google-sheet-to-db

How to save data from google sheet to mysql database using google Apps script

The task is:

```
    1. Create Mysql DB with any 10 columns
    2. Write CRUD API server in node.js to manipulate data in created at step 1 DB
via API
    3. Create 10 columns table in Google sheet, fill 3 rows with dummy data
    4. Write Google App Script application to connect spreadsheet with db via API
created in step 2
    5. You should be able to add/update all data from table to DB
```

Node.js application and mysql db server will be deployed on Red Hat Openshift cluster using openshift sendbox.

Google sheet and Google Apps Script will be on google. It is obvious

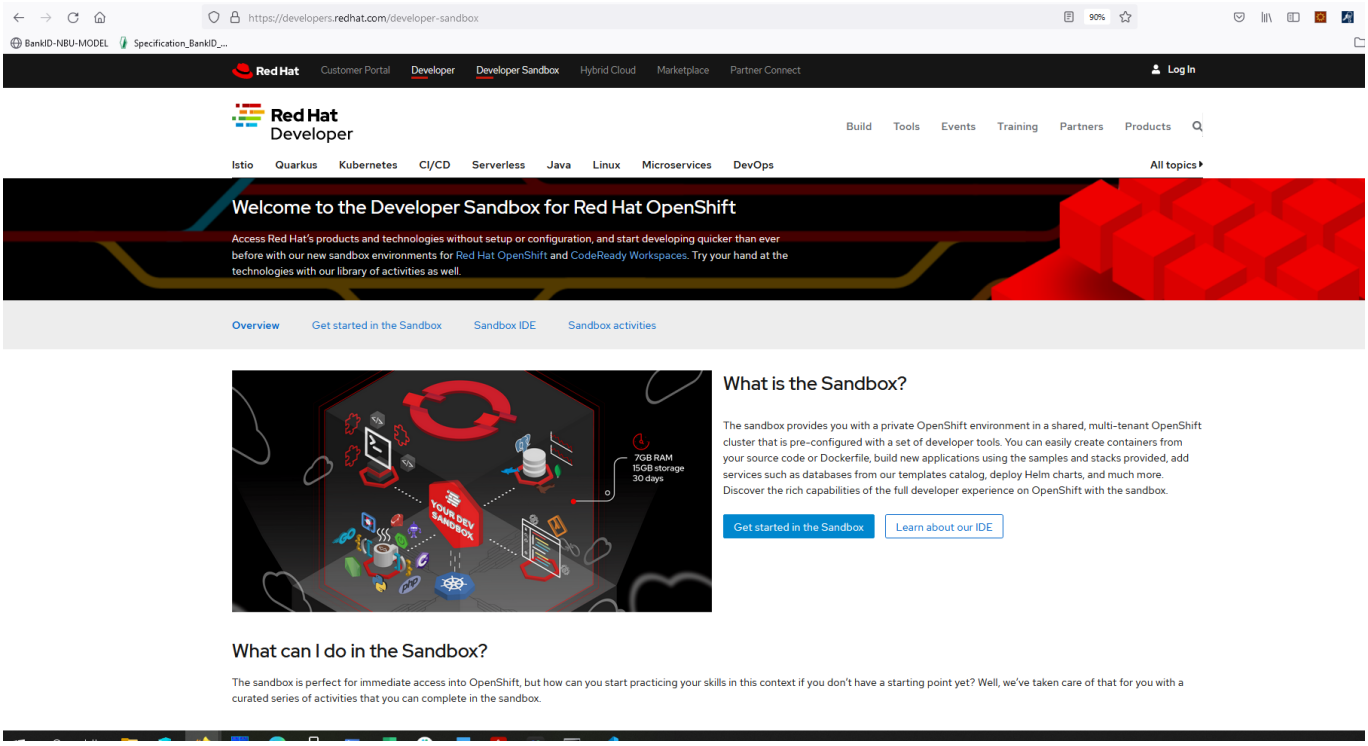
1. Create Mysql DB with any 10 columns

To acomplish this step, we must:

- create openshift sendbox;
- deploy into you senbox project MySQL server
- make conection from your laptop to MySQL server on openshift
- write DDL scritps end create database structure

create openshift sendbox

It is easy. You have to regester youself as developer on openshift sendbox pic-01.

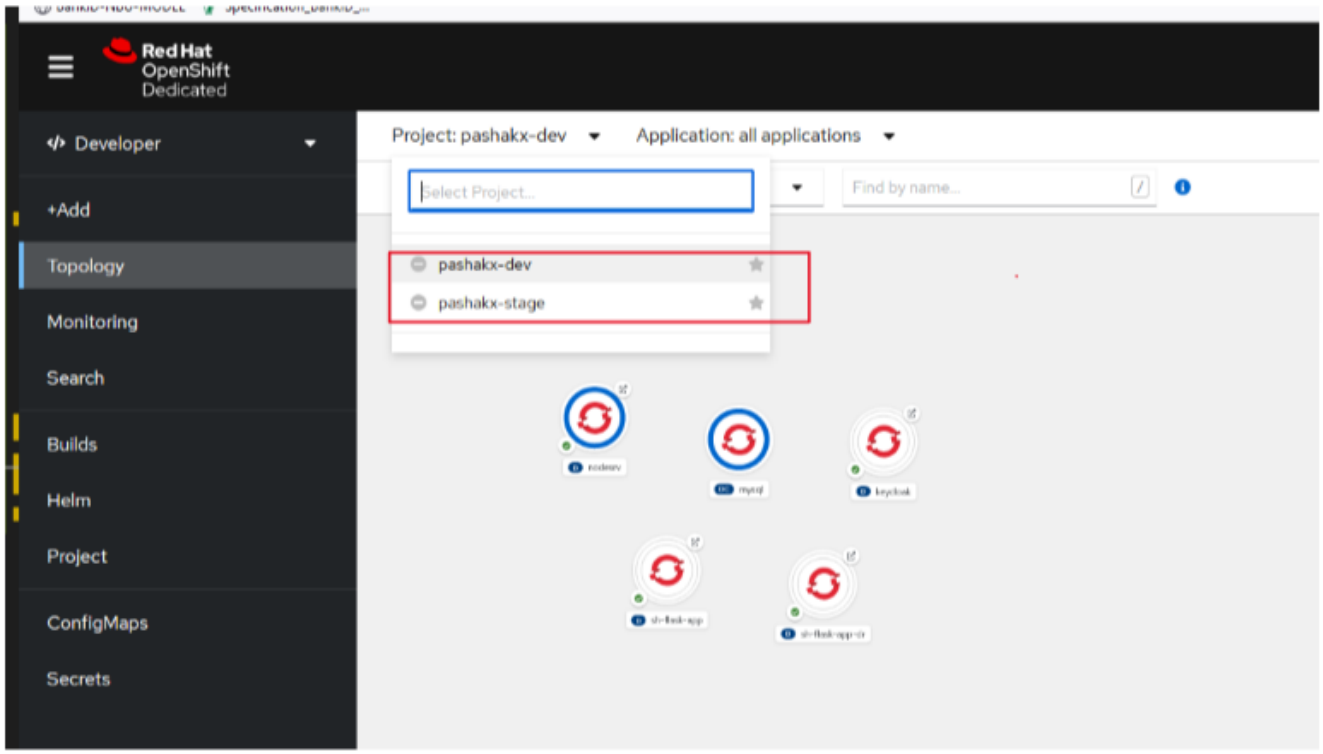


pic-01

As a result you will get openshift cluster with 2 projects (with 2 namespace in kubernetes terms).

> Q: What kind of resources do I get with my sandbox?
>
> A: Your private OpenShift environment includes two projects (namespaces) and a resource quota of 7 GB RAM, 15GB storage. The two namespaces can be used to emulate "development" and "stage" phases for your application.

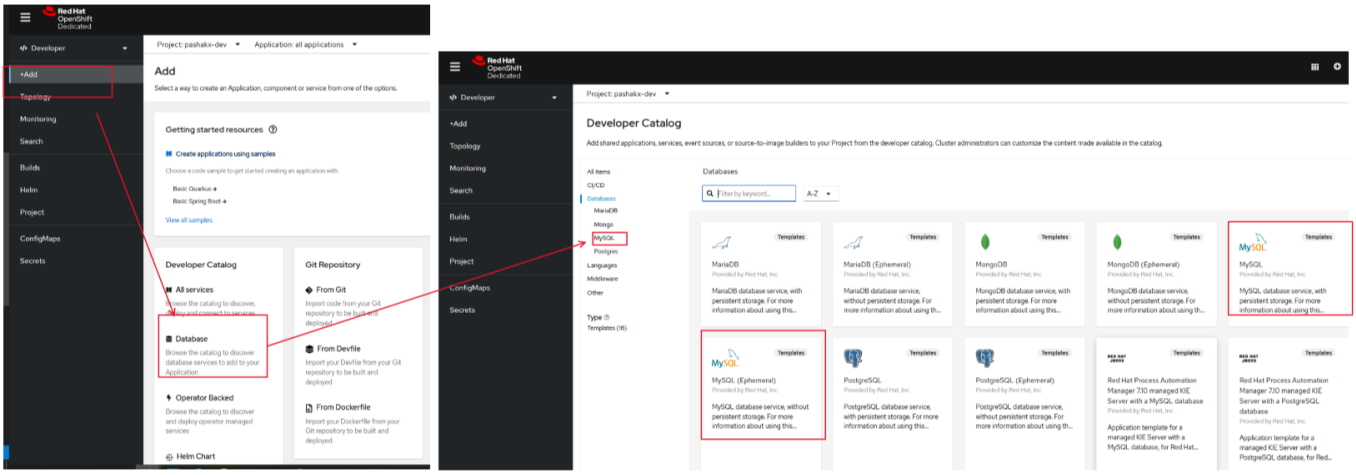
As an example, you can see on pic-02



pic-02

deploy into you senbox project MySQL server

Openshift have already had MySql database as an own template, pic-03.



pic-03

So MySQL server might be created from Openshift template using openshift CLI oc. The deployment script in filder [/openshift-deployment]](/openshift-deployment]).

- deploy database server: 1-create-mysql-db.cmd

```
rem
================================================================================
================================================
rem Create MySql DB From OPenshift template
rem
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
rem get templates:          oc get templates -n openshift
rem get template's params:  oc process --parameters -n openshift mysql-persistent
rem
================================================================================
================================================
rem      Parameters of mysql-persistent template
rem
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
rem NAME                    DESCRIPTION
GENERATOR           VALUE
rem MEMORY_LIMIT            Maximum amount of memory the container can use.
512Mi
rem NAMESPACE               The OpenShift Namespace where the ImageStream resides.
openshift
rem DATABASE_SERVICE_NAME   The name of the OpenShift Service exposed for the
database.                                 mysql
rem MYSQL_USER              Username for MySQL user that will be used for
accessing the database.   expression          user[A-Z0-9]{3}
```

```
rem MYSQL_PASSWORD          Password for the MySQL connection user.
expression          [a-zA-Z0-9]{16}
rem MYSQL_ROOT_PASSWORD     Password for the MySQL root user.
expression          [a-zA-Z0-9]{16}
rem MYSQL_DATABASE          Name of the MySQL database accessed.
sampledb
rem VOLUME_CAPACITY         Volume space available for data, e.g. 512Mi, 2Gi.
1Gi
rem MYSQL_VERSION           Version of MySQL image to be used (8.0-el7, 8.0-el8,
or latest).                           8.0-el8
rem
================================================================================
================================================

call ..\login.cmd
oc project %APP_PROJ%

echo ============================================
echo Create MySQL DB
echo ============================================

pause

oc new-app --template=openshift/mysql-persistent --param=MEMORY_LIMIT=512Mi --
param=NAMESPACE=openshift --param=DATABASE_SERVICE_NAME=mysqldb --
param=MYSQL_USER=devadm --param=MYSQL_PASSWORD=22 --param=MYSQL_ROOT_PASSWORD=22 -
-param=VOLUME_CAPACITY=1Gi -l app=mysqldb

pause
```

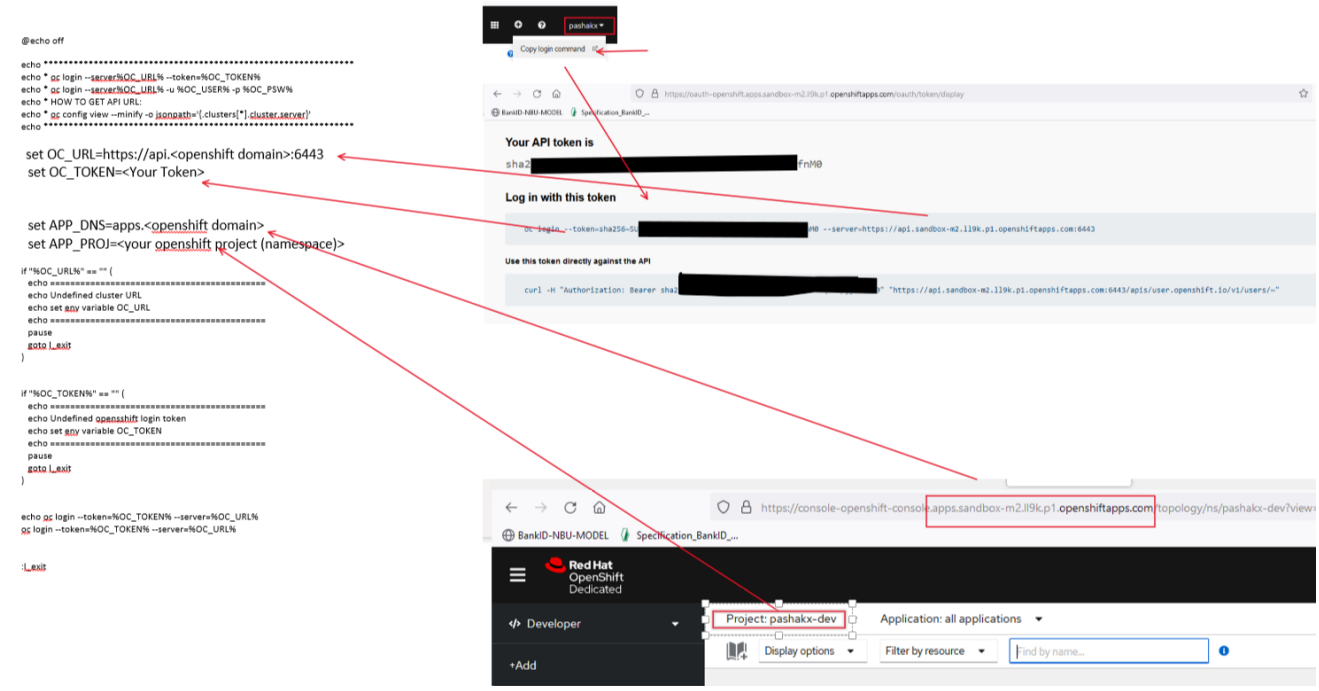
- delete deployment database server: 1-delete-mysql-db.cmd

```
rem
================================================================================
============================================
rem  Delete MySql DB
rem
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
rem
================================================================================
============================================

call ..\login.cmd
oc project %APP_PROJ%

oc delete all -l app=mysqldb
oc delete secret -l app=mysqldb
```

```
pause
```

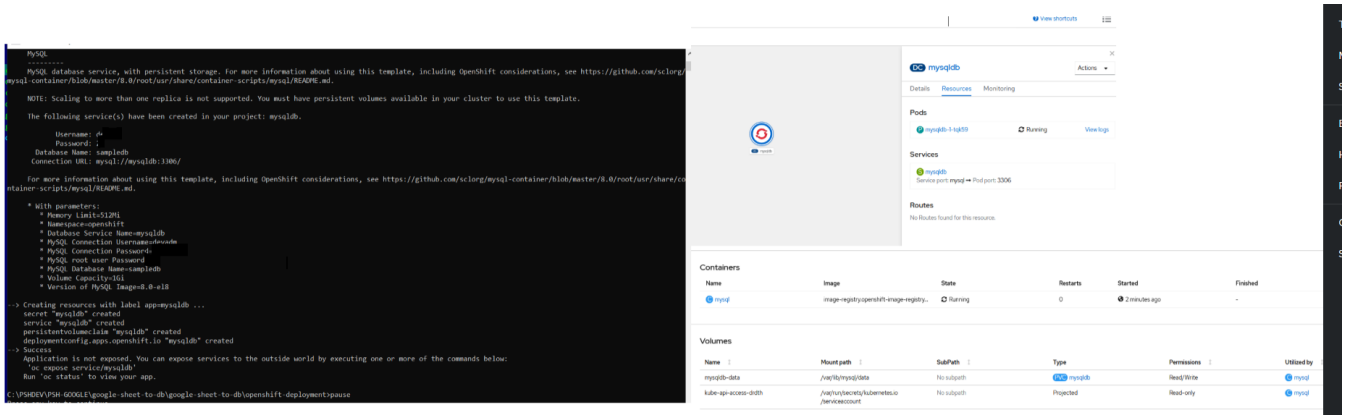
Before running any deployments script you have to add some paramters in your **login.cmd**. Which parameters and from where is shown on pic-04.

```
@echo off

echo ************************************************************
echo * oc login --server%OC_URL% --token=%OC_TOKEN%
echo * oc login --server%OC_URL% -u %OC_USER% -p %OC_PSW%
echo * HOW TO GET API URL:
echo * oc config view --minify -o jsonpath='{.clusters[*].cluster.server}'
echo ************************************************************

 set OC_URL=https://api.<openshift domain>:6443
 set OC_TOKEN=<Your Token>


 set APP_DNS=apps.<openshift domain>
 set APP_PROJ=<your openshift project (namespace)>

if "%OC_URL%" == "" (
  echo ==========================================
  echo Undefined cluster URL
  echo set env variable OC_URL
  echo ==========================================
  pause
  goto l_exit
)


if "%OC_TOKEN%" == "" (
  echo ==========================================
  echo Undefined opensshift login token
  echo set env variable OC_TOKEN
  echo ==========================================
  pause
  goto l_exit
)


echo oc login --token=%OC_TOKEN% --server=%OC_URL%
oc login --token=%OC_TOKEN% --server=%OC_URL%


:l_exit
```



pic-04

Deployment result is shown on pic-05.

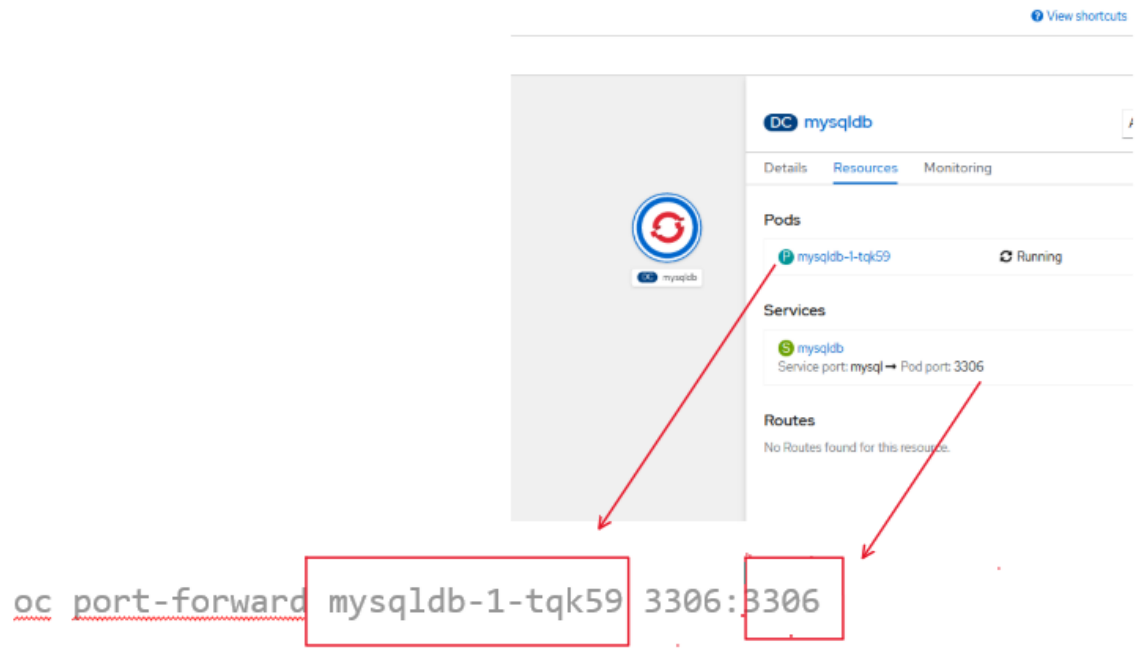


pic-05

Make conection from your laptop to MySQL server on openshift

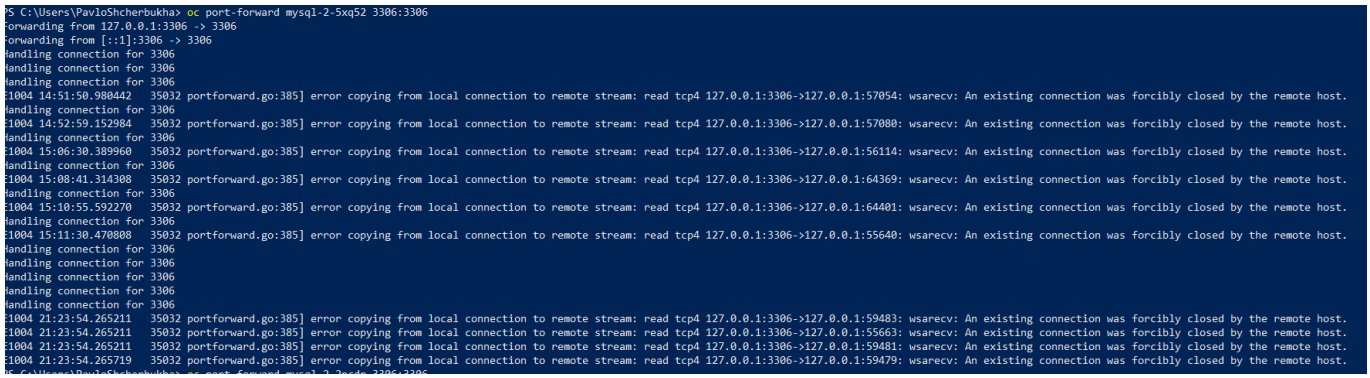
We can connect to db using port forward command pic-06.

```
# oc port-forward  <your pod>  <your local port> : <your remote port>
oc login --token=%OC_TOKEN% --server=%OC_URL%
oc project <your project>
oc port-forward mysqldb-1-tqk59 3306:3306
```



pic-06

If you run this commands from PowerShell you will see something like on pic-7.



pic-07

Let's check connection from your laptop to MySQL on openshift, using **mysql.exe**. I had stopped my local MySQL server before action which is described below. You can find in folder **ddl** file **mysqlrun2.cmd** which make connection to database server as a root. In case of successful connection I must run ddl **db-grn.sql** which grant permittion for user **devadm**.

```
C:\PSHDEV\PSH-GOOGLE\google-sheet-to-db\google-sheet-to-db\ddl>mysqlrun2.cmd
=========================================================
RUN  DDL, DML from  MySQL CLI
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

C:\PSHDEV\PSH-GOOGLE\google-sheet-to-db\google-sheet-to-db\ddl>mysql.exe  -uroot -
p22 --default-character-set=utf8mb4 -v --port 3306
```

```
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 383
Server version: 8.0.21 Source distribution

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> source db-grn.sql;
--------------
grant super on *.* to 'devadm'@'%'
--------------

Query OK, 0 rows affected, 1 warning (0.18 sec)

--------------
show grants for 'devadm'@'%'
--------------

+-----------------------------------------------------+
| Grants for devadm@%                                 |
+-----------------------------------------------------+
| GRANT SUPER ON *.* TO `devadm`@`%`                  |
| GRANT ALL PRIVILEGES ON `sampledb`.* TO `devadm`@`%` |
+-----------------------------------------------------+
2 rows in set (0.18 sec)

mysql>
```

Then, you can connect under user 'DEVADM' and create database. Then, you can connect under user 'DEVADM' and create database. Connection under **DEVADM** will be created using **mysqlrun1.cmd**. Database structure stored in **db-build.sql**. Let's run it:

```
C:\PSHDEV\PSH-GOOGLE\google-sheet-to-db\google-sheet-to-db\ddl>mysqlrun1.cmd
=========================================================
RUN  DDL, DML from  MySQL CLI
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

C:\PSHDEV\PSH-GOOGLE\google-sheet-to-db\google-sheet-to-db\ddl>mysql.exe  -udevadm
-p22 --default-character-set=utf8mb4 -v --port 3306
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 485
Server version: 8.0.21 Source distribution
```

```
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> source db-build.sql;


mysql> source db-build.sql;
Logging to file 'db-build.log'
--------------
drop database IF EXISTS test4
--------------

Query OK, 0 rows affected, 1 warning (0.18 sec)

--------------
create database test4
--------------

Query OK, 1 row affected (0.18 sec)

--------------
show databases
--------------

+--------------------+
| Database           |
+--------------------+
| information_schema |
| sampledb           |
| test4              |
+--------------------+
3 rows in set (0.17 sec)

Database changed
--------------
CREATE TABLE APP2$EMP
(
    IDREC    INT AUTO_INCREMENT,
.
.
.
.
.
.
```

Also, let's insert test data into the table **APP2$EMP**. Run script in the file **data-ins1.sql**:

```
mysql> source data-ins1.sql;
Logging to file 'data-ins1.log'
Database changed
--------------
delete from APP2$EMP
--------------

Query OK, 0 rows affected (0.17 sec)

--------------
insert into APP2$EMP( CODEBRN,
                      NAMEBRN,
                      TABNUM ,
                      FAM,
                      IM,
                      OTCH,
                      ADRESS,
                      MSTATUS,
                      COUNTRY,
                      DS,
                      DF)
VALUES (
                      '00',
                      'ГОЛОВНИЙ',
                      '00001',
                      'ВАСЕЧКИН',
                      'ПЕТРО',
                      'ПЕТРОВИЧ',
                      'На розі біля цирку',
                      'M',
                      'UA',
                      '2005-03-09',
.
.
.
.
.
.

--------------
select A.* from  APP2$EMP A
--------------

+-------+---------+-----------------------+--------+------------------+----------
----------+----------------------+-----------------------------------+---------+--
-------+------------+------+--------------------+------------+------+--------+
| IDREC | CODEBRN | NAMEBRN               | TABNUM | FAM              | IM
| OTCH                | ADRESS                            | MSTATUS | COUNTRY |
DS         | DF   | IDT                | IUSRNM     | MDT  | MUSRNM |
+-------+---------+-----------------------+--------+------------------+----------
```

```
----------+----------------------+-----------------------------------+---------+--
-------+------------+------+--------------------+------------+------+--------+
|     1 | 00      | ГОЛОВНИЙ               | 00001  | ВАСЕЧКИН         | ПЕТРО
| ПЕТРОВИЧ             | На розі біля цирку                | M       | UA      |
2005-03-09 | NULL | 2021-10-04 21:54:26 | devadm@::1 | NULL | NULL   |
|     2 | 00      | ГОЛОВНИЙ               | 00002  | ПЕТРЕНКО         | СЕМЕН
| СЕМЕНОВИЧ            | БІЛЯ ПАРКУ                        | M       | UA      |
2007-02-03 | NULL | 2021-10-04 21:54:27 | devadm@::1 | NULL | NULL   |
|     3 | 01      | ЦЕНТРАЛЬНИЙ            | 00003  | САЄНКО           | МАРГАРИТА
| СЕРГІЇВНА            | БІЛЯ ТЕАТРУ                       | M       | UA      |
2007-02-03 | NULL | 2021-10-04 21:54:27 | devadm@::1 | NULL | NULL   |
|     4 | 01      | ЦЕНТРАЛЬНИЙ            | 00004  | ДУДКА            | АНАСТАСІЯ
| ВІКТОРІВНА           | БІЛЯ ТЕАТРУ                       | S       | UA      |
2019-07-23 | NULL | 2021-10-04 21:54:27 | devadm@::1 | NULL | NULL   |
+-------+---------+----------------------+--------+------------------+----------
----------+----------------------+-----------------------------------+---------+--
-------+------------+------+--------------------+------------+------+--------+
4 rows in set (0.17 sec)

mysql>
```

Finally, the database **test4** created. Test data inserted. So, this step is finished.

2. Write CRUD API server in node.js to manipulate data in created at step 1 DB via API