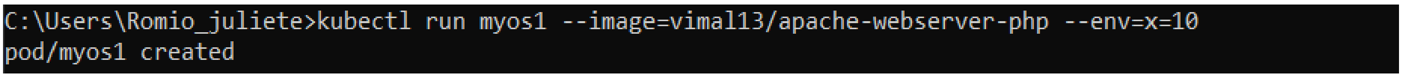
kubectl run myos1 --image=vimal13/apache-webserver-php --env=x=10

kubectl exec -it myos1 bash

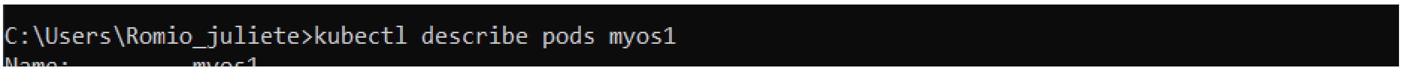
echo $x



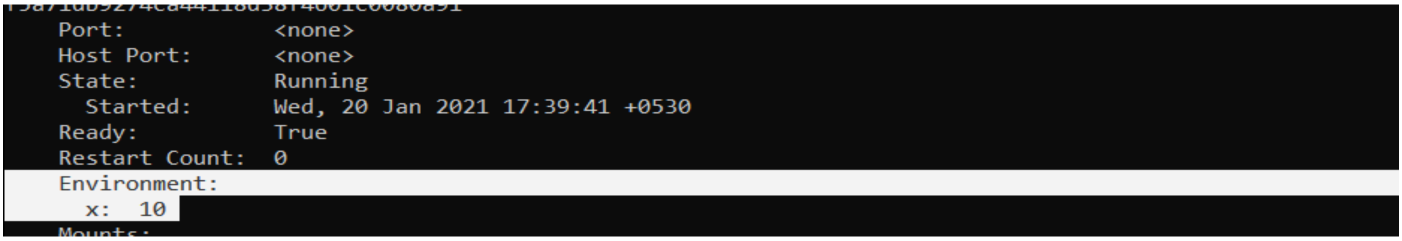
```
C:\Users\Romio_juliete>kubectl delete pods myos1
pod "myos1" deleted
```



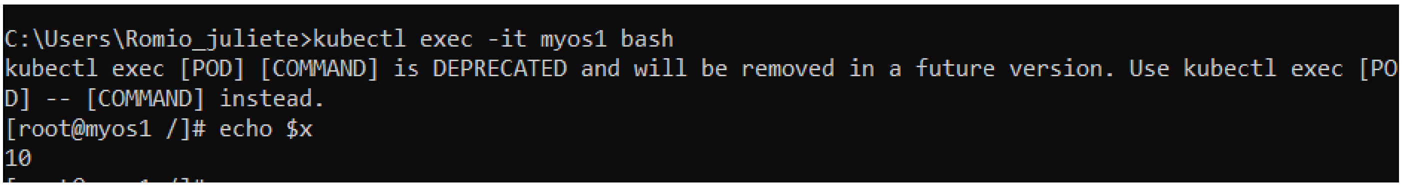
```
C:\Users\Romio_juliete>kubectl run myos1 --image=vimal13/apache-webserver-php --env=x=10
pod/myos1 created
```



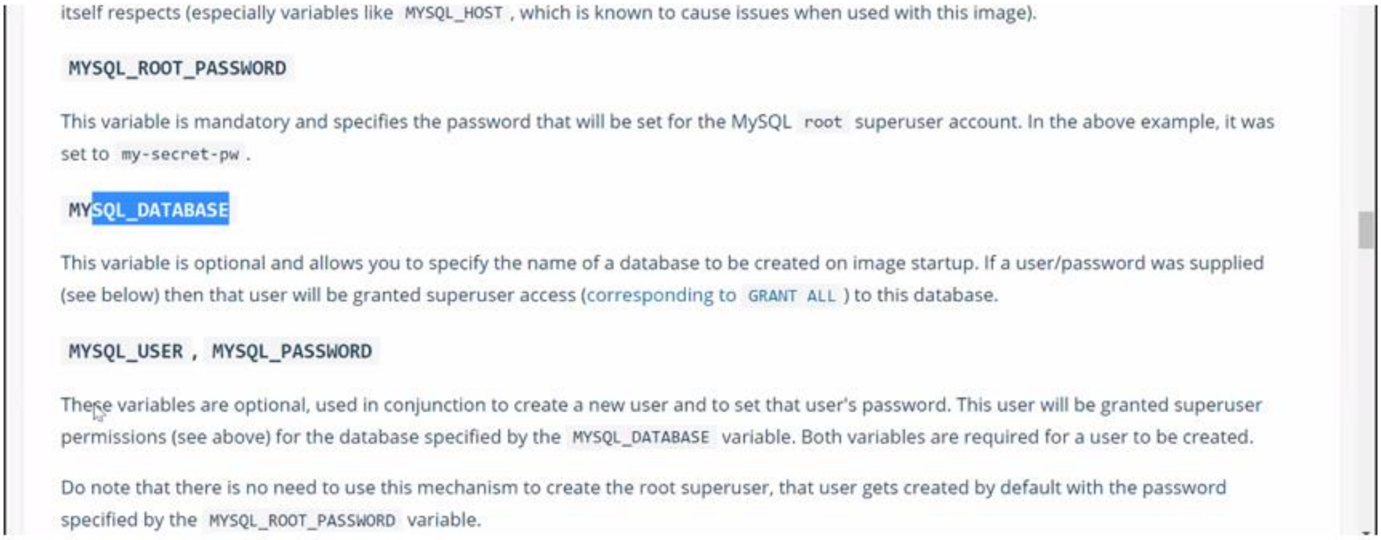
```
C:\Users\Romio_juliete>kubectl describe pods myos1
Name:         myos1
```



```
    Port:           <none>
    Host Port:      <none>
    State:          Running
      Started:      Wed, 20 Jan 2021 17:39:41 +0530
    Ready:          True
    Restart Count:  0
    Environment:
      x:  10
    Mounts:
```



```
C:\Users\Romio_juliete>kubectl exec -it myos1 bash
kubectl exec [POD] [COMMAND] is DEPRECATED and will be removed in a future version. Use kubectl exec [PO
D] -- [COMMAND] instead.
[root@myos1 /]# echo $x
10
```



itself respects (especially variables like `MYSQL_HOST`, which is known to cause issues when used with this image).

`MYSQL_ROOT_PASSWORD`

This variable is mandatory and specifies the password that will be set for the MySQL `root` superuser account. In the above example, it was set to `my-secret-pw`.

`MYSQL_DATABASE`

This variable is optional and allows you to specify the name of a database to be created on image startup. If a user/password was supplied (see below) then that user will be granted superuser access (corresponding to `GRANT ALL`) to this database.

`MYSQL_USER`, `MYSQL_PASSWORD`

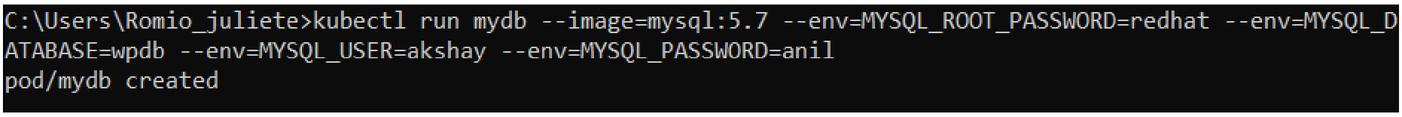
These variables are optional, used in conjunction to create a new user and to set that user's password. This user will be granted superuser permissions (see above) for the database specified by the `MYSQL_DATABASE` variable. Both variables are required for a user to be created.

Do note that there is no need to use this mechanism to create the root superuser, that user gets created by default with the password specified by the `MYSQL_ROOT_PASSWORD` variable.

lets setup the DB now…..

1

kubectl run mydb --image=mysql:5.7 --env=MYSQL_ROOT_PASSWORD=redhat --env=MYSQL_DATABASE=wpdb --env=MYSQL_USER=akshay --env=MYSQL_PASSWORD=anil

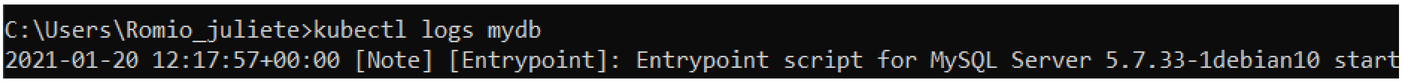


```
C:\Users\Romio_juliete>kubectl run mydb --image=mysql:5.7 --env=MYSQL_ROOT_PASSWORD=redhat --env=MYSQL_D
ATABASE=wpdb --env=MYSQL_USER=akshay --env=MYSQL_PASSWORD=anil
pod/mydb created
```

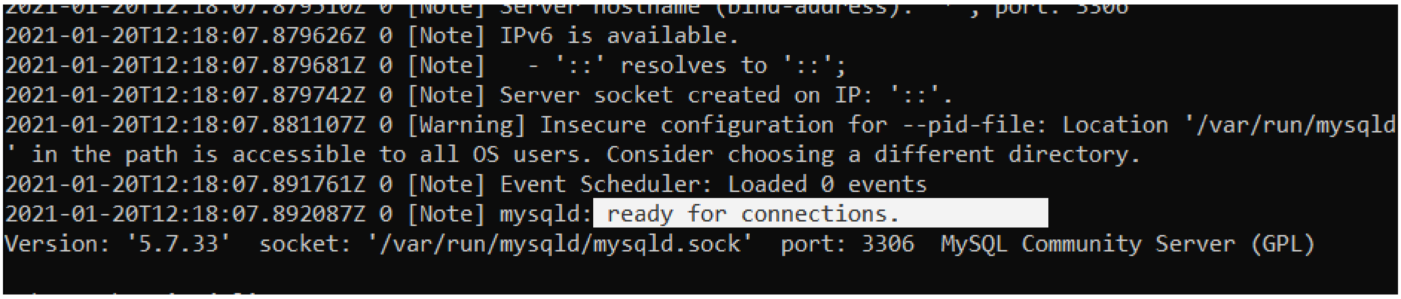
2

let’s check the logs, for any error.

kubectl logs mydb



```
C:\Users\Romio_juliete>kubectl logs mydb
2021-01-20 12:17:57+00:00 [Note] [Entrypoint]: Entrypoint script for MySQL Server 5.7.33-1debian10 start
```

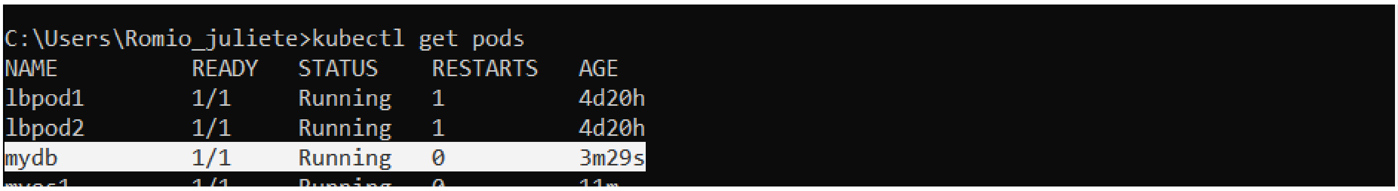


```
2021-01-20T12:18:07.879510Z 0 [Note] Server hostname (bind-address): '*'; port: 3306
2021-01-20T12:18:07.879626Z 0 [Note] IPv6 is available.
2021-01-20T12:18:07.879681Z 0 [Note]   - '::' resolves to '::';
2021-01-20T12:18:07.879742Z 0 [Note] Server socket created on IP: '::'.
2021-01-20T12:18:07.881107Z 0 [Warning] Insecure configuration for --pid-file: Location '/var/run/mysqld
' in the path is accessible to all OS users. Consider choosing a different directory.
2021-01-20T12:18:07.891761Z 0 [Note] Event Scheduler: Loaded 0 events
2021-01-20T12:18:07.892087Z 0 [Note] mysqld: ready for connections.
Version: '5.7.33'  socket: '/var/run/mysqld/mysqld.sock'  port: 3306  MySQL Community Server (GPL)
```

3.

status of running.

kubectl get pods

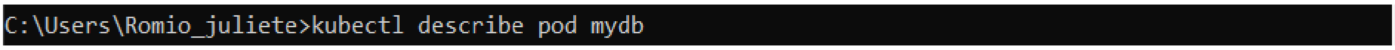


```
C:\Users\Romio_juliete>kubectl get pods
NAME           READY   STATUS    RESTARTS   AGE
lbpod1         1/1     Running   1          4d20h
lbpod2         1/1     Running   1          4d20h
mydb           1/1     Running   0          3m29s
mysql1         1/1     Running   0          11m
```

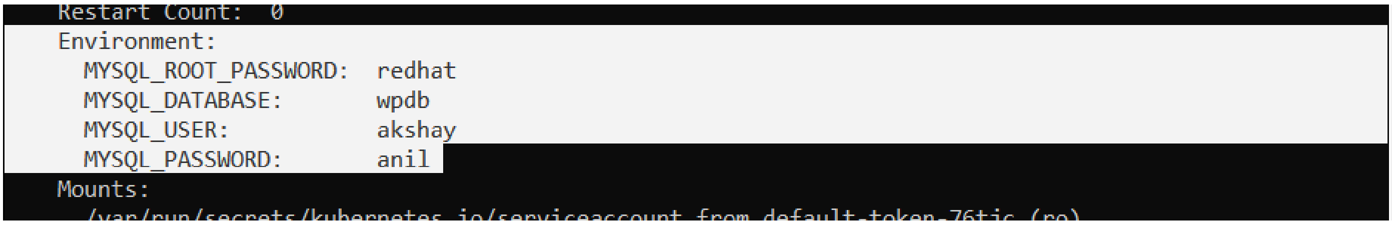
4.

checking the Env Var.

kubectl describe pod mydb



```
C:\Users\Romio_juliete>kubectl describe pod mydb
```

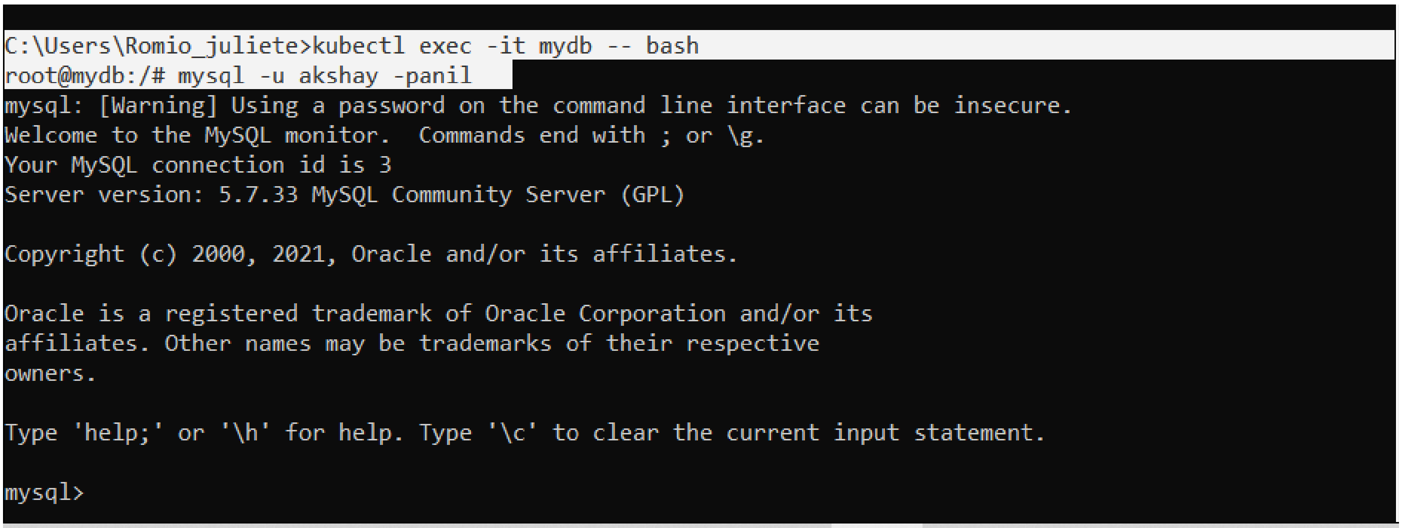


```
    Restart Count:  0
    Environment:
      MYSQL_ROOT_PASSWORD:  redhat
      MYSQL_DATABASE:       wpdb
      MYSQL_USER:           akshay
      MYSQL_PASSWORD:       anil
    Mounts:
      /var/run/secrets/kubernetes.io/serviceaccount from default-token-76tjc (ro)
```

5.

lets login into mydb : kubectl exec -it mydb – bash

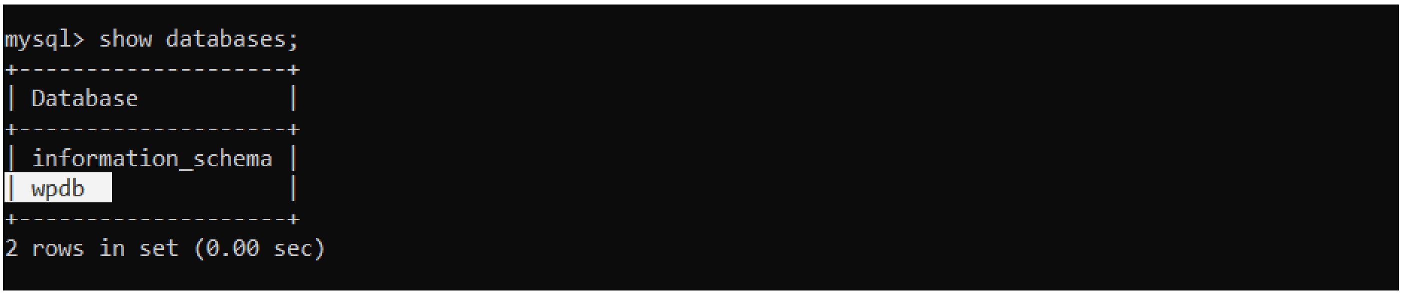
then further login in mysql installed in the pod(mydb) - mysql -u akshay -panil



```
C:\Users\Romio_juliete>kubectl exec -it mydb -- bash
root@mydb:/# mysql -u akshay -panil
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3
Server version: 5.7.33 MySQL Community Server (GPL)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>
```

lets check the databases….

mysql> show databases;



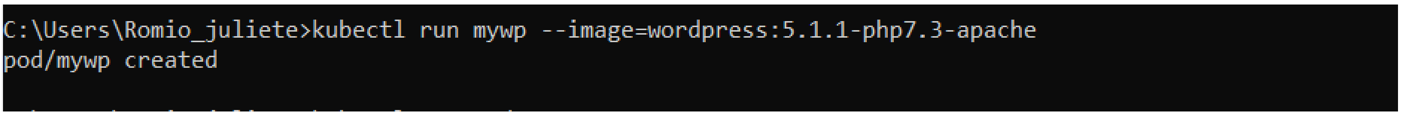
```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| wpdb               |
+--------------------+
2 rows in set (0.00 sec)
```

done…………..

lets configure the wordpress

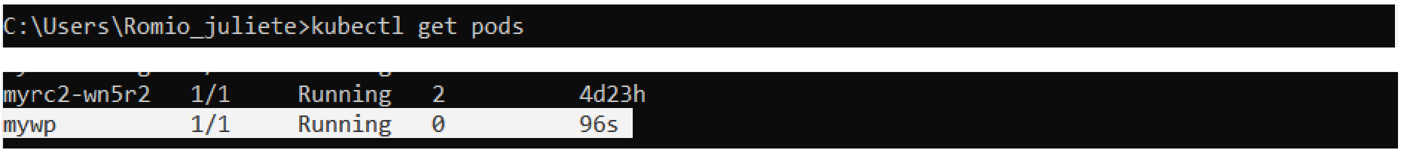
1.

kubectl run mywp –image=wordpress:5.1.1-php7.3-apache



```
C:\Users\Romio_juliete>kubectl run mywp --image=wordpress:5.1.1-php7.3-apache
pod/mywp created
```

kubectl get pods



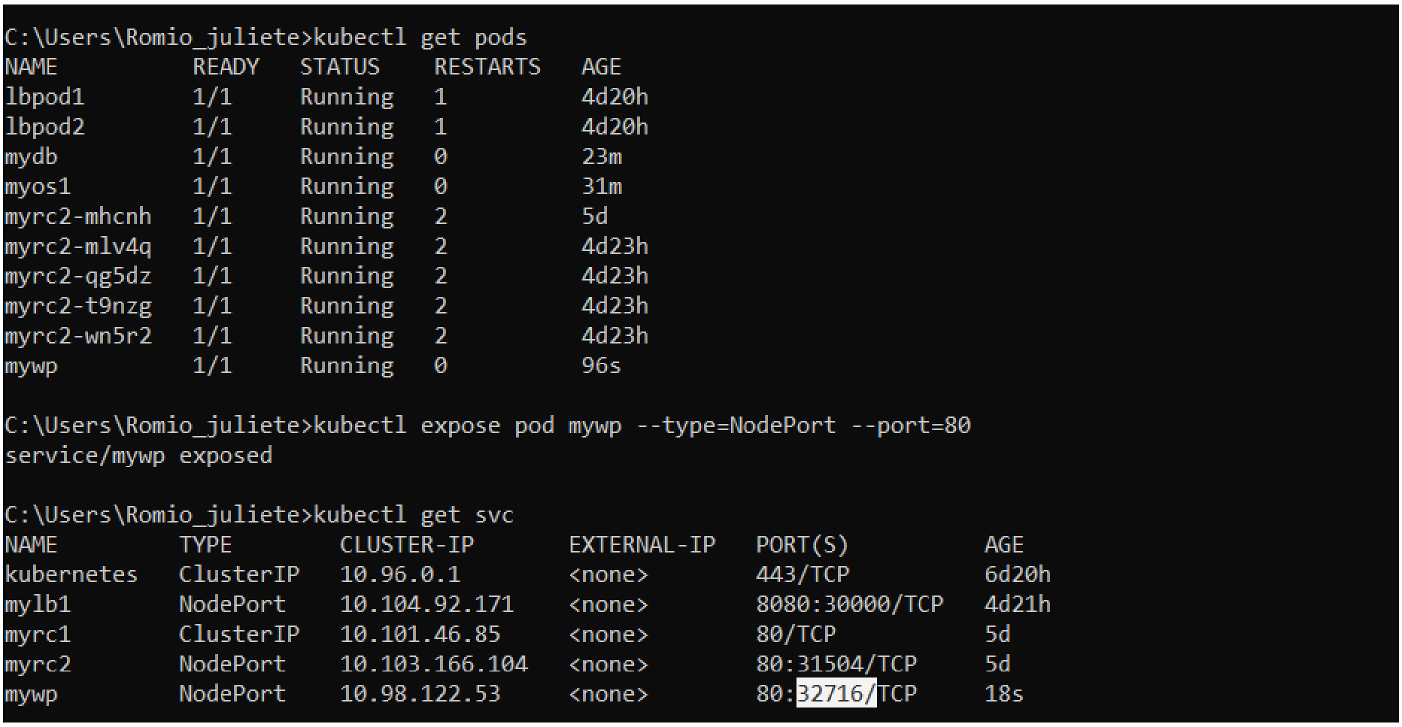
```
C:\Users\Romio_juliete>kubectl get pods
```

```
myrc2-wn5r2   1/1     Running   2          4d23h
mywp          1/1     Running   0          96s
```

2.

expose the mywp pod

kubectl expose pod mywp --type=NodePort --port=80

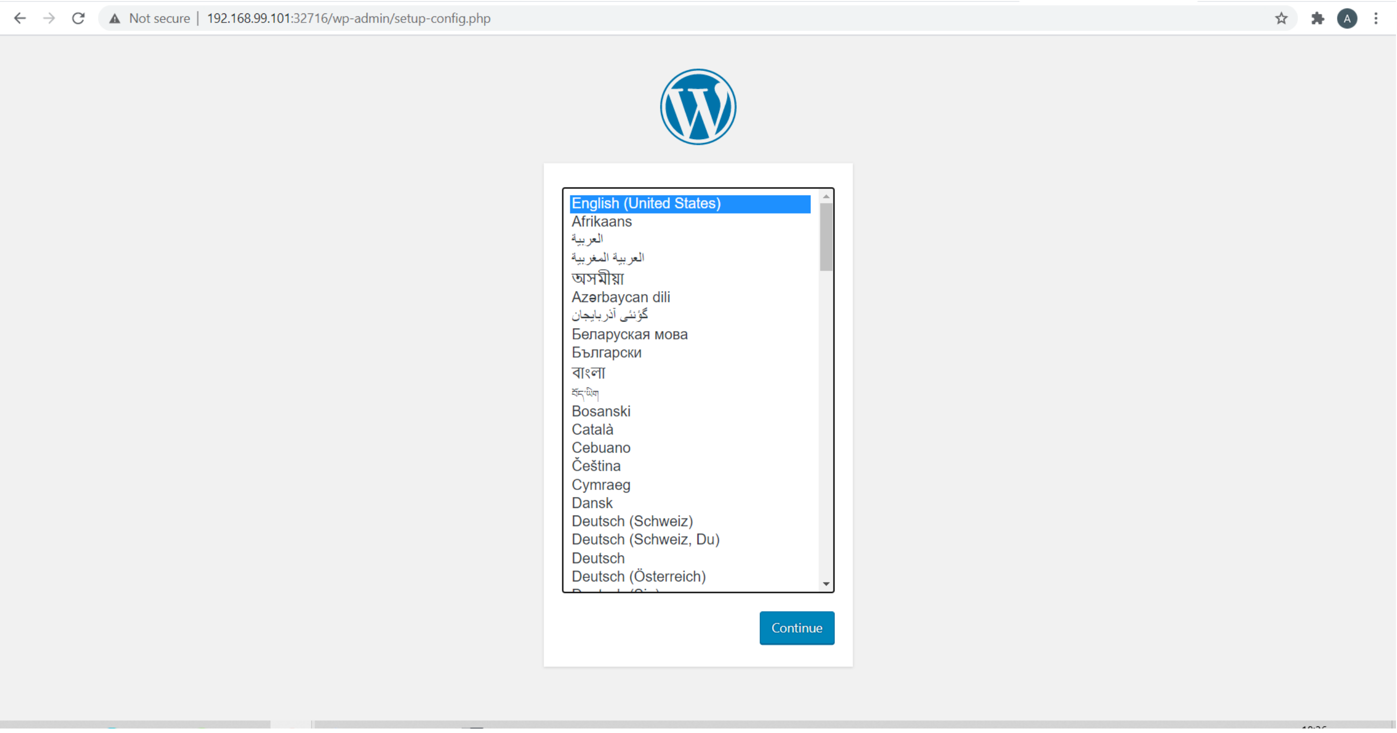


```
C:\Users\Romio_juliete>kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
lbpod1        1/1     Running   1          4d20h
lbpod2        1/1     Running   1          4d20h
mydb          1/1     Running   0          23m
myos1         1/1     Running   0          31m
myrc2-mhcnh   1/1     Running   2          5d
myrc2-mlv4q   1/1     Running   2          4d23h
myrc2-qg5dz   1/1     Running   2          4d23h
myrc2-t9nzg   1/1     Running   2          4d23h
myrc2-wn5r2   1/1     Running   2          4d23h
mywp          1/1     Running   0          96s

C:\Users\Romio_juliete>kubectl expose pod mywp --type=NodePort --port=80
service/mywp exposed

C:\Users\Romio_juliete>kubectl get svc
NAME         TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)          AGE
kubernetes   ClusterIP   10.96.0.1        <none>        443/TCP          6d20h
mylb1        NodePort    10.104.92.171    <none>        8080:30000/TCP   4d21h
myrc1        ClusterIP   10.101.46.85     <none>        80/TCP           5d
myrc2        NodePort    10.103.166.104   <none>        80:31504/TCP     5d
mywp         NodePort    10.98.122.53     <none>        80:32716/TCP     18s
```

this is the exposed port number for our frontend “ wordpress image” and the IP of minikube where k8 is installed

http://192.168.99.101:32716

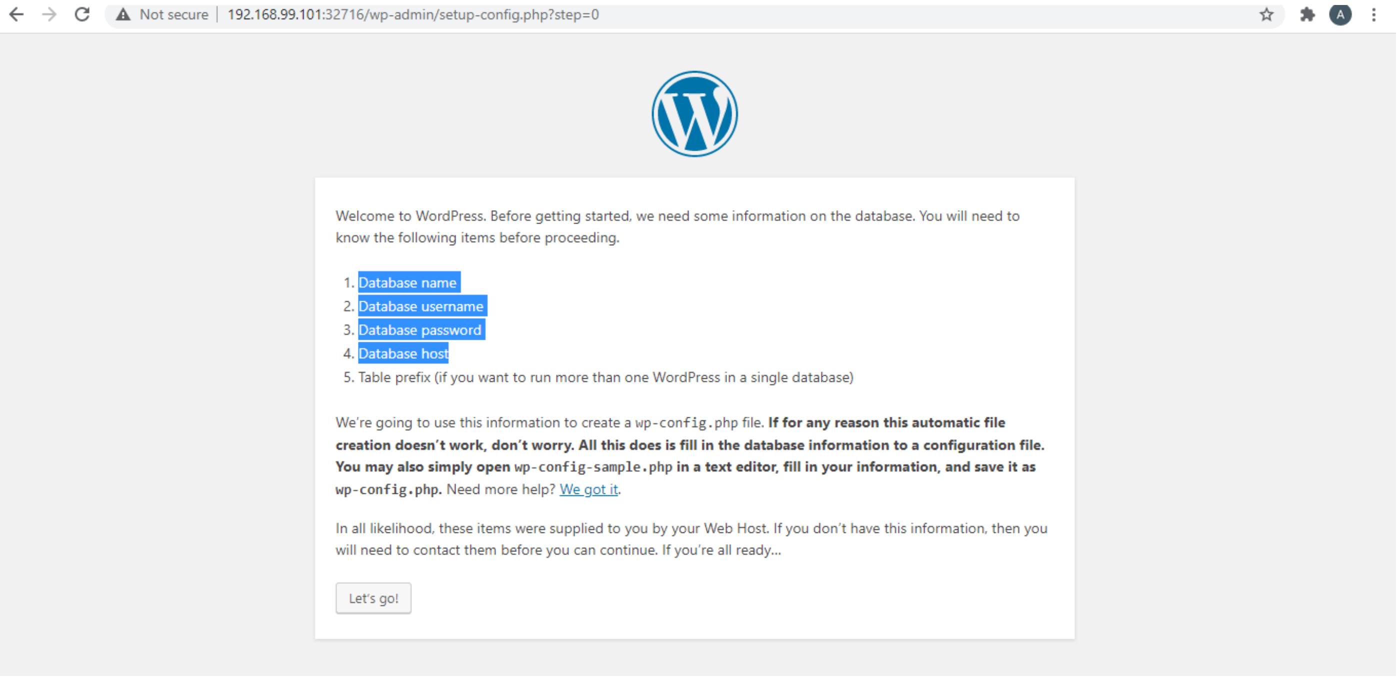
1. pod configured as frontend successfully



Not secure | 192.168.99.101:32716/wp-admin/setup-config.php

English (United States)
Afrikaans
العربية
العربية المغربية
অসমীয়া
Azərbaycan dili
گؤنئی آذربایجان
Беларуская мова
Български
বাংলা
བོད་ཡིག
Bosanski
Català
Cebuano
Čeština
Cymraeg
Dansk
Deutsch (Schweiz)
Deutsch (Schweiz, Du)
Deutsch
Deutsch (Österreich)

2. this shows the list of info we have to provide here for authentication.



Not secure | 192.168.99.101:32716/wp-admin/setup-config.php?step=0

Welcome to WordPress. Before getting started, we need some information on the database. You will need to know the following items before proceeding.

1. Database name
2. Database username
3. Database password
4. Database host
5. Table prefix (if you want to run more than one WordPress in a single database)

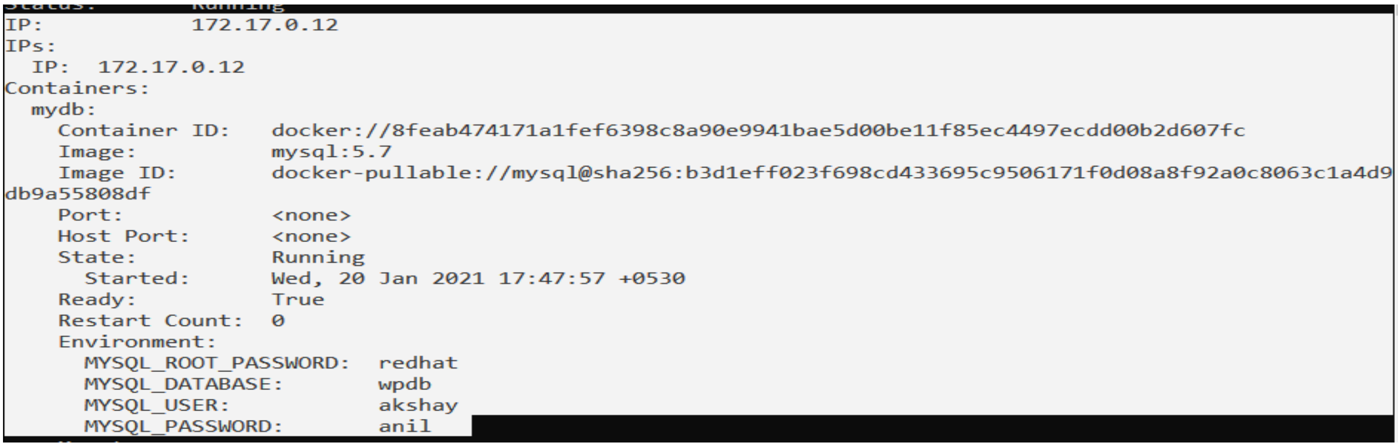
We're going to use this information to create a `wp-config.php` file. **If for any reason this automatic file creation doesn't work, don't worry. All this does is fill in the database information to a configuration file. You may also simply open `wp-config-sample.php` in a text editor, fill in your information, and save it as `wp-config.php`.** Need more help? We got it.

In all likelihood, these items were supplied to you by your Web Host. If you don't have this information, then you will need to contact them before you can continue. If you're all ready…

3.

lets view all the environmental variables we have provided :

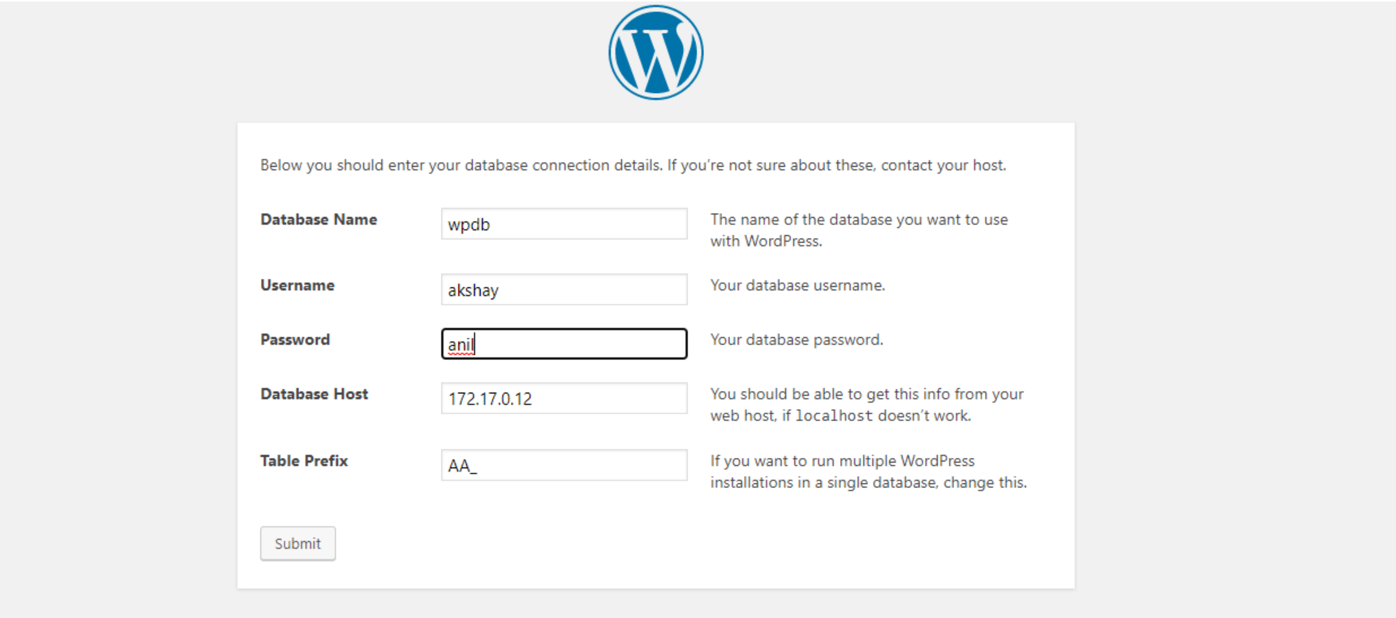
kubectl describe pod mywp



```
Status:         Running
IP:             172.17.0.12
IPs:
  IP:  172.17.0.12
Containers:
  mydb:
    Container ID:   docker://8feab474171a1fef6398c8a90e9941bae5d00be11f85ec4497ecdd00b2d607fc
    Image:          mysql:5.7
    Image ID:       docker-pullable://mysql@sha256:b3d1eff023f698cd433695c9506171f0d08a8f92a0c8063c1a4d9
db9a55808df
    Port:           <none>
    Host Port:      <none>
    State:          Running
      Started:      Wed, 20 Jan 2021 17:47:57 +0530
    Ready:          True
    Restart Count:  0
    Environment:
      MYSQL_ROOT_PASSWORD:  redhat
      MYSQL_DATABASE:       wpdb
      MYSQL_USER:           akshay
      MYSQL_PASSWORD:       anil
```

4.

apply all the information.

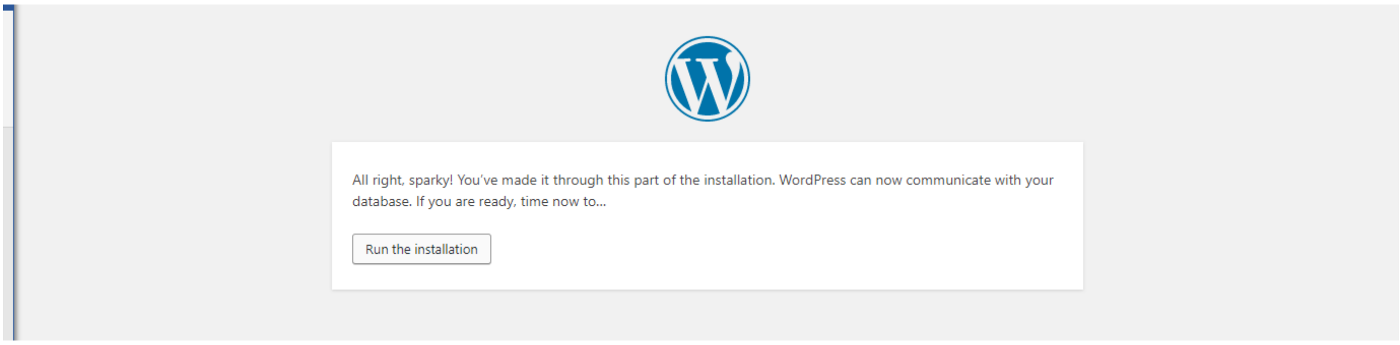


Below you should enter your database connection details. If you're not sure about these, contact your host.

Database Name	wpdb	The name of the database you want to use with WordPress.
Username	akshay	Your database username.
Password	anil	Your database password.
Database Host	172.17.0.12	You should be able to get this info from your web host, if `localhost` doesn't work.
Table Prefix	AA_	If you want to run multiple WordPress installations in a single database, change this.

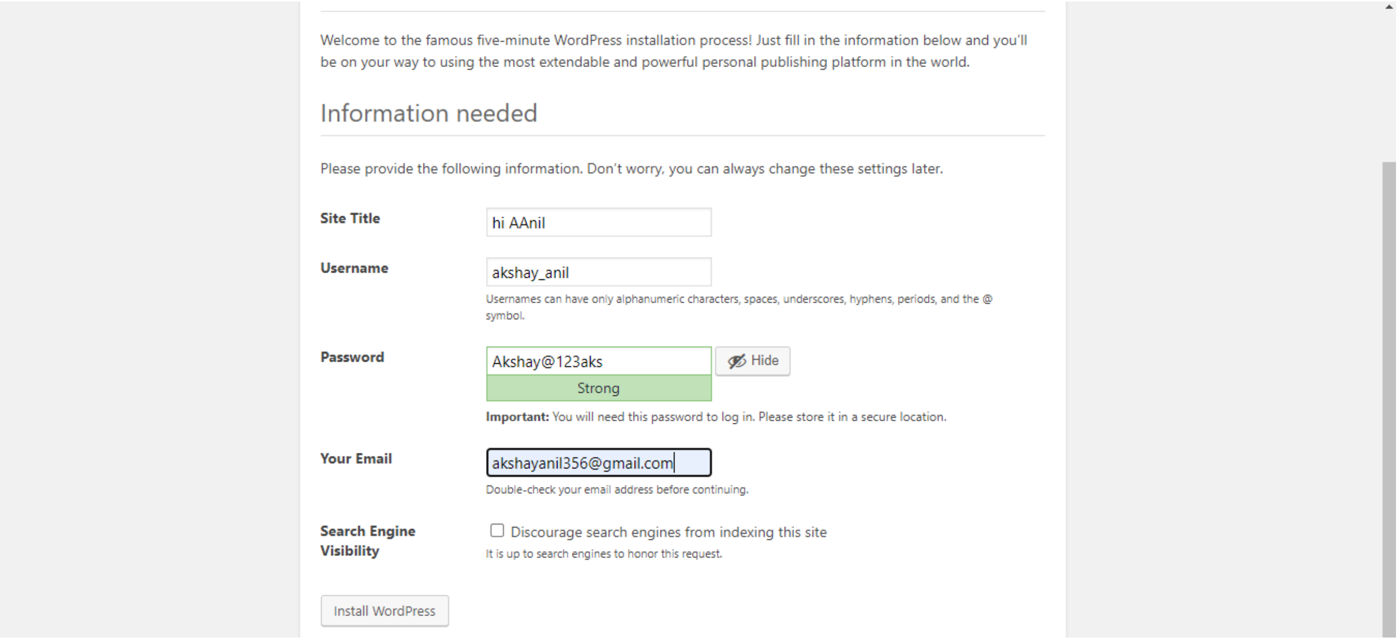
click in Submit

7. run the installation.



All right, sparky! You've made it through this part of the installation. WordPress can now communicate with your database. If you are ready, time now to…

8. set the title ,username of database, password, and give the email – this is to we are accessing the frontend web app as client.



Welcome to the famous five-minute WordPress installation process! Just fill in the information below and you'll be on your way to using the most extendable and powerful personal publishing platform in the world.

Information needed

Please provide the following information. Don't worry, you can always change these settings later.

Site Title: hi AAnil

Username: akshay_anil

Usernames can have only alphanumeric characters, spaces, underscores, hyphens, periods, and the @ symbol.

Password: Akshay@123aks Hide

Strong

Important: You will need this password to log in. Please store it in a secure location.

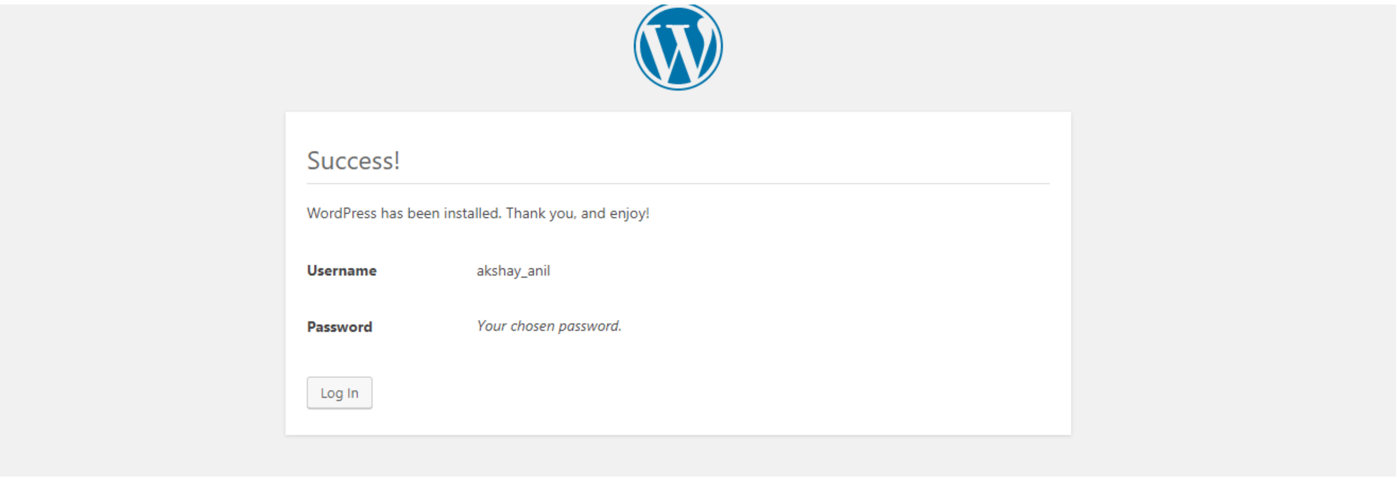
Your Email: akshayanil356@gmail.com

Double-check your email address before continuing.

Search Engine Visibility: □ Discourage search engines from indexing this site

It is up to search engines to honor this request.

9. success..



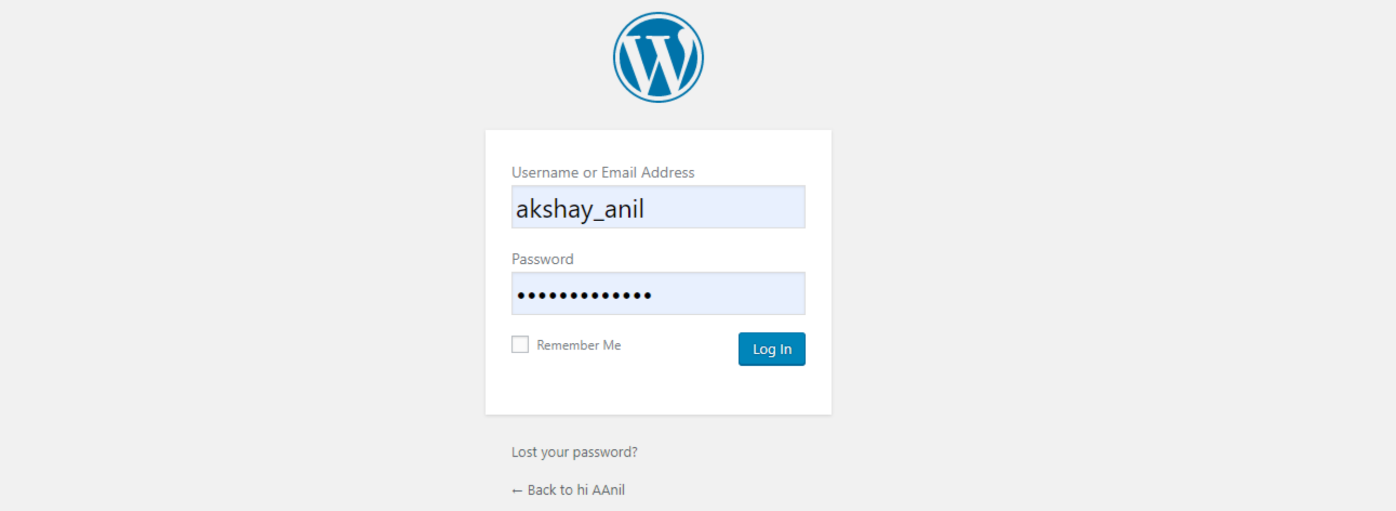
Success!

WordPress has been installed. Thank you, and enjoy!

Username: akshay_anil

Password: *Your chosen password.*

10. sign up is done. Now login..



Username or Email Address

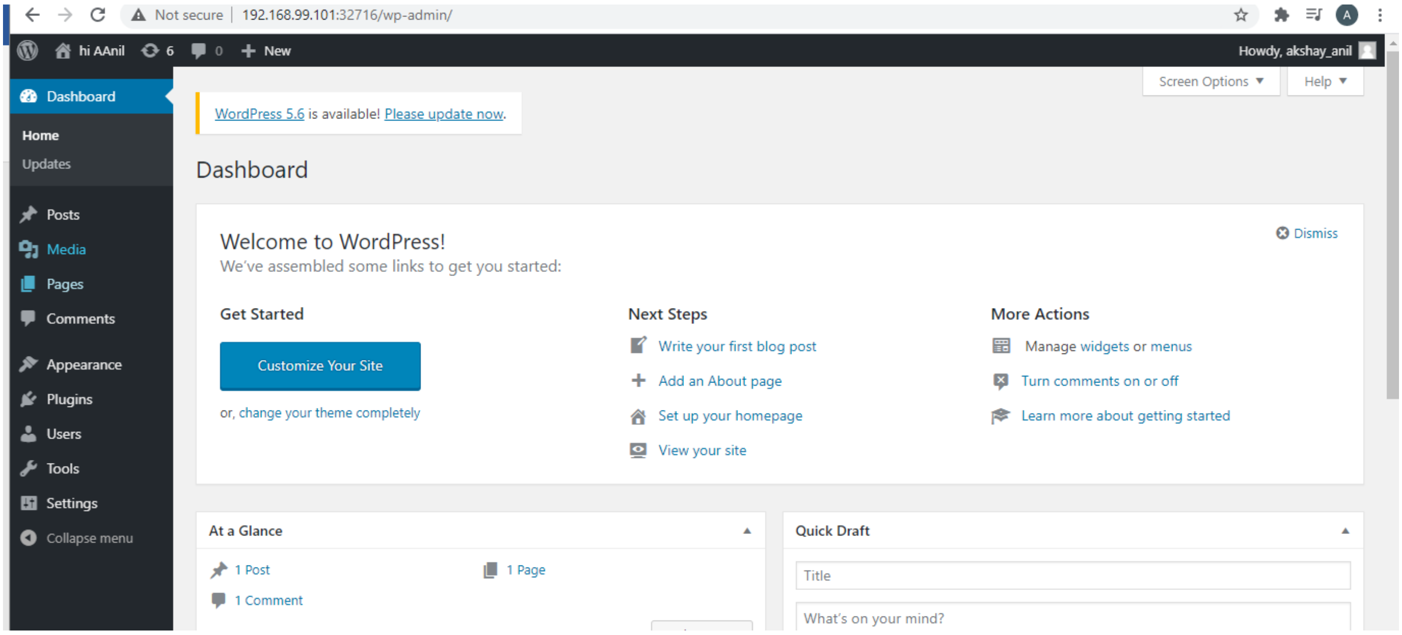
akshay_anil

Password

•••••••••••••

□ Remember Me

11. view the dashboard



Not secure | 192.168.99.101:32716/wp-admin/

WordPress 5.6 is available! Please update now.

Dashboard

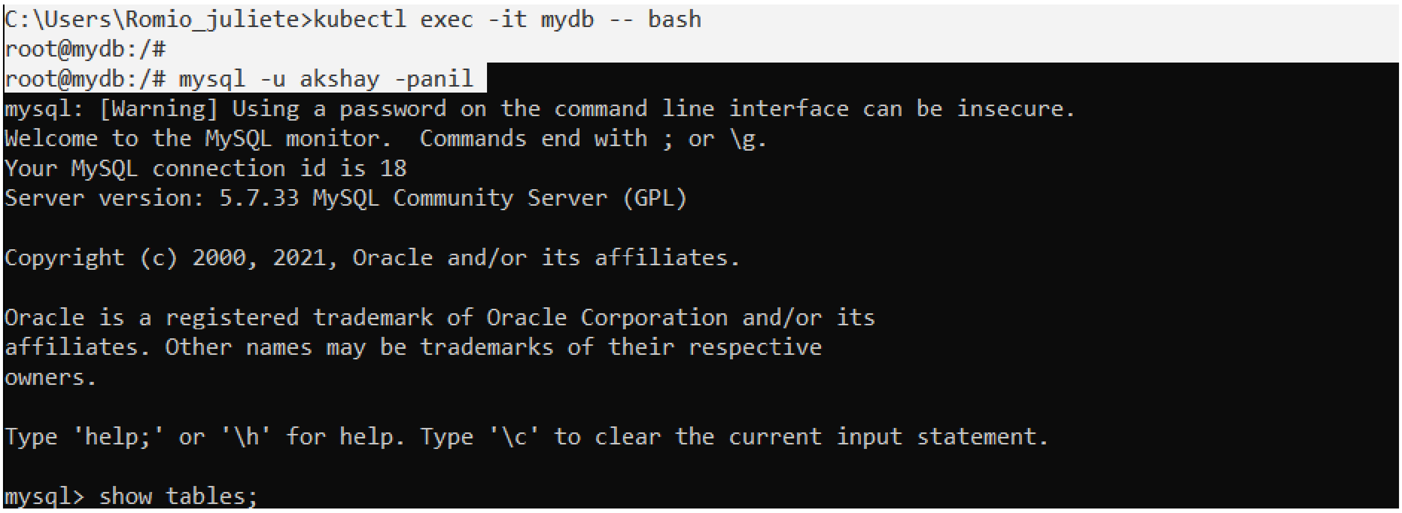
Welcome to WordPress!

We've assembled some links to get you started:

12.

login into pod and then further into mysql wpdb

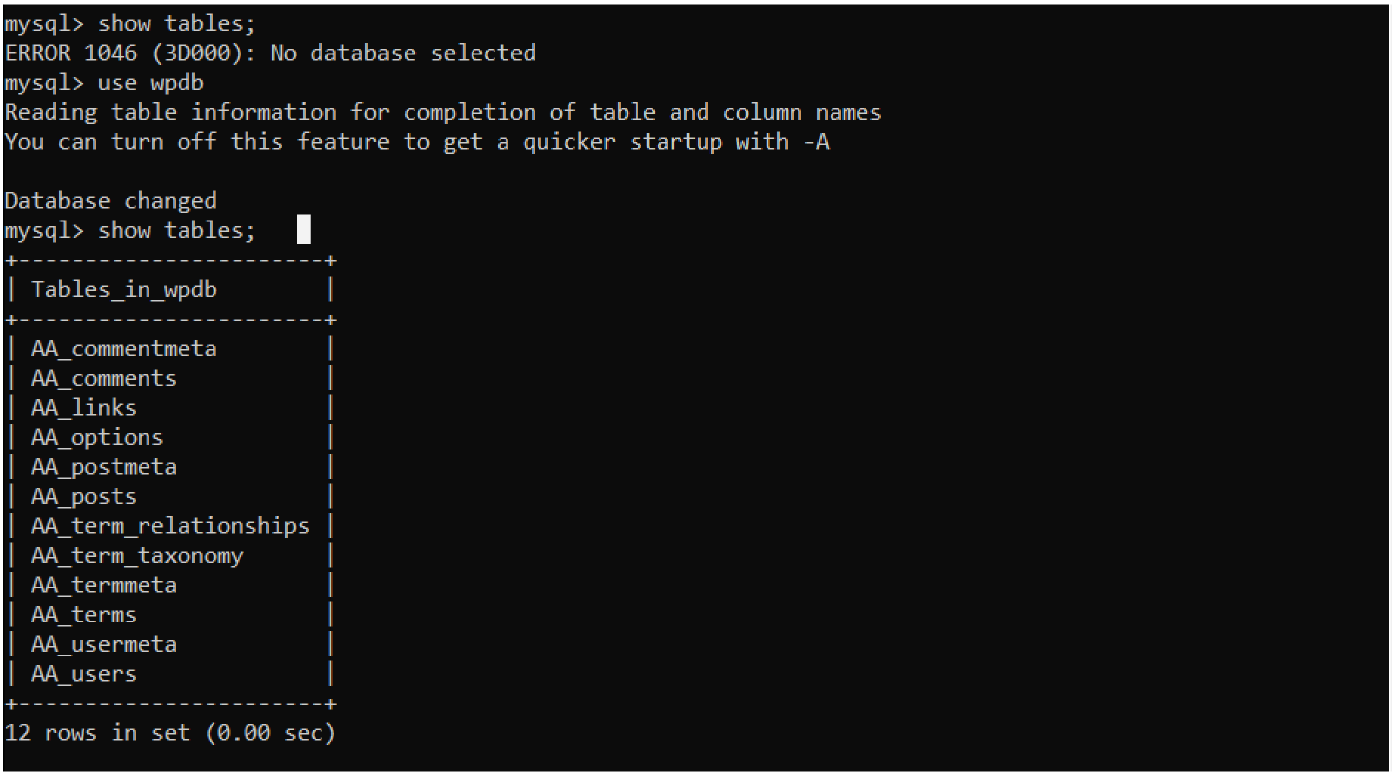
```
C:\Users\Romio_juliete>kubectl exec -it mydb -- bash
root@mydb:/# mysql -u akshay -panil
```



```
C:\Users\Romio_juliete>kubectl exec -it mydb -- bash
root@mydb:/#
root@mydb:/# mysql -u akshay -panil
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 18
Server version: 5.7.33 MySQL Community Server (GPL)

Copyright (c) 2000, 2021, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show tables;
```

use wpdb

show tables;



```
mysql> show tables;
ERROR 1046 (3D000): No database selected
mysql> use wpdb
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> show tables;
+-----------------------+
| Tables_in_wpdb        |
+-----------------------+
| AA_commentmeta        |
| AA_comments           |
| AA_links              |
| AA_options            |
| AA_postmeta           |
| AA_posts              |
| AA_term_relationships |
| AA_term_taxonomy      |
| AA_termmeta           |
| AA_terms              |
| AA_usermeta           |
| AA_users              |
+-----------------------+
12 rows in set (0.00 sec)
```

Successfully configured the multi-tier architecture with K8.

track me down: **https://akshayanil1080.github.io/mywebsite/**

and much more to go…………