Lewis University  
CPSC 50900: Database Systems  
Term Project

World population & migrants database

Vineeth Reddy Gaddam, vineethreddygaddam@lewisu.edu

Prem Kumar Baddam, Premkumarbaddam@lewisu.edu

Sushanth Reddy Teegala, sushanthreddyteega@lewisu.edu

<https://github.com/vgaddam12/Database-system>

Table of Contents

[Initial Proposal 2](#_Toc85814569)

[Data Sources 2](#_Toc85814570)

[Data Storage Alternatives 2](#_Toc85814571)

[Relational Database Design Process 3](#_Toc85814572)

[Relational Database Design 3](#_Toc85814573)

[Data Definition Language (DDL) Scripts 3](#_Toc85814574)

[Data Manipulation Language Scripts 4](#_Toc85814575)

[Indexes 5](#_Toc85814576)

[Views 5](#_Toc85814577)

[Triggers 5](#_Toc85814578)

[Transactions 5](#_Toc85814579)

[Database Security 6](#_Toc85814580)

[Locking and Concurrent Access 6](#_Toc85814581)

[Backing Up Your Database 6](#_Toc85814582)

[Python Programming 7](#_Toc85814583)

[PHP Programming 7](#_Toc85814584)

[Suggested Future Work 8](#_Toc85814585)

[Activity Log 8](#_Toc85814586)

# Initial Proposal

# We will be storing world’s Population, surface area and density. Also, how many people are migrants

We are interested in this data because increasing population is the world’s biggest problem considering the resources are scarce. Also, migrants count will be provided. It is important to keep track of population. Also, how migrants are affecting in population of countries

It is important to keep track of population. Also, how migrants are affecting in population of countries

The data will come from

<http://data.un.org/>

Governments will use this data to keep control of their population

An informative application that may help students or researchers use data in an effective manner.

**Title**

World population & migrant’s database

# Data Souces

Gathered data in csv files from UN website. Data files contain world population by regions and overall as well and they contain number of refugees in divided by sexes, people who seek asylum etc. We are planning to include number of migrants grouped by regions and population overall of those regions.

Id(int, primary key, Unique, [NOT NULL](https://www.tutorialspoint.com/sql/sql-not-null.htm))

Region(string(100))

Population(integer)

Refugees(integer)

# Data Storage Alternatives

**Hierarchical Model:**  
The hierarchical database model is most appropriate for use cases in which the main focus of information gathering is based on a concrete hierarchy, such as several individual employees reporting to a single department at a company. That forms a tree-like organization If there are multiple nodes appear at the top level, then these can be called as root segments.

Records are linked with the help of pointers.

One parent may have many child.

**Advantages:**  
they are conceptually very simple

Very efficient

Ensure data integrity

**Disadvantages:**

Needed to know how the data is stored in order to access it.

Rigid design- adding a new field requires an entire redesign.

No standardized data access language.

**Network Database Model**:

Its structure is similar to hierarchical except, any given node can have multiple parents.

**Advantages:**

In this model, you could create a network that shows how data is related to each other

It implements 1:1, 1:n and also many to many relations.

It has standard language to access data.

To organize records, it uses graphs

Records are linked with the help of linked list.

There is no insertion anomaly.

**Disadvantages**

Navigation requires knowledge to data organization.

Structural dependence

# Relational Database Design Process

**Functional dependencies**

1. In population,surface area:

RegionCode,Country/area,year---->value

1. In migrants and Refugees:

RegionCode, Country/area, year, Series---->value

Naming entity sets:

Population

Migrant

Relationship among entity sets:

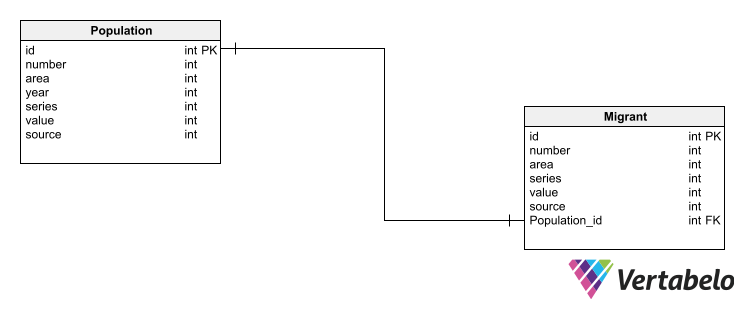
# 

# Relational Database Design

*4th normal form*

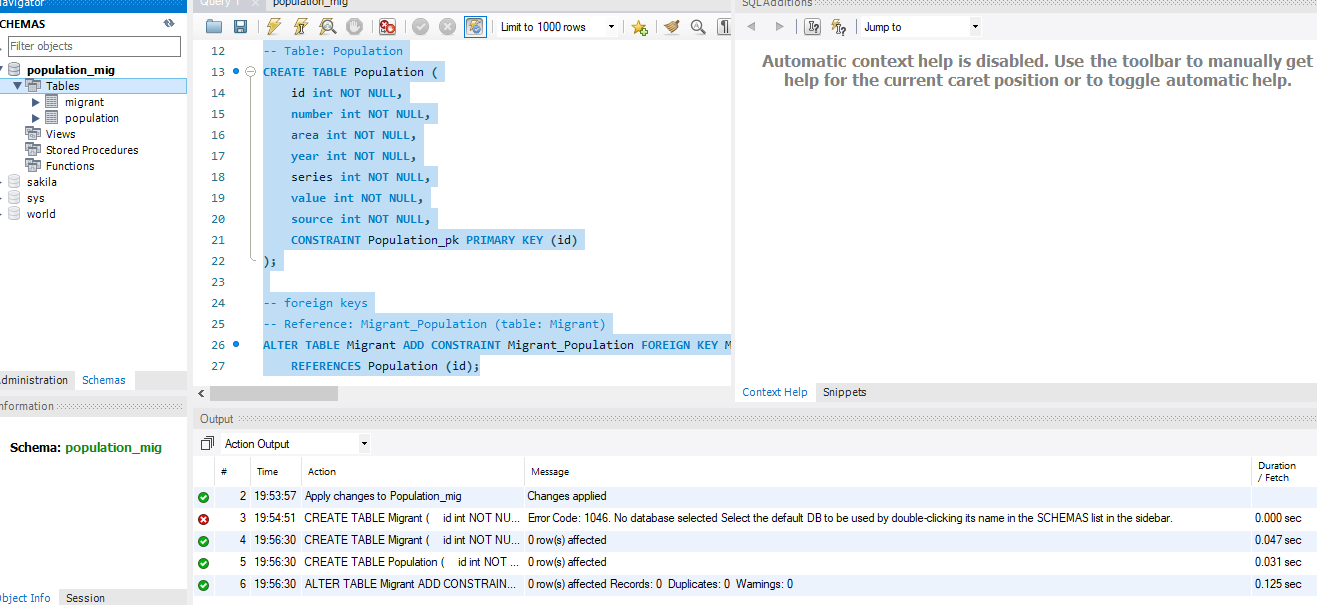
# 

# Physical database design



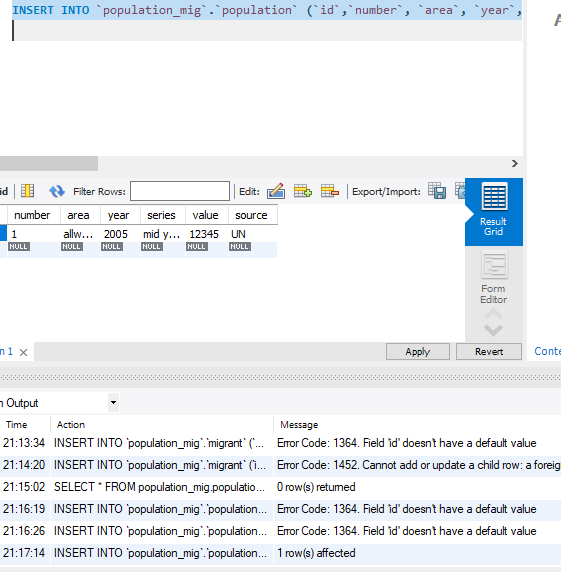
# Data Definition Language (DDL) Scripts

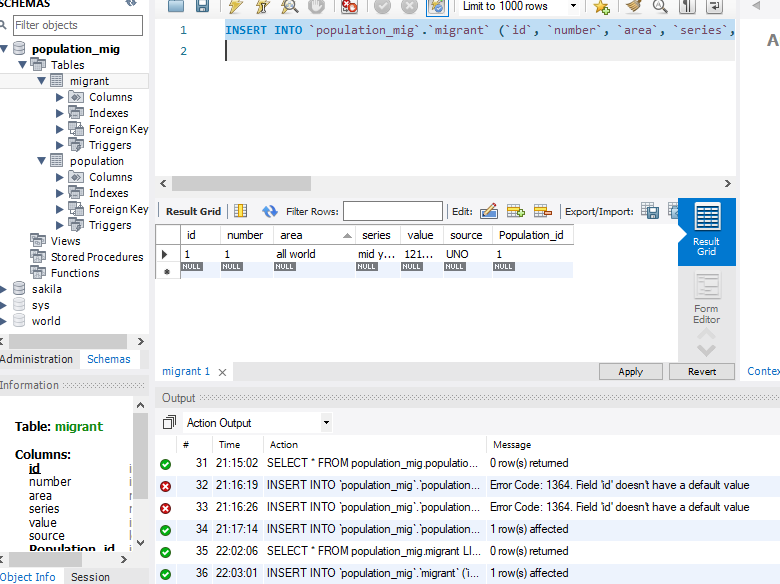
Vertabello has helped my generate sql commands which will help me create actual database tables with constraints applied.



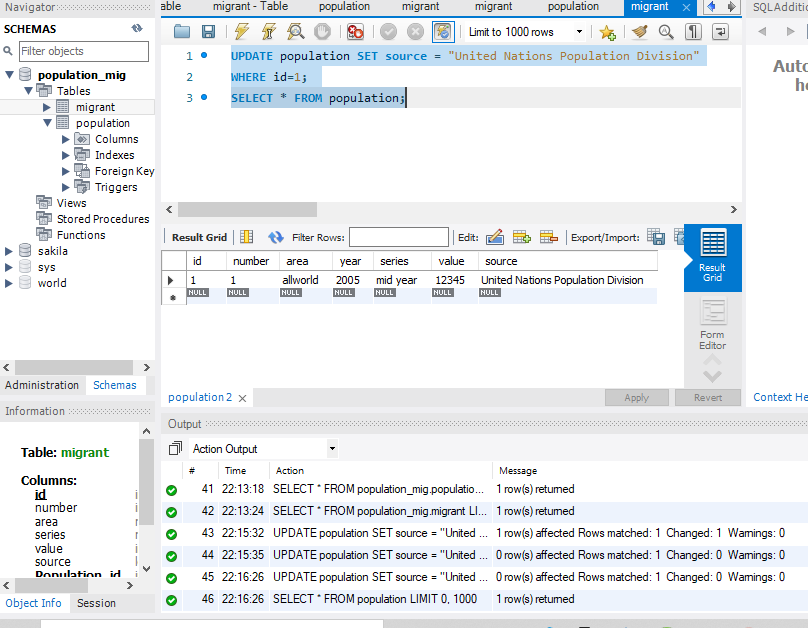
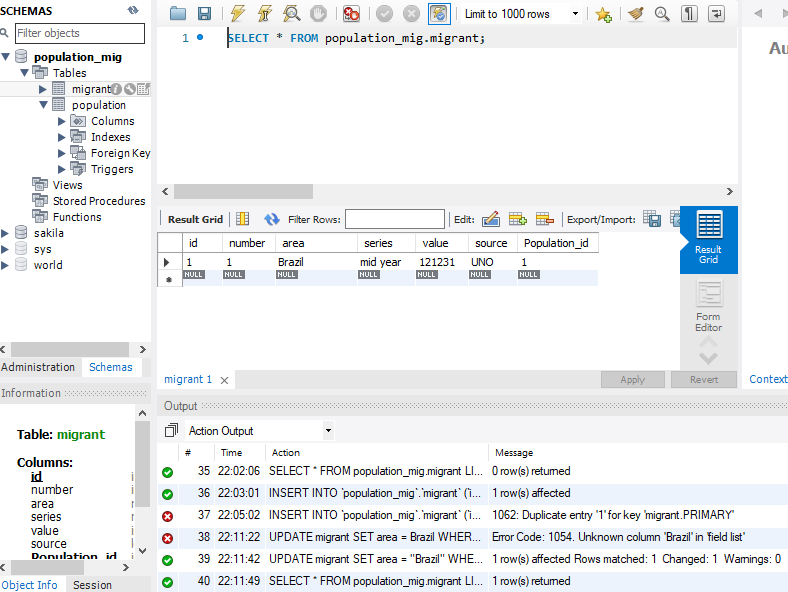
# Data Manipulation Language Scripts

*INSERT commands*

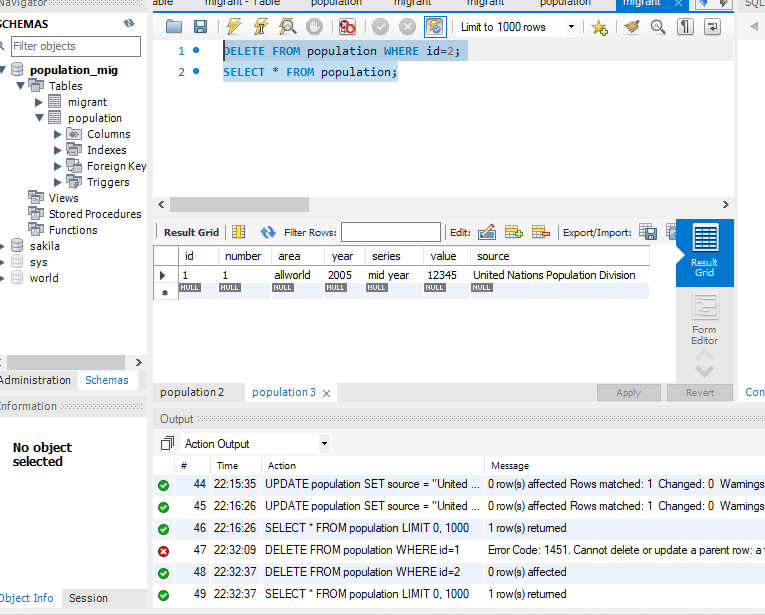




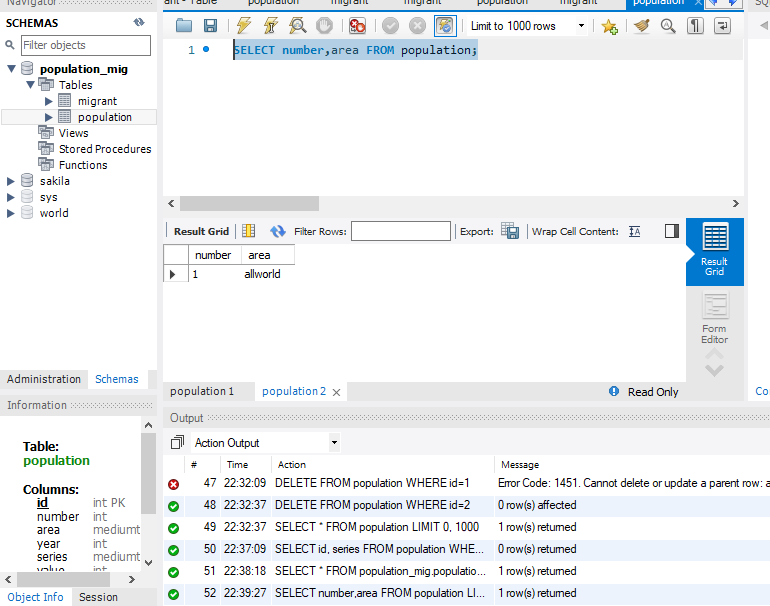
Updates commands



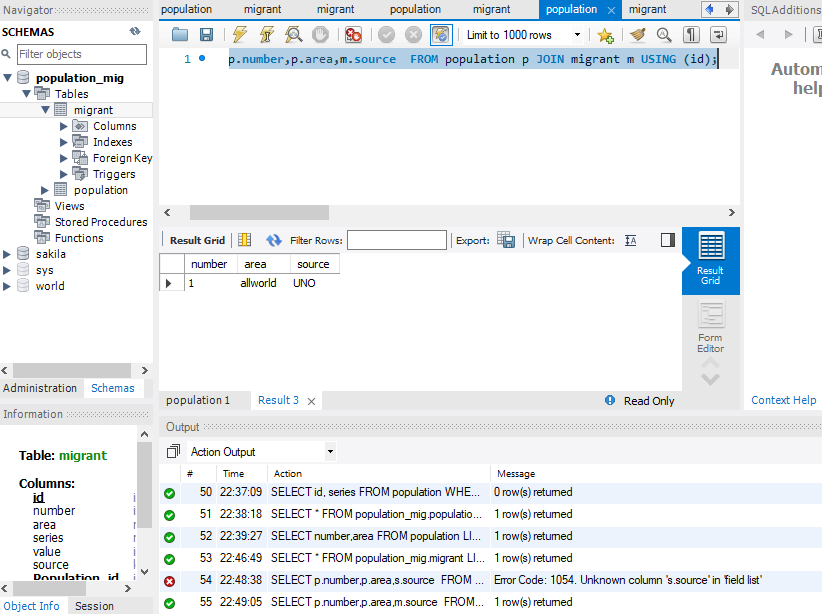
Delete command

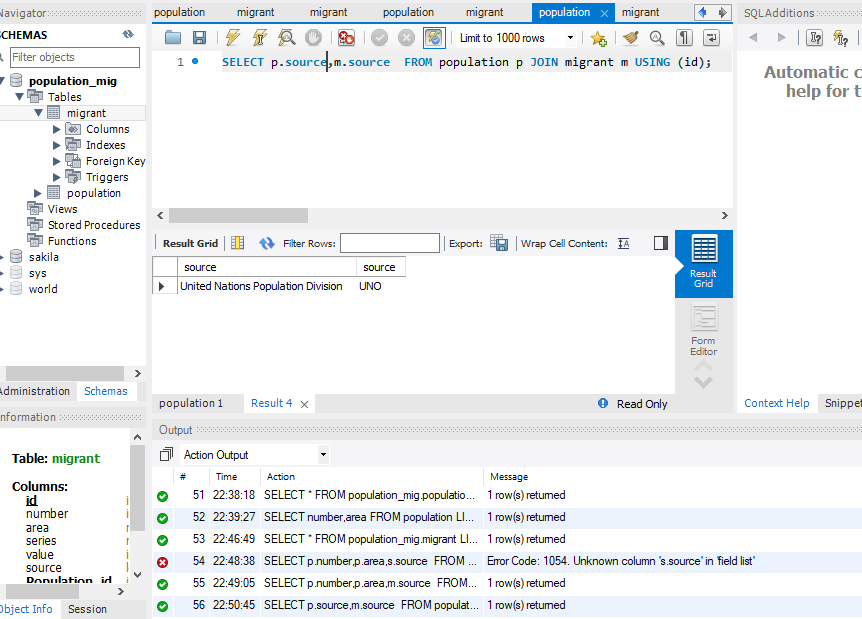


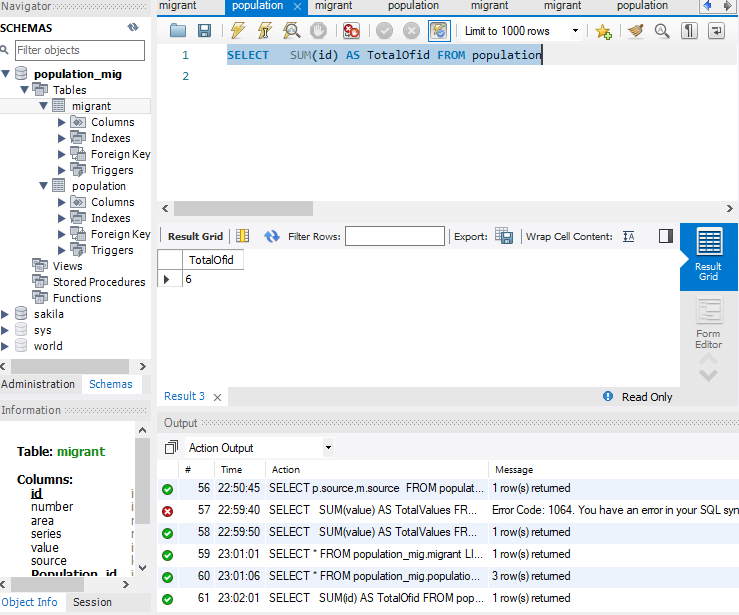
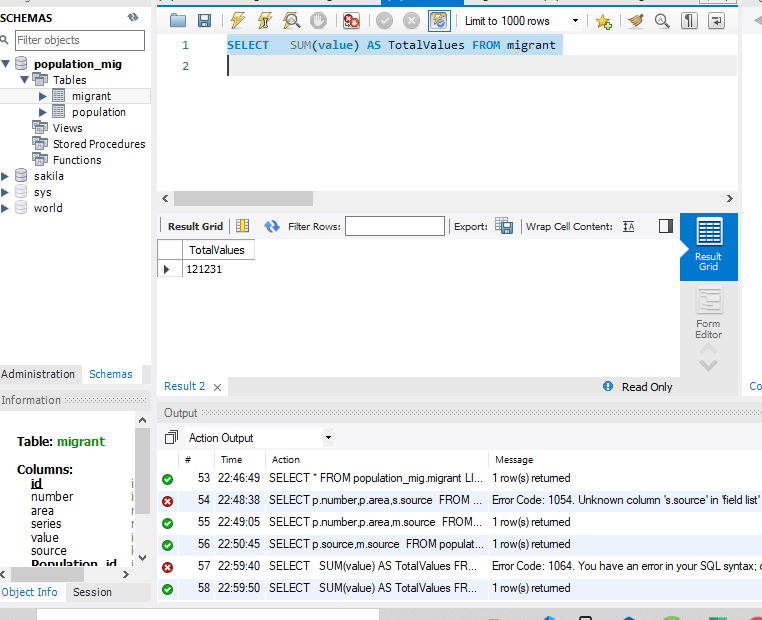
SELECT command

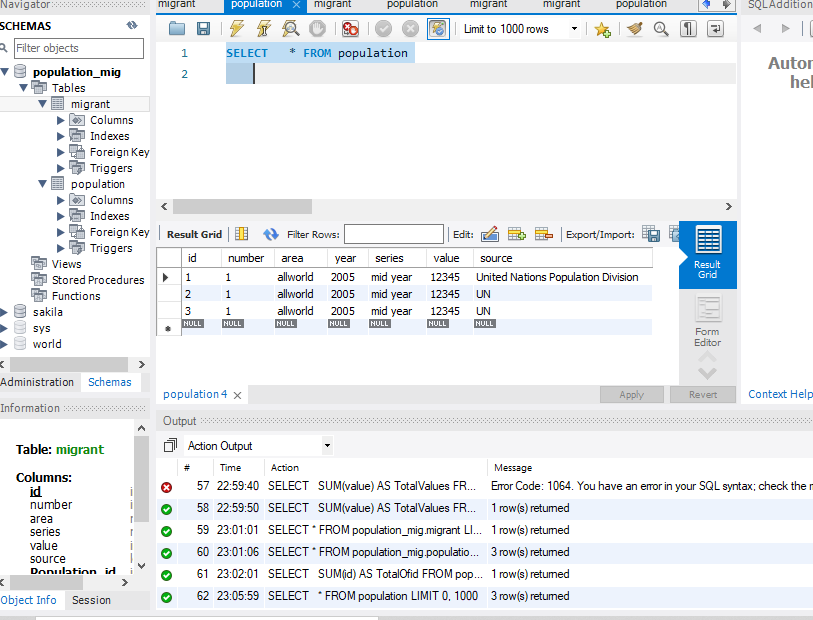


JOIN commands







Query of choice:  


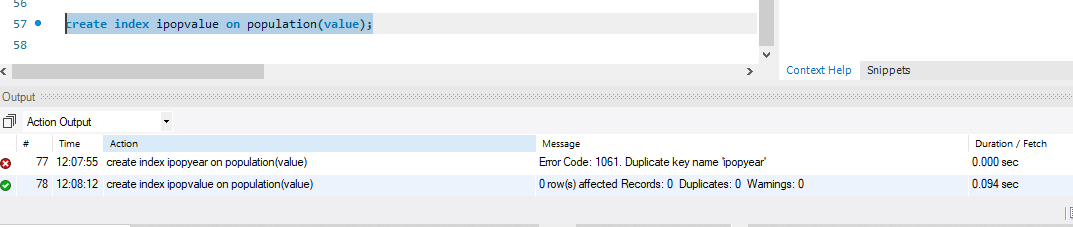
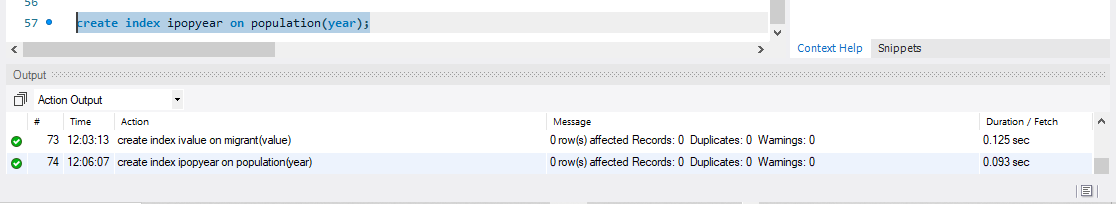
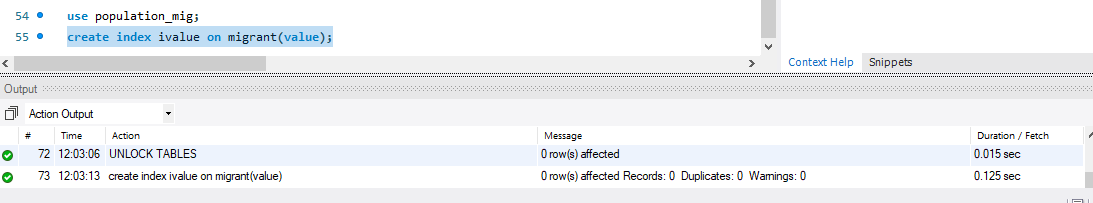
# Indexes

*Description: Improve the performance of your design by adding indexes to various tables. Show the SQL needed to add the indexes. Explain why you chose the ones you added. Explain how you would demonstrate the impact the indexes had on the performance of various queries.*

*Rubric: Your work will be graded as follows:*

* *6 points for clearly defining at least three indexes and explaining why you chose them.*
* *3 points for showing the sql needed to generate the indexes*
* *3 points for explaining how you would demonstrate the performance improvement afforded by the indexes.*

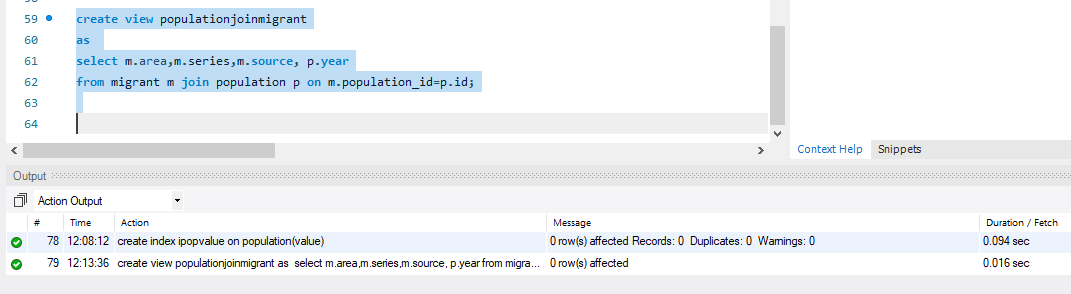
*Total points possible: 12*



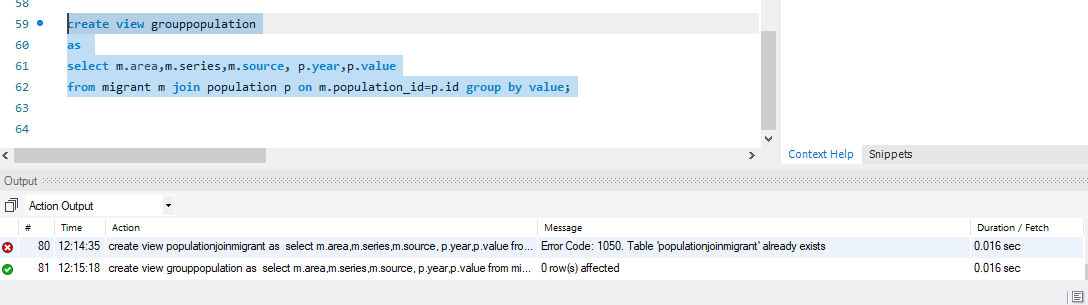
These indexes make querying data faster when it comes to filtering by columns other than primary keys

# Views

This view join two tables



This view is used when grouping by values



# Triggers



Trigger is used to perform an action based on another action.

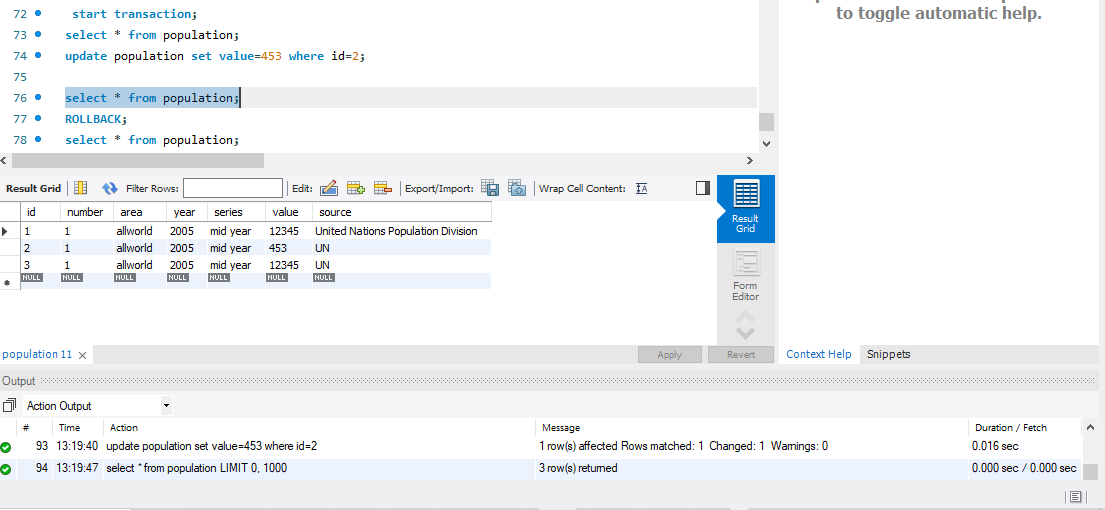
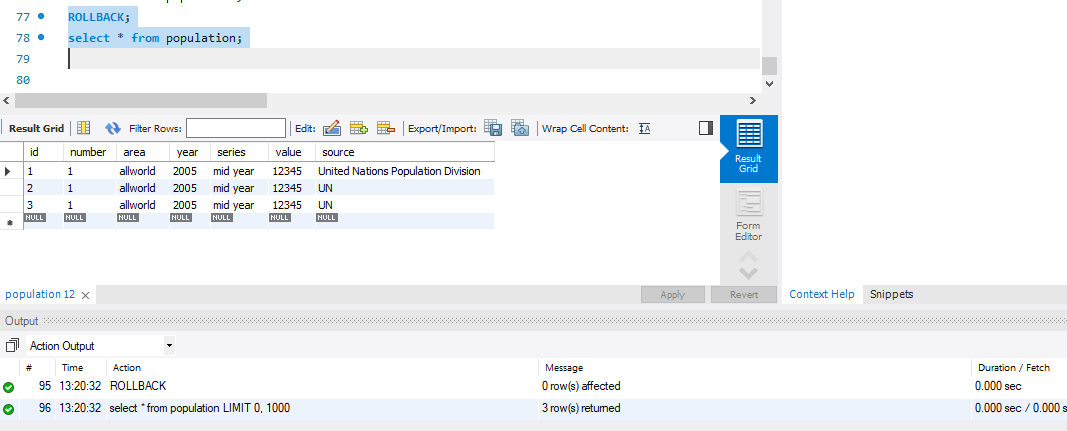
# Transactions

*Description: Demonstrate that you know how to define and use a transaction. Why are transactions important for ensuring ACID behavior?*

*Rubric: Your work will be graded as follows:*

* *3 points for clearly explaining the importance of transactions to ensuring ACID behavior*
* *3 points for including a screenshot and accompanying explanation of a MySQL transaction.*

*Total points possible: 6*

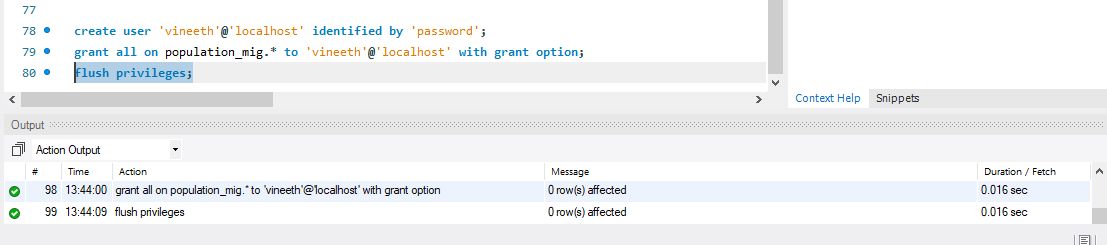


Transaction Group together multiple statements so that they are either all done, or none are done.

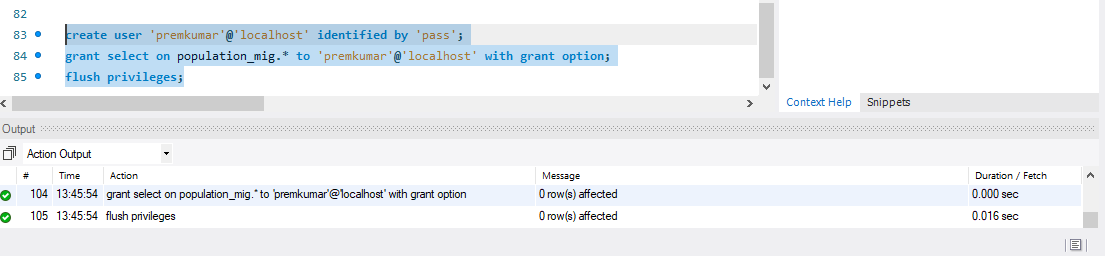
This makes sure ACID behavior which is consistency in data, atomicity, isolation, and durability

# Database Security

Vineeth is a user with all privileges

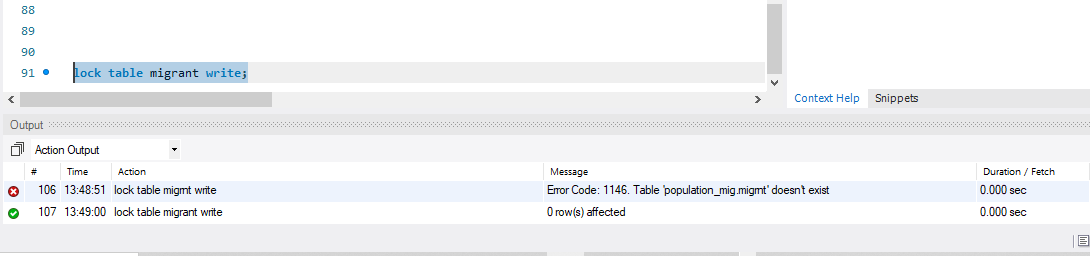
**

Premkumar has privileges of selecting data only

**

# Locking and Concurrent Access

We lock table to prevent other user to access/modify tables in database



# Backing Up Your Database

We need to run this command in bin folder of mysql installation to make a backup of database

mysqldump -h localhost -u vineeth -p password population\_mig>popMigrant.sql

# Python Programming

import mysql.connector

from mysql.connector import errorcode

//defines a function to grab data

def select(conn,query):

cursor = conn.cursor()

cursor.execute(query)

results = []

for row in cursor.fetchall():

results.append(row)

cursor.close()

return results

//defines a function to update delete and insert data

def execute(conn,query): # update, delete, and insert

cursor = conn.cursor()

cursor.execute(query)

conn.commit()

//defines a function that shows data to the screen

def show(rows):

for row in rows:

print(row)

//tries to establish connection with the database instance

try:

conn = mysql.connector.connect(

user="root",

password="Mrcomputer",

host="localhost",

database="population\_mig")

except mysql.connector.Error as err:

print("Cannot connect.")

exit()

//calls the select function defined above with connection string

rows = select(conn,"select \* from migrant where source = 'UNO'")

//calls function to show result on screen

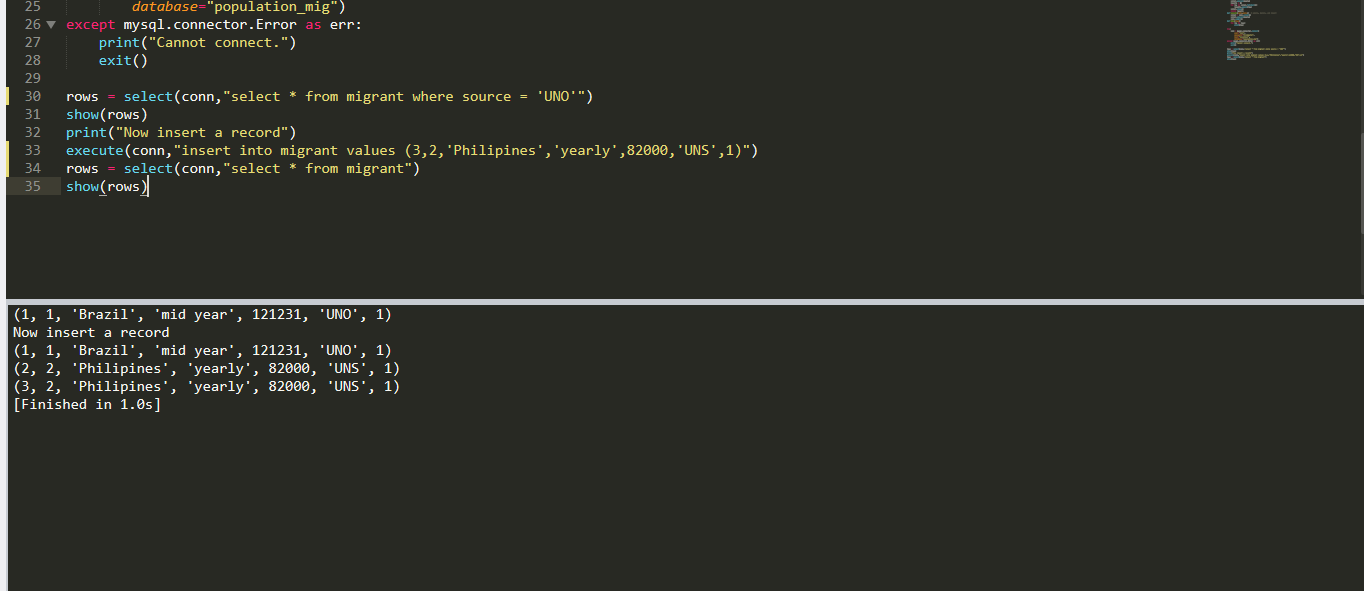
show(rows)

print("Now insert a record")

execute(conn,"insert into migrant values (3,2,'Philipines','yearly',82000,'UNS',1)")

rows = select(conn,"select \* from migrant")

show(rows)



# PHP Programming

HTML search form

<form action="search.php" method="POST">

    <input type="text" name="search" placeholder="search" >

    <button type="submit" name="submit-search">search</button>

</form>

Php script that uses search criteria and displays result

<?php

include\_once("config.php");

include\_once("Database.php");

$db = new Database();

?>

<h1>search page</h1>

<div>

<?php

if(isset($\_POST['submit-search'])){

    // prevent SQL injection

//$search=mysqli\_real\_escape\_string($db->dbh, $\_POST['search']);

$search= $\_POST['search'];

$db->query("SELECT \* FROM migrant where

id LIKE '%$search%'

-- OR

-- number LIKE '%$search%' OR

-- area LIKE '%search%'OR

-- series LIKE '%search%'OR

-- source LIKE '%search%'

");

$results = $db->resultSet();

foreach ($results as $result) {

    echo $result->id . "\t" . $result->number . "\t" .

       $result->area . "\t" . $result->value;

}

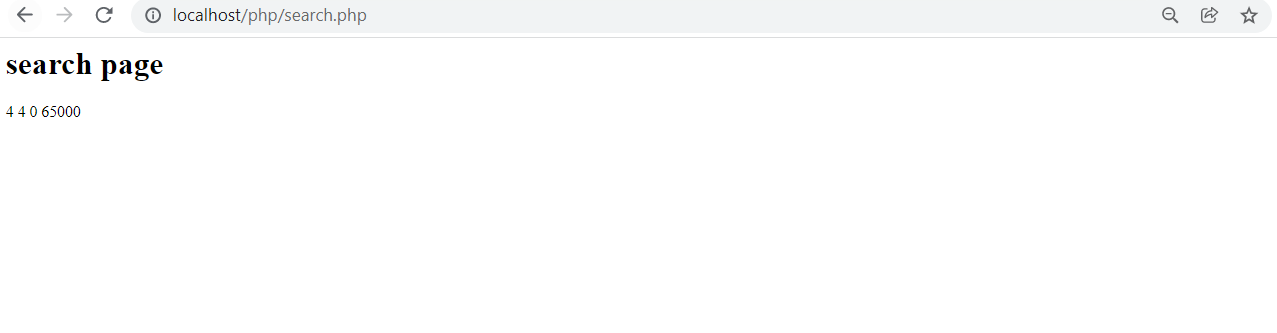
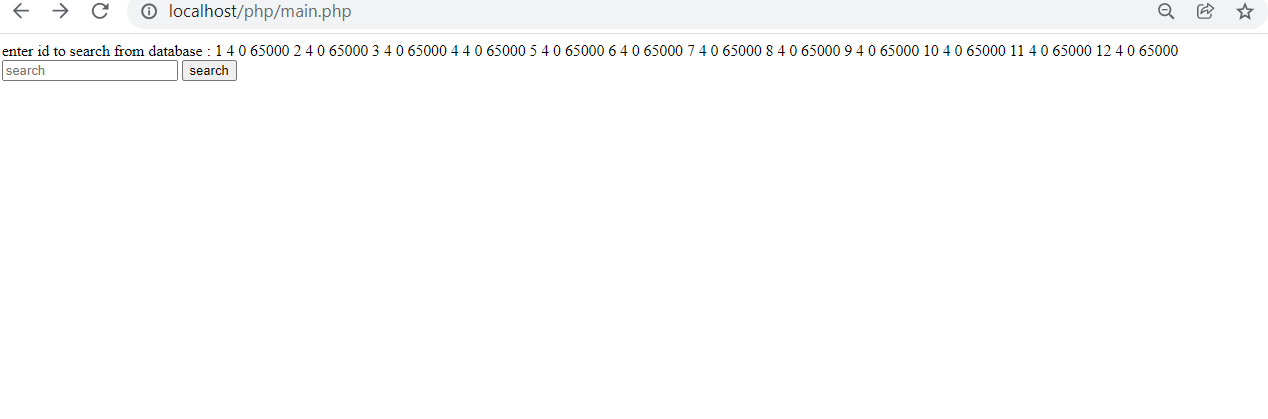
}

?>

</div>

SQL injection is caused by using search field to enter malicious queries. I have prevented it by using PDO approach

Website



# Suggested Future Work

We can only access this database in localhost at the moment we will deploy it onto some remote server and use it through UI created for web.

With using nosql we will be able to handle large volumes of data at high speed with a scale-out architecture.

We will be able to easily update schema in nosql.

We will take advantage of cloud to deliver zero downtime.

# Activity Log

Vineeth Reddy Gaddam found the data source. Data description was also completed

Premkumar baddam attached the data to csv files.

Sushanth Reddy Teegalaresearched and described the techniques to store data in relational databases

Vineeth Reddy Gaddam did the design till ERdiagram

Premkumar baddam worked DDL

Sushanth Reddy Teegala worked with DML

Vineeth delivered indexes work.

Premkumar-Views

Sushant-Trigger, transaction, future work

Vineeth-Security locking, backup

Vineeth Reddy Gaddam made changes to python script to grab data from our database

Premkumar and Sushanth did html code and php scripts to work with search and listing of data from database