



---

# YOUTH HOMELESSNESS PROJECT FALL 2022 MANUAL

---



DUNCAN LI  
NICK ASTOR  
ANGELINA PERSAUD  
STRAWBERRY ANN TRINIDAD  
MAYSON FARRAR  
DAVID GONZALEZ  
GAGE VAN KUILENBURG  
BELLINDTON CAYO

## Table Of Content

<b>Process</b>	<b>Description</b>	<b>Page</b>
Setting up MYSQL	Step-by-step guide for MYSQL setup.	<b>3</b>
Importing SQL Scripts	Step-by-step guide on importing SQL scripts into MYSQL workbench.	<b>13</b>
MYSQL to AWS Quicksight	Step-by-step guide on importing MySQL data into quick sight for visualized insights with charts or graphs.	<b>16</b>
AWS Beanstalk Deployment	Step-by-step guide on deploying to AWS Beanstalk, accessible to anyone on the internet outside your local machine.	<b>22</b>

### **Introduction**

The fall 2022 team picked up from where the Summer 2022 project team left off. The project goal is to provide housing, dependent, and food resource information for students in the Orlando area. The web app is meant to help students find the proper resources through surveys. Based on the questions answered on the survey and the zip code provided by the student, the app provides available resources.

### **Summer 2022**

The summer team used Spring boot to get the project started more quickly. Some advantages of spring-boot are that it makes deployment easier, has simplified scalability, is compatible with containers, has a little configuration, and minimizes time in production. The web app uses Rest API and CRUD commands to create, read, update, and delete users, surveys, and resource information. Also, the web app uses JPA hibernate to persist data to a MySQL database.

## Setting Up MySQL

**Step 1:** Go to <https://www.mysql.com> and click “downloads”

The screenshot shows the MySQL website at [mysql.com](https://www.mysql.com). The page features the MySQL logo and navigation links for MySQL.COM, DOWNLOADS, DOCUMENTATION, and DEVELOPER ZONE. The 'DOWNLOADS' link is highlighted with a yellow background. The top right corner includes links for Contact MySQL, Login, and social media (Facebook, Twitter, LinkedIn).

**Step 2:** Select “MySQL Community (GPL) Downloads >>”

The screenshot shows the MySQL Downloads page at [mysql.com/downloads/](https://mysql.com/downloads/). The left sidebar contains links for MySQL Newsletter, Free Webinars, What's New with MySQL, Contact Sales, and a contact phone number. The main content area lists MySQL Enterprise Edition, MySQL Cluster CGE, and MySQL Cluster. At the bottom of the content area, a yellow box highlights the 'MySQL Community (GPL) Downloads >>' link.

**Step 3:** Select “MySQL Community Server”

The screenshot shows the MySQL Community Downloads page at [dev.mysql.com/downloads/](https://dev.mysql.com/downloads/). The left sidebar lists various MySQL products: MySQL Yum Repository, MySQL APT Repository, MySQL SUSE Repository, MySQL Community Server (which is highlighted with a yellow box), MySQL Cluster, MySQL Router, MySQL Shell, MySQL Operator, MySQL NDB Operator, MySQL Workbench, MySQL Installer for Windows, and MySQL for Visual Studio. The right sidebar lists additional MySQL components: C API (libmysqlclient), Connector/C++, Connector/J, Connector/.NET, Connector/Node.js, Connector/ODBC, Connector/Python, MySQL Native Driver for PHP, MySQL Benchmark Tool, Time zone description tables, and Download Archives.

**Step 4:** Select the most appropriate download for your OS. In this example, macOS was used.

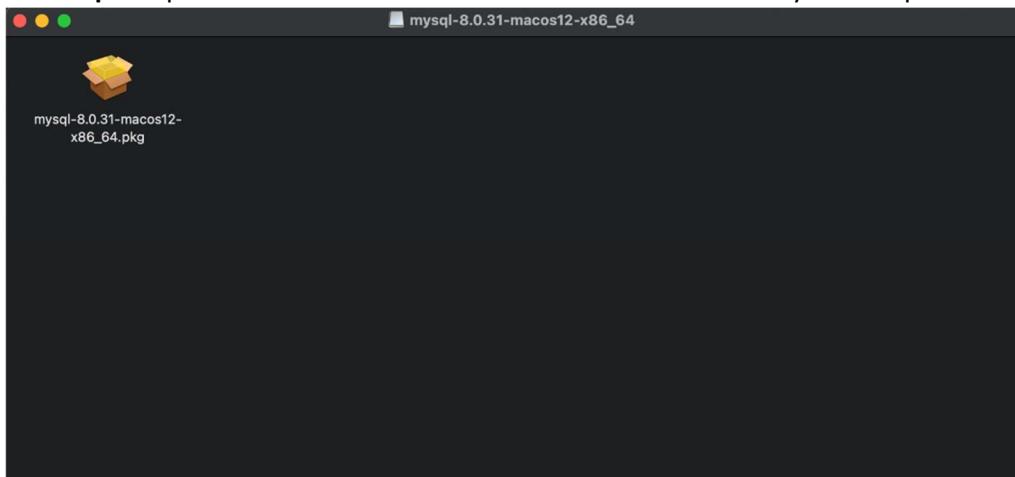
The screenshot shows the MySQL Community Server 8.0.31 download page. The operating system is set to "macOS" and the version is "All". A note says "Packages for Monterey (12) are compatible with Ventura (13) and Big Sur (11)". There are four download options listed:

- macOS 12 (ARM, 64-bit), DMG Archive**: Version 8.0.31, 454.9M, Download button
- macOS 12 (x86, 64-bit), DMG Archive**: Version 8.0.31, 461.8M, Download button
- macOS 12 (ARM, 64-bit), Compressed TAR Archive**: Version 8.0.31, 172.3M, Download button
- macOS 12 (x86, 64-bit), Compressed TAR Archive**: Version 8.0.31, 178.1M, Download button

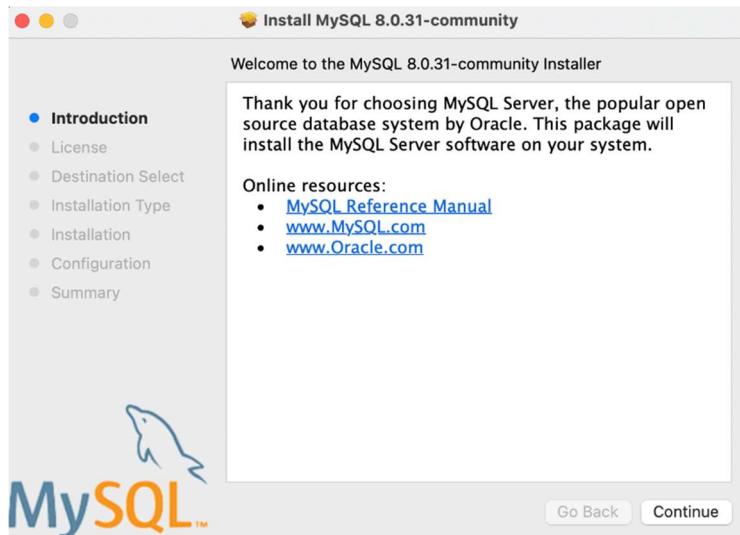
**Step 5:** You can either choose to sign up for a free account or select “No thanks, just start my download.”

The screenshot shows the MySQL Community Downloads page. It features two main buttons: "Login >" and "Sign Up >". Below these buttons, it says "MySQL.com is using Oracle SSO for authentication. If you already have an Oracle Web account, click the Login link. Otherwise, you can signup for a free account by clicking the Sign Up link and following the instructions." At the bottom, there is a link "No thanks, just start my download."

**Step 6:** Open folder from browser or find download folder on your computer



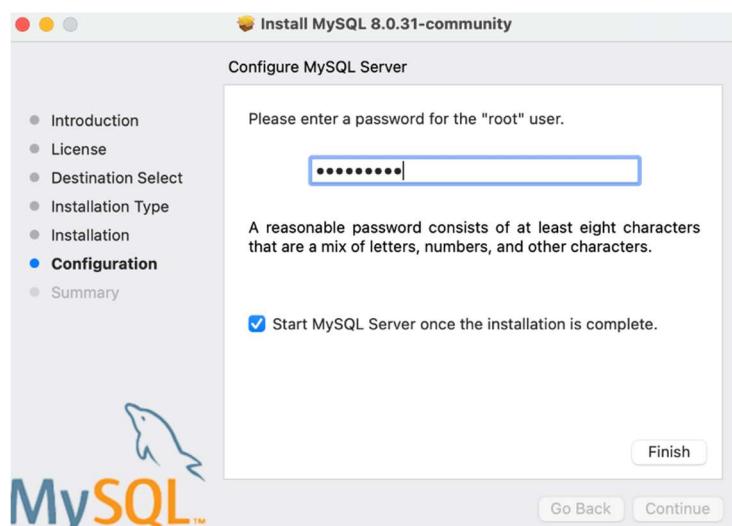
**Step 7:** MySQL installer will open up. Click through each section. No changes are necessary.



**Step 8:** Next, it will ask you to create a password for your MySQL server. Select “Use Strong Password Encryption”



**Step 9:** Create password for “root” user, then select “Finish”



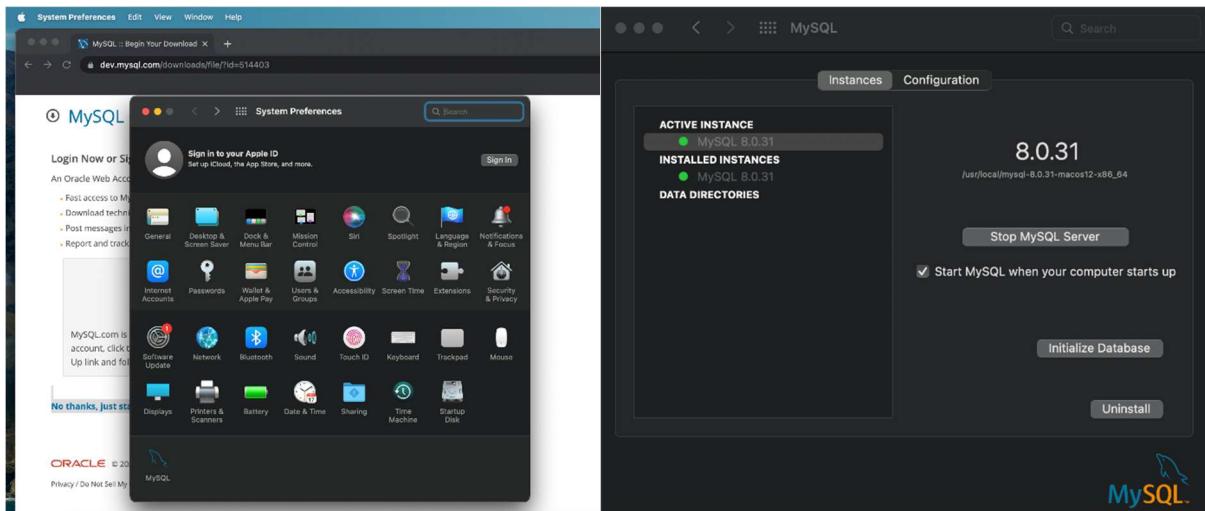
**Step 10:** The installation should now be completed successfully. Close installer window.



**Step 11:** Ensure that the MySQL server is running

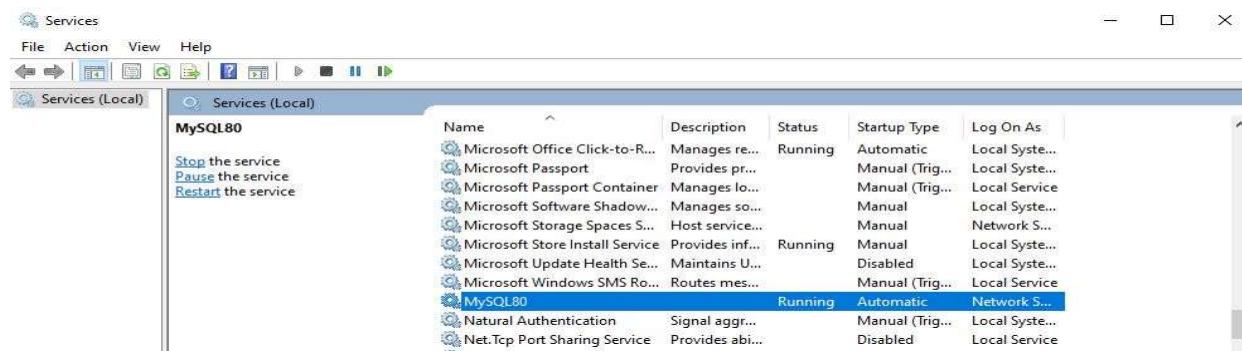
macOS:

Navigate to the Apple icon in the top left hand side of the desktop, select “System Preferences” and select “MySQL” at the bottom of the window. From here, you should see an active instance running, indicated by a green dot.



Windows:

Search for “Services” on your computer, then find “MySQL80” from the list. It should state “running” if the MySQL instance is running.



**Step 12:** Add MySQL to Path

macOS:

Open the terminal and type “ls -a”. Ensure that there is a “bash\_profile” file. If there is not, you can create one by typing “touch .bash\_profile”

```
Last login: Sun Nov  6 09:19:27 on ttys000
mayson@Maysons-Air ~ % ls -a
.                                .docker          Downloads
..                               .ssh             Library
.CFUserTextEncoding      .vscode          Movies
.DS_Store                  .zsh_history    Music
.Trash                     .zsh_sessions  Pictures
.base_profile              Applications   Public
.bash_history              Desktop        getting-started
.bash_profile               Documents
mayson@Maysons-Air ~ %
```

Once you know you have the bash\_profile file on your machine, open up the file from the terminal by typing “open -t .bash\_profile”. From here, you want to ensure to add the path to SQL with the export command below.

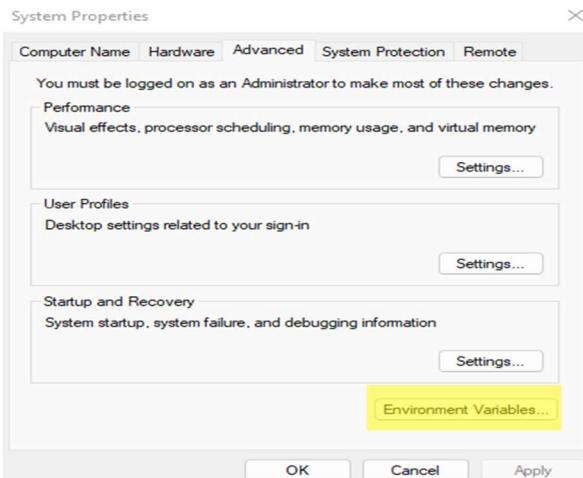
```
export PATH=${PATH}:/usr/local/mysql/bin/
```

From here, you want to connect to your MySQL. Type “/usr/local/mysql/bin/mysql -u root -p” then enter the root password that was created in the installer.

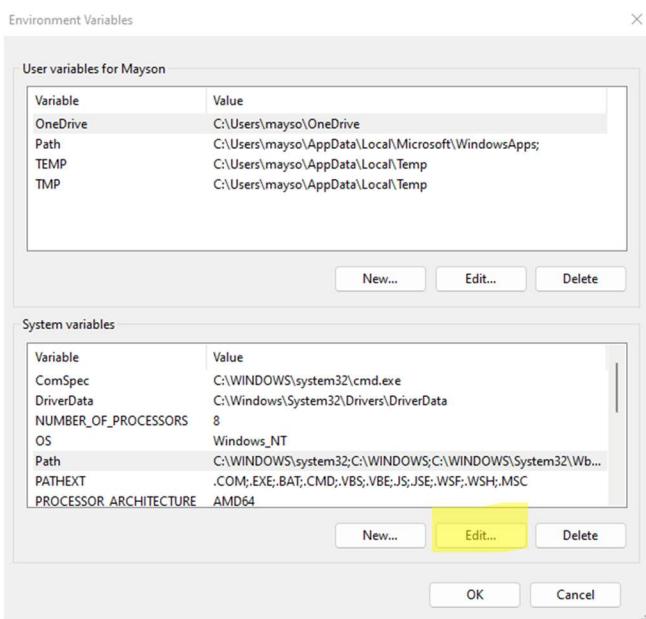
```
Last login: Sun Nov  6 09:44:44 on ttys000
mayson@Maysons-Air ~ % /usr/local/mysql/bin/mysql -u root -p
Enter password: ?
```

Windows:

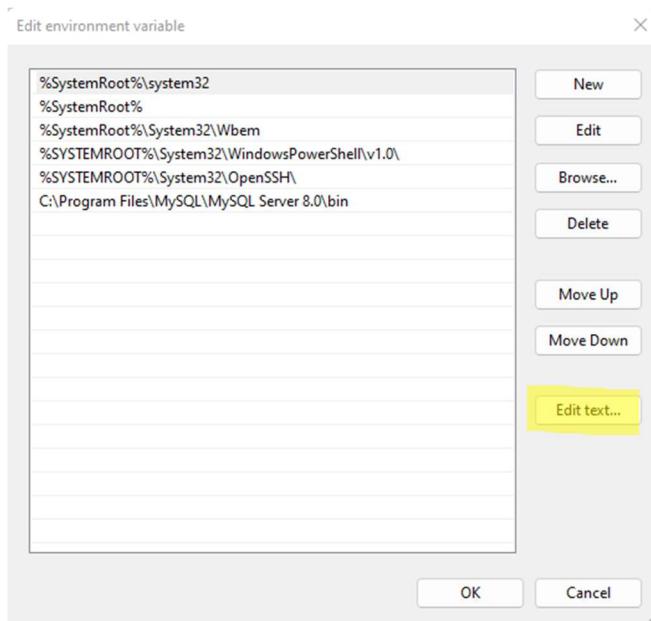
Search for and select “edit the system environment variables” in desktop search bar. From here, select “Environment Variables..”



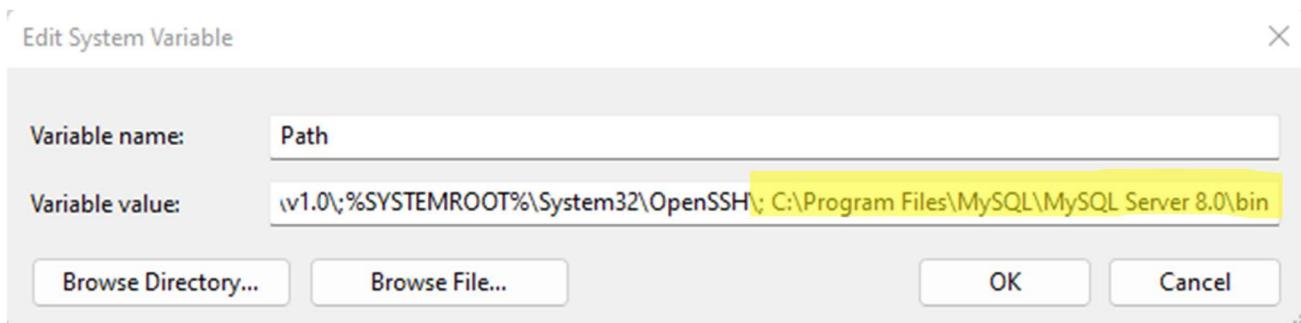
Select on “Path” under system variables, then select “Edit...”



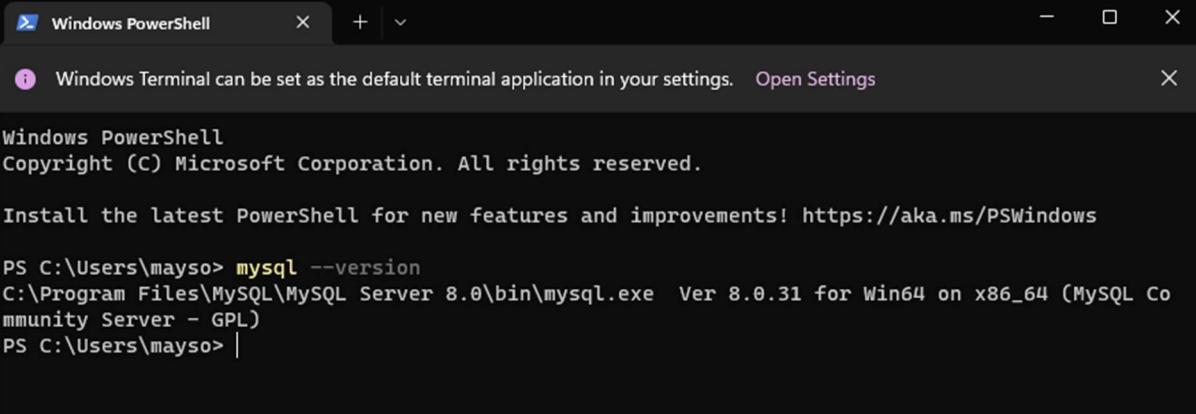
Select “Edit text...”



In the “variable value:” section, add “;” at the end of the existing statement, then paste the file address for the MySQL Server bin found on your local machine. (You can search for this in your File Explorer). Click “OK”



From here, MySQL server should be up and running. To test the connection, open up the terminal and type “mysql – version”. If it provides the server version, then you’re all set to go.

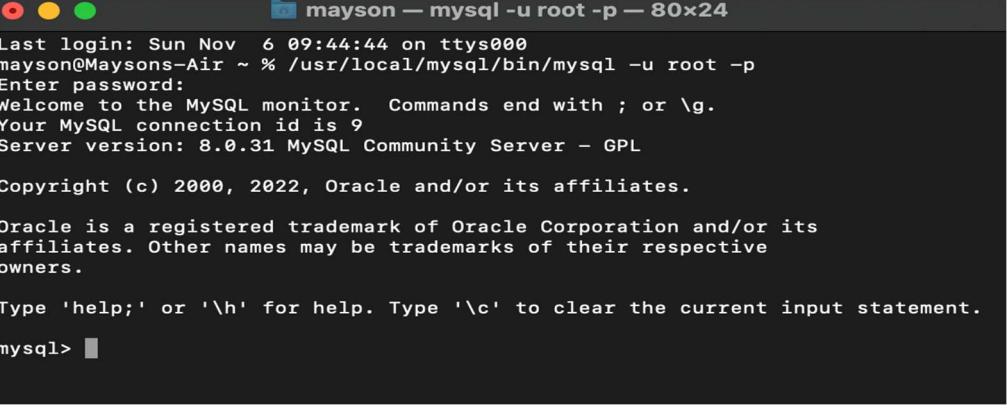


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\mayso> mysql --version
C:\Program Files\MySQL\MySQL Server 8.0\bin\mysql.exe Ver 8.0.31 for Win64 on x86_64 (MySQL Co
mmunity Server - GPL)
PS C:\Users\mayso> |
```

**Step 13:** You should now be connected to the MySQL server on your machine.



```
mayson - mysql -u root -p - 80x24
Last login: Sun Nov  6 09:44:44 on ttys000
mayson@Maysons-Air ~ % /usr/local/mysql/bin/mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.31 MySQL Community Server - GPL

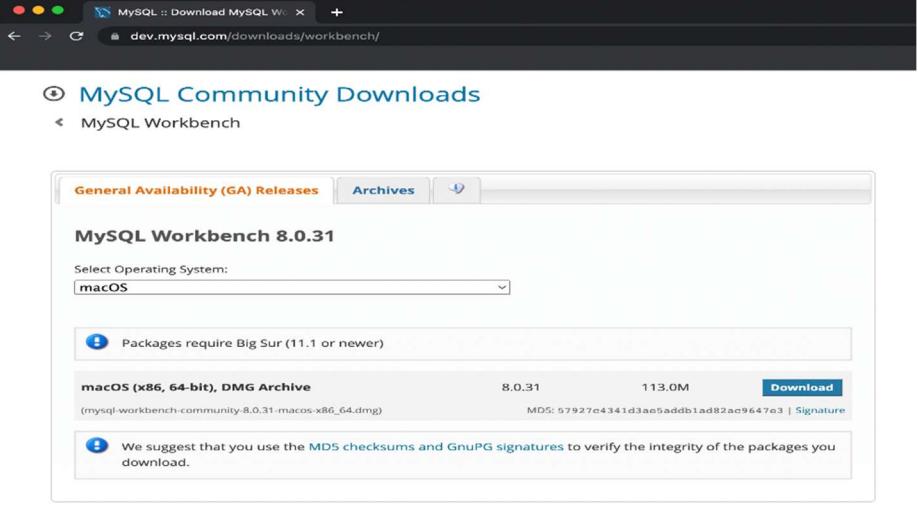
Copyright (c) 2000, 2022, Oracle and/or its affiliates.

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> |
```

**Step 14:** Download “SQL Workbench” on computer by going to <https://dev.mysql.com/downloads/workbench/>. Ensure to choose your appropriate OS.

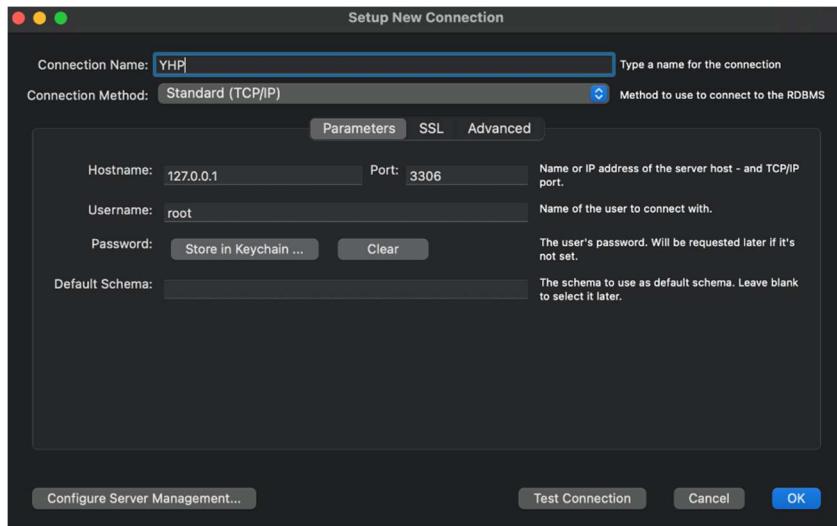


The screenshot shows a web browser window with the URL [dev.mysql.com/downloads/workbench/](https://dev.mysql.com/downloads/workbench/). The page title is "MySQL Community Downloads". Below the title, there is a breadcrumb navigation: "MySQL Workbench". The main content area is titled "MySQL Workbench 8.0.31". It has tabs for "General Availability (GA) Releases" (which is selected), "Archives", and a download icon. A dropdown menu for "Select Operating System" is open, showing "macOS" as the current choice. Below the dropdown, a note says "Packages require Big Sur (11.1 or newer)". There is a download link for "macOS (x86, 64-bit), DMG Archive" with the file name "(mysql-workbench-community-8.0.31-macos-x86\_64.dmg)". The file size is 113.0M and the version is 8.0.31. A "Download" button is present. At the bottom, there is a note about verifying integrity using MD5 checksums and GnuPG signatures.

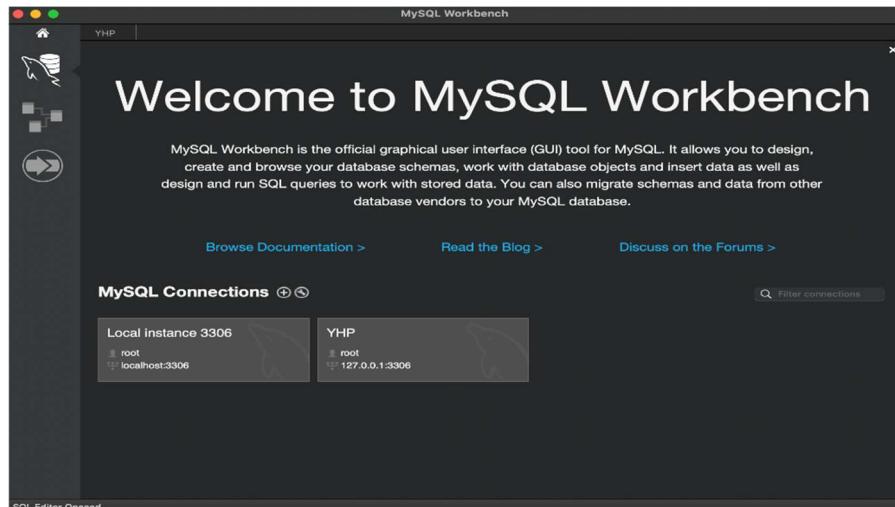
**Step 15:** Once downloaded, open the MySQL Workbench application. Select “+” symbol next to “MySQL Connections”



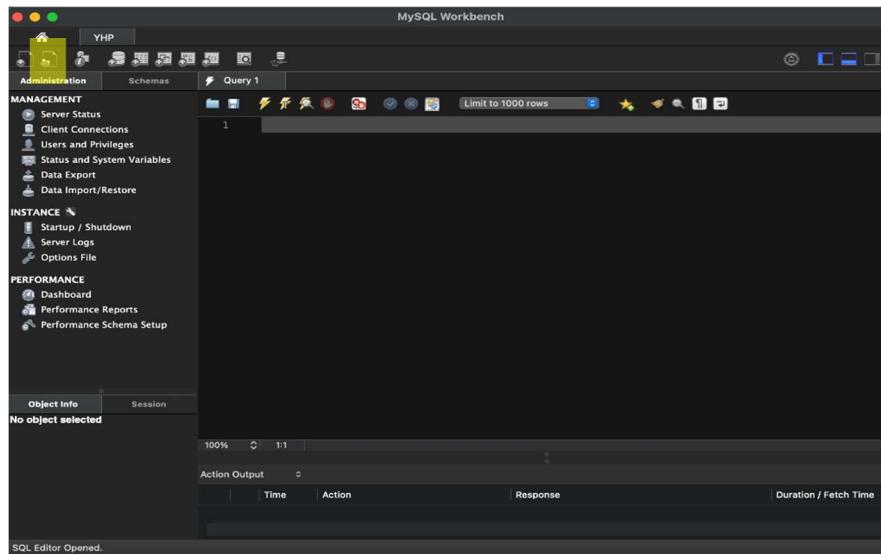
**Step 16:** Set up connection with database.



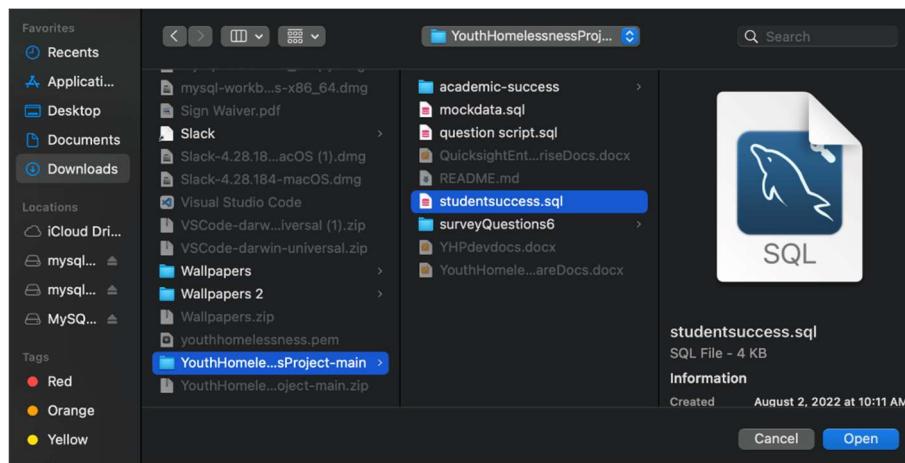
**Step 17:** You should now see the database under “MySQL Connections”.



**Step 18:** Open newly created database. Click the “open file” icon.



**Step 19:** Select SQL files from the YouthHomelessnessProject-main folder.



**Step 20:** Open “studentsuccess” sql script

```
MySQL Workbench
```

Administration Schemas Query 1 studentsuccess

MANAGEMENT

- Server Status
- Client Connections
- Users and Privileges
- Status and System Variables
- Data Export
- Data Import/Restore

INSTANCE

- Startup / Shutdown
- Server Logs
- Options File

PERFORMANCE

- Dashboard
- Performance Reports
- Performance Schema Setup

Action Output

Time Action Response Duration / Fetch Time

No object selected

Object Info Session

100% 1:1

Added new script editor

```

1 /* Welcome to the City of Orlando Youth Homelessness Project Team!
2 To get started, please execute only step one. After the database has
3 been created, launch the application in your preferred IDE, and come back.
4 (Be sure to update lines 3 & 4 in the resources/application.properties file
5 with the user/password of THIS MySQL connection).
6
7
8 When the app launches, Spring Data JPA and Hibernate will create tables
9 in the database that correspond to each entity in the models package.
10
11 Now, uncomment steps 2 and 3 to populate some default data
12 into these newly created tables.
13
14 Finally, open a browser and navigate to localhost:8080/ and log in.
15 Although passwords are encrypted below, User/Password combinations are:
16 student/student, employee/employee, admin/admin
17
18 Happy Coding!
19 */
20

```

**Step 21:** find “CREATE DATABASE IF NOT EXISTS studentsuccess;” line, highlight it with mouse, and execute (lightning bolt)

The screenshot shows the MySQL Workbench interface. In the left sidebar, under the 'Schemas' tab, there is a table with one row labeled 'studentsuccess'. The main pane displays a SQL query window with the following content:

```

MySQL Workbench

Administration Schemas Query 1 studentsuccess
Limit to 1000 rows

15 Although passwords are encrypted below, User/Password combinations are:
16 student/student, employee/employee, admin/admin
17
18 Happy Coding!
19 */
20
21 -- 1) Create Database
22 • DROP DATABASE IF EXISTS studentsuccess;
23 • CREATE DATABASE IF NOT EXISTS studentsuccess;
24
25
26 -- 1.5) Start app and let JPA/Hibernate create tables
27 --      (or write insert statements for each table below)
28
29
30 • USE studentsuccess;
31
32 -- 2) Populate users
33 -- Admin, Employee, Survey Admin, Student
34 -- INSERT INTO tbl_admins (id, first_name, last_name, password, user_role, username)

```

The line 'CREATE DATABASE IF NOT EXISTS studentsuccess;' is highlighted with a blue selection bar. Below the query window, the 'Action Output' pane shows a single log entry:

Action	Time	Response	Duration / Fetch Time
CREATE DATABASE IF NOT EXISTS stud...	10:13:14	1 row(s) affected	0.0031 sec

At the bottom, a status bar indicates 'Query Completed'.

**Step 22:** Open YouthHomelessProject in IDE and under the academic-success folder > src > resources, select “application.properties”. From here, ensure that the MySQL data source for local development reflects your machines database information and creditnals.

The screenshot shows the IntelliJ IDEA interface with the project 'YouthHomelessProject-main' open. The file tree on the left shows the directory structure:

- Project
  - academic-success
  - src
    - main
    - resources
  - test

The file 'application.properties' is selected in the resources directory. The code editor shows the following content:

```

# MySQL data source (local development)
spring.datasource.url=jdbc:mysql://localhost:3306/studentsuccess
spring.datasource.username=admin
spring.datasource.password=$$Newnew99

server.port=8080

# AWS Login info
# spring.datasource.url=jdbc:mysql://database-1.cmrgcn1rk9c.us-east-1.rds.amazonaws.com:
# spring.datasource.username=admin
# spring.datasource.password=reaper01

# AWS server port
# server.port=5000

spring.jpa.hibernate.ddl-auto=update
spring.jpa.hibernate.cache=false

```

The line 'spring.datasource.url=jdbc:mysql://localhost:3306/studentsuccess' is highlighted with a yellow selection bar.

## Importing SQL Scripts into MYSQL Workbench

**Objective:** This document is meant to give you the step-by-step procedure on how to import an SQL script into MSQL Workbench.

**Prerequisites:** This Documentation assumes that you already have MSQL set up with a MYSQL connection:

**Note:** This guide makes references to the youth homelessness project and its web application that may already be running on your local instance; however, you do not need a local instance of an application running to import SQL scripts into MYSQL workbench

<b>Step</b>	<b>Procedure</b>	<b>Expected Result</b>	<b>Comment</b>
1	Click the File drop-down menu on the top left of the page.	File drop menu display	
2	Click open script.	It will open your folders directory	
3	Search for the YouthHomelessnessProjectSoftwareDocs or desired folder, then click on the project folder.	The project folder will display all files and folders.	
4	Click on the studentsuccess file or desired file	The file will be imported into MSQl workbench	
5	To create the database, go to step one in the script and highlight the entire statement in step one, then click on the lightning bolt icon under the script tab to execute the highlighted statement. To run all the statements, click the lightning bolt. Note: if you started the app in IntelliJ, then JPA hibernate will have automatically created the columns. Make sure you read the Comments in the MYQL script. Beware, if IntelliJ already created the tables and columns, you will get errors since those tables and columns will already exist in the database.	MSQL will Create the Database. If you run the entire script all at once, then MYSQL will Also create all necessary columns for each table in the database.	
6	Click the File drop-down menu on the top left of the page	File drop menu display	
7	Click open script	It will open your folders directory	
8	Search for the YouthHomelessnessProjectSoftwareDocs, then click on the project folder.	The project folder will display all files and folders.	
9	Click on the script file	The file will be imported into MSQl workbench	
10	Beware, if you run the entire script simultaneously, you will get errors. Please run the statement in increments by highlighting the portion you want to run, then click lightning bolt under the script tab. If you get an error message, read the error code and message. Most likely, you can skip the code you are trying to run. Some of the errors are caused by unsupported comment tags or because a table has been created already. If the table is already created,	Statements are executed to create new questions and constraints, such as foreign keys, relationships, and many more.	

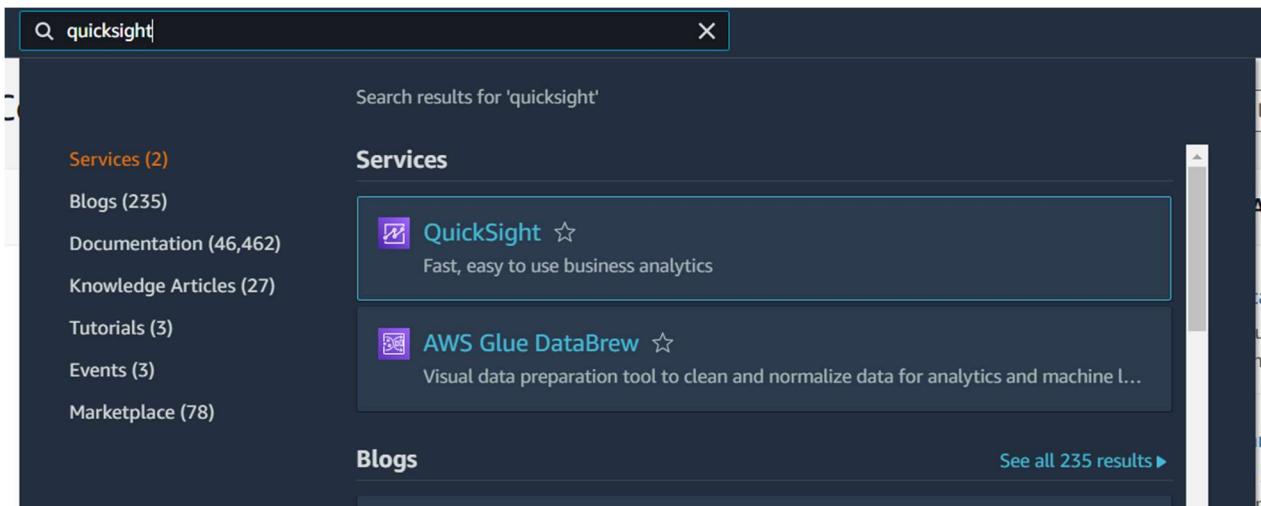
	skip that statement, and move to the next statement or statements.		
--	--	--	--

## MySQL to AWS Quicksight

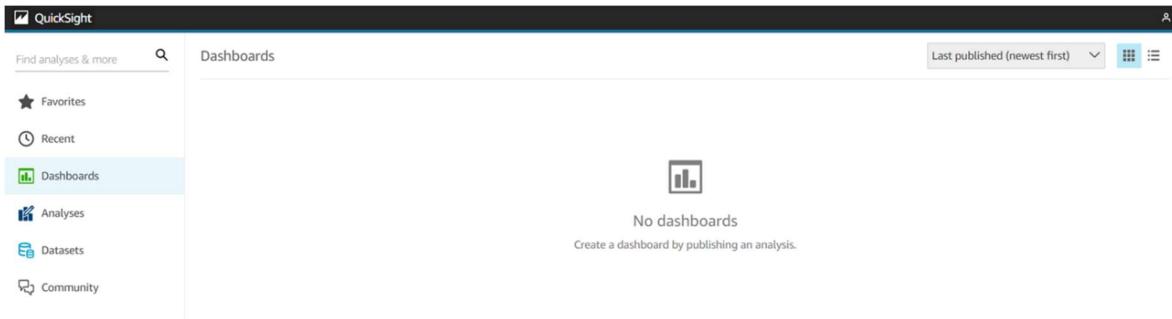
### Step 1: Log in to your AWS Account



### Step 2: Once signed in, search for “Quicksight”



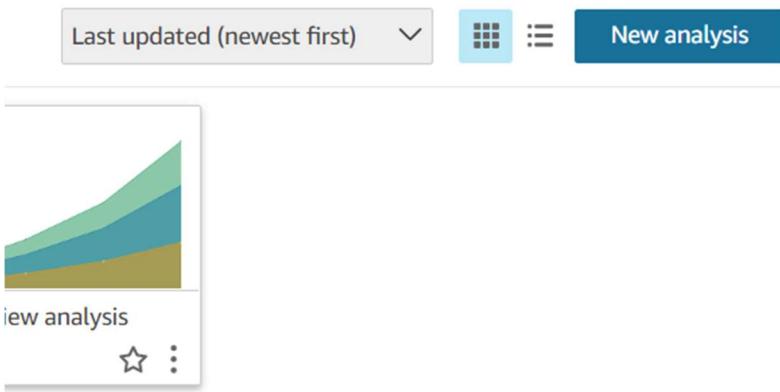
### Step 3: It will prompt you to create an account for Quicksight. Once, the account is created you will be able to see your dashboard.



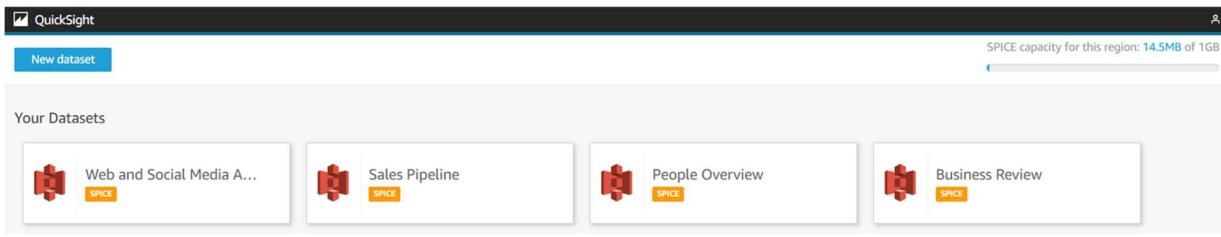
### Step 4: Open “Analyses”



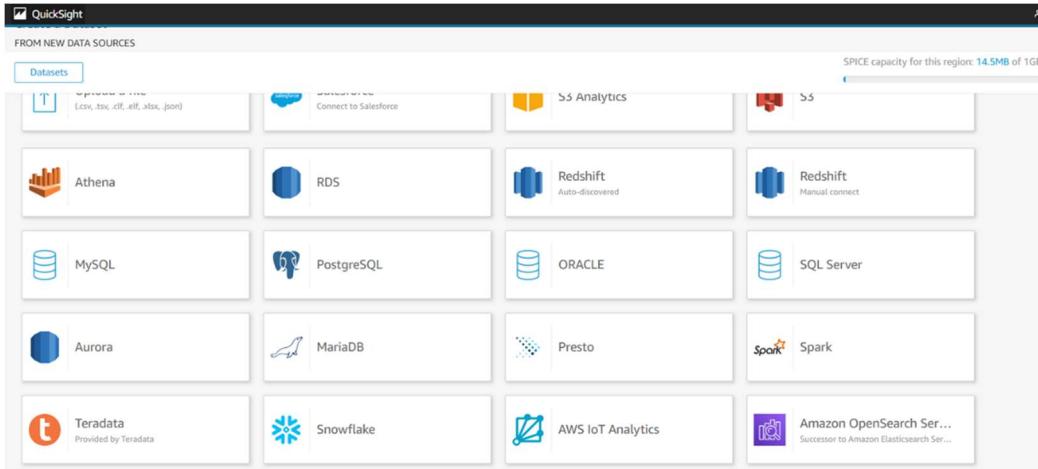
### Step 5: Click on “New analysis”



### Step 6: Click on “New dataset”



### Step 7: Choose “MySQL” from the options.



**Step 8:** A window should pop-up prompting you to enter details such as server, port, username, password, etc. Enter your MySQL credentials and validate credentials. (*Note: This is one way to connect MySQL to Quicksight. However, my team decided to upload the files to Quicksight as that is more convenient*).

(The following next steps are on how to upload the MySQL files to Quicksight)

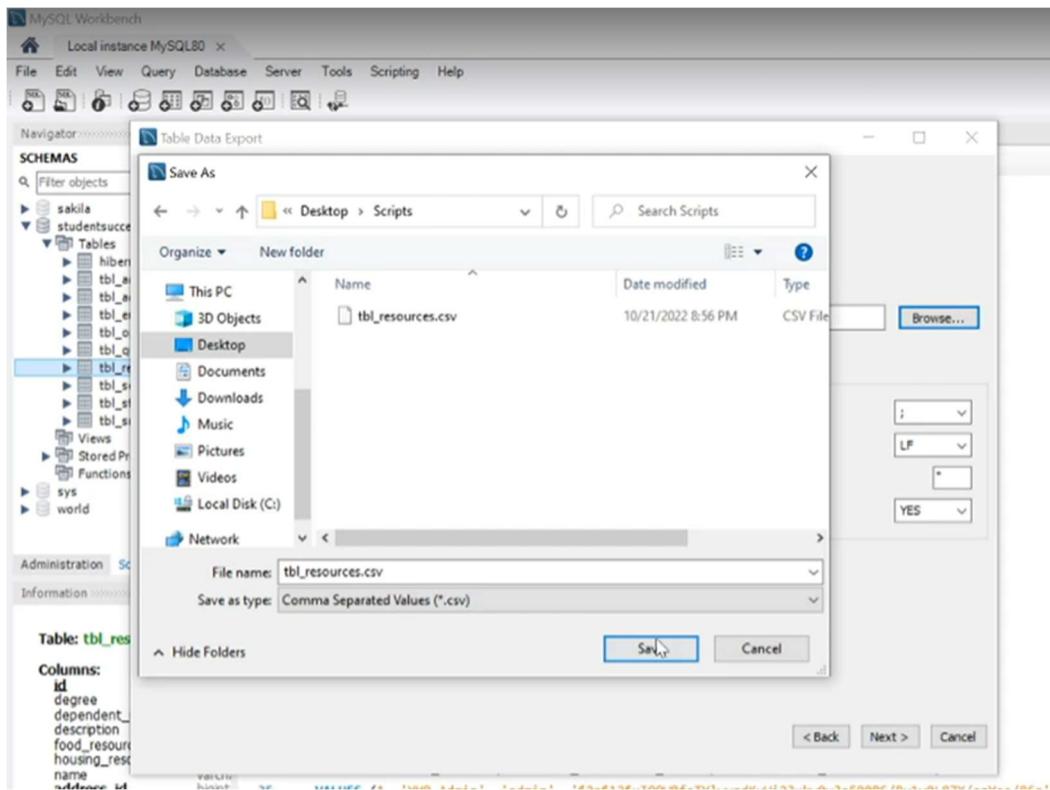
**Step 1:** Log in to MySQL Workbench



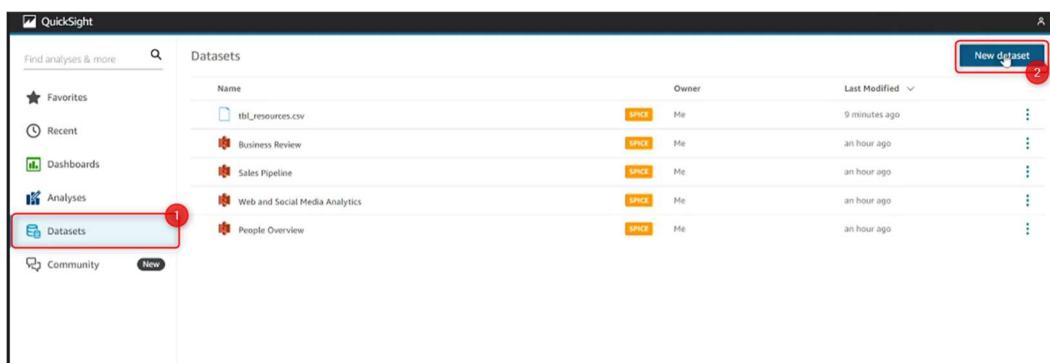
## Step 2: Navigate to the table and export data

This screenshot shows the MySQL Workbench interface with the "studentsuccess" database selected. The Navigator pane on the left lists tables under the "studentsuccess" schema. A context menu is open over the "tbl\_res" table, with several options highlighted with red numbers: 1. "studentsuccess", 2. "tbl\_res", 3. "tbl\_res", and 4. "Table Data Export Wizard". The "Table Data Export Wizard" option is the target of the step. The main pane displays SQL code related to the "studentsuccess" database, including table creation and data insertion statements.

## Step 3: Use export wizard to save file as a .CSV file



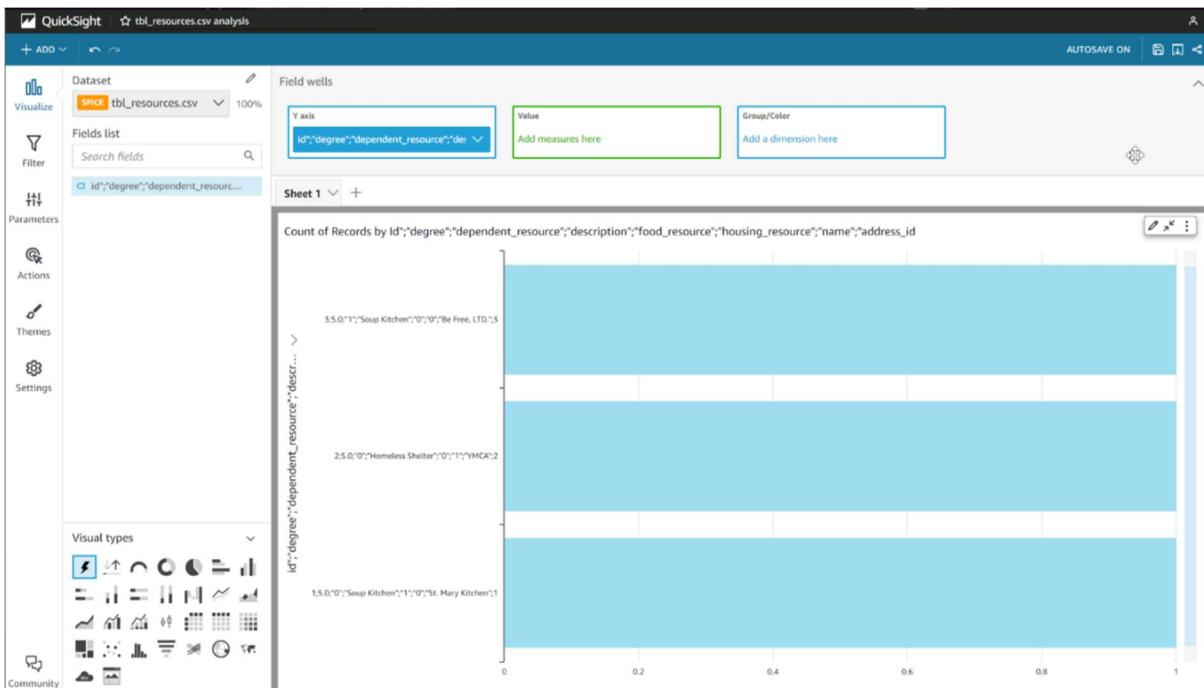
#### Step 4: Navigate to Quicksight Datasets and create a New dataset



#### Step 5: Upload the MySQL .CSV file

The screenshot shows the 'Create a Dataset' page in QuickSight. At the top left, there's a 'Datasets' tab and a note about SPICE capacity. Below it, a section titled 'FROM NEW DATA SOURCES' contains a grid of 16 icons representing different data sources. The first icon, 'Upload a file (.csv, .txt, .clif, .aif, ...)', is highlighted with a red border.

## Step 6: View your datasets



## AWS BEANSTALK Deployment

(Reference video: [https://www.youtube.com/watch?v=EtDSJRdpJM4&ab\\_channel=vaadinofficial](https://www.youtube.com/watch?v=EtDSJRdpJM4&ab_channel=vaadinofficial))

### Step 1

Create and define other properties files (application-prod.properties) in the main resources folder. Prod refers to the profile we use when we start the application, and we will also configure it in AWS so that the properties get recognized by AWS Elastic Beanstalk. The server port for AWS should be mapped to 5000 so it gets served. We will also set up an MYSQL production database. Also, we will set up environment variables in AWS.

The screenshot shows the IntelliJ IDEA interface with the project navigation bar at the top. Below it, the project structure is displayed, showing a package named 'com.youthhomelessnessproject.aca' containing several subfolders like config, controllers, dto, models, repositories, services, and utils. Inside this package is an 'AcademicSuccessApplication' class. Underneath the package, there is a 'resources' folder containing static and templates subfolders, along with two properties files: 'application.properties' and 'application-prod.properties'. The 'application-prod.properties' file is open in the editor, showing the following configuration:

```

server.port=5000
spring.datasource.url=jdbc:mysql://${RDS_HOSTNAME}:${RDS_PORT}/${RDS_DB_NAME}
spring.datasource.username=${RDS_USERNAME}
spring.datasource.password=${RDS_PASSWORD}
spring.jpa.hibernate.ddl-auto=create

```

### Step 2

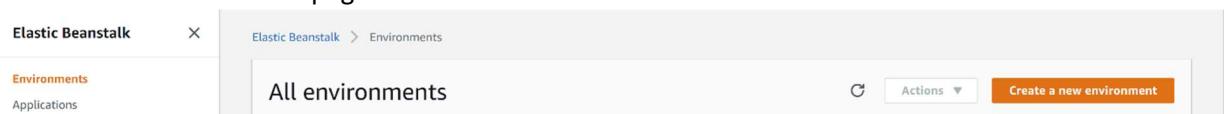
Build a production Build. It's optional to use the keyword -DskipTests to build faster. If you are using IntelliJ Idea, click the terminal tab at the bottom.

**Run:** mvn clean package -DskipTests

The screenshot shows the IntelliJ IDEA interface with the terminal tab active at the bottom. The terminal window displays the command 'mvn clean package -DskipTests' being run in a Windows PowerShell session. The output shows the command being entered and the progress of the Maven build process.

### Step 3

- Go to AWS beanstalk to get started <https://aws.amazon.com/elasticbeanstalk/>. If you already have an AWS account, you can also go to the services section to find AWS Elastic Beanstalk. Once signed in, “Create new environment” on the main page.



B. After you choose to create an application, choose web server environment tier

C. When you choose an application name, the environment will auto-populate

**Elastic Beanstalk**

**Create a web server environment**

Launch an environment with a sample application or your own code. By creating an environment, you allow Amazon Elastic Beanstalk to manage Amazon Web Services resources and permissions on your behalf. [Learn more](#)

**Application information**

Application name  
Your\_application\_name

Up to 100 Unicode characters, not including forward slash (/).

► Application tags (optional)

**Environment information**

Choose the name, subdomain, and description for your environment. These cannot be changed later.

Environment name  
Yourapplicationname-env

Domain  
Leave blank for autogenerated value .us-east-1.elasticbeanstalk.

- D. For the platform, we choose Java and Corretto 8 branch because the app used java 8 in development.

**Platform**

Managed platform  
Platforms published and maintained by Amazon Elastic Beanstalk. [Learn more](#)

Custom platform  
Platforms created and owned by you.

Platform  
Java

Platform branch  
Corretto 8 running on 64bit Amazon Linux 2

Platform version  
3.4.1 (Recommended)

- E. For now, choose the sample code instead of uploading your code. Then, click "Configure more options."

**Application code**

- Sample application  
Get started right away with sample code.
- Existing version  
Application versions that you have uploaded for `Your_application_name`.
- Upload your code  
Upload a source bundle from your computer or copy one from Amazon S3.

-- Choose a version --

Cancel    Configure more options    **Create environment**

- F. Leave presets on single instance (Free tier eligible).

#### Configure Yourapplicationname-env

**Presets**

Start from a preset that matches your use case or choose *Custom configuration* to unset recommended values and use the service's default values.

Configuration presets

- Single instance (*Free Tier eligible*)
- Single instance (using Spot instance)
- High availability
- High availability (using Spot and On-Demand instances)
- Custom configuration

- G. Click edit in the Software section to add a dependent variable to recognize the application-prod.properties file of your application.

Software		
Amazon X-Ray: disabled	Log streaming: disabled (default)	Environment properties: 3
Rotate logs: disabled (default)		GRADLE_HOME, M2, M2_HOME

Edit

- H. Under the Environment properties section, add the property name `JAVA_HOME` and the value `/usr/lib/jvm/java`. Next, add the property name `SPRING_PROFILES_ACTIVE` and the value `prod`. Click save.

**Environment properties**

The following properties are passed in the application as environment properties. [Learn more](#)

Name	Value
GRADLE_HOME	/usr/local/gradle
M2	/usr/local/apache-maven/bin
M2_HOME	/usr/local/apache-maven
JAVA_HOME	/usr/lib/jvm/java
SPRING_PROFILES_ACTIVE	prod

[«](#) [»](#)

[Cancel](#) [Save](#)

- I. Back in the “Configure more options” page, click edit in the database section.

**Database**

Engine:	Instance class:	Multi-AZ:
--	--	--
	Storage (GB):	
	--	

[Edit](#)

- J. (In final production, you may want to click learn more where it says, “Add an RDS SQL database integrated with your environment. [Learn more](#) ”)  
For now, we will create a MySQL instance on AWS beanstalk. So in the Database settings, add a username and password, which are part of your environment variables in the application-prod.properties file in your application. For the demo application, choose “Delete” for the database deletion policy. Click save.

**Database settings**

Choose an engine and instance type for your environment's database.

**Engine**  
mysql

**Engine version**  
8.0.28

**Instance class**  
db.t2.micro

**Storage**  
Choose a number between 5 GB and 1024 GB.  
5

**Username**  
Your\_user\_name

**Password**  
\*\*\*\*\*

**Database deletion policy**  
This policy applies when you decouple a database or terminate the environment coupled to it.

- Create snapshot**  
Elastic Beanstalk saves a snapshot of the database and then deletes it. You can restore a database from a snapshot when you add a DB to an Elastic Beanstalk environment or when you create a standalone database. You might incur charges for storing database snapshots.
- Retain**  
The decoupled database will remain available and operational external to Elastic Beanstalk.
- Delete**  
Elastic Beanstalk terminates the database. The database will no longer be available.

**Cancel** **Save**

K. Back in the “Configure more options page”, click “Create environmental”

**Cancel** **Previous** **Create environment**

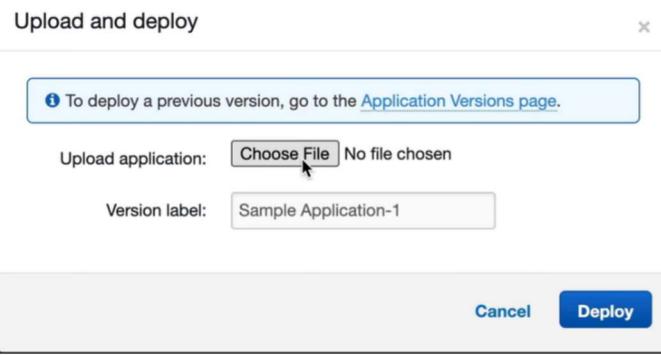
(It can take 10-15 minutes to create the environment)

L. Once the instance is running, click “Upload application.”

## Overview



- M. Click “Choose File.” Select the academic-success jar file in the target folder, then click “deploy.”



academic-success-1.0.0.jar

(It can take 10-15 minutes to deploy)

- N. The health status should be green. Click the URL to View the application online.

