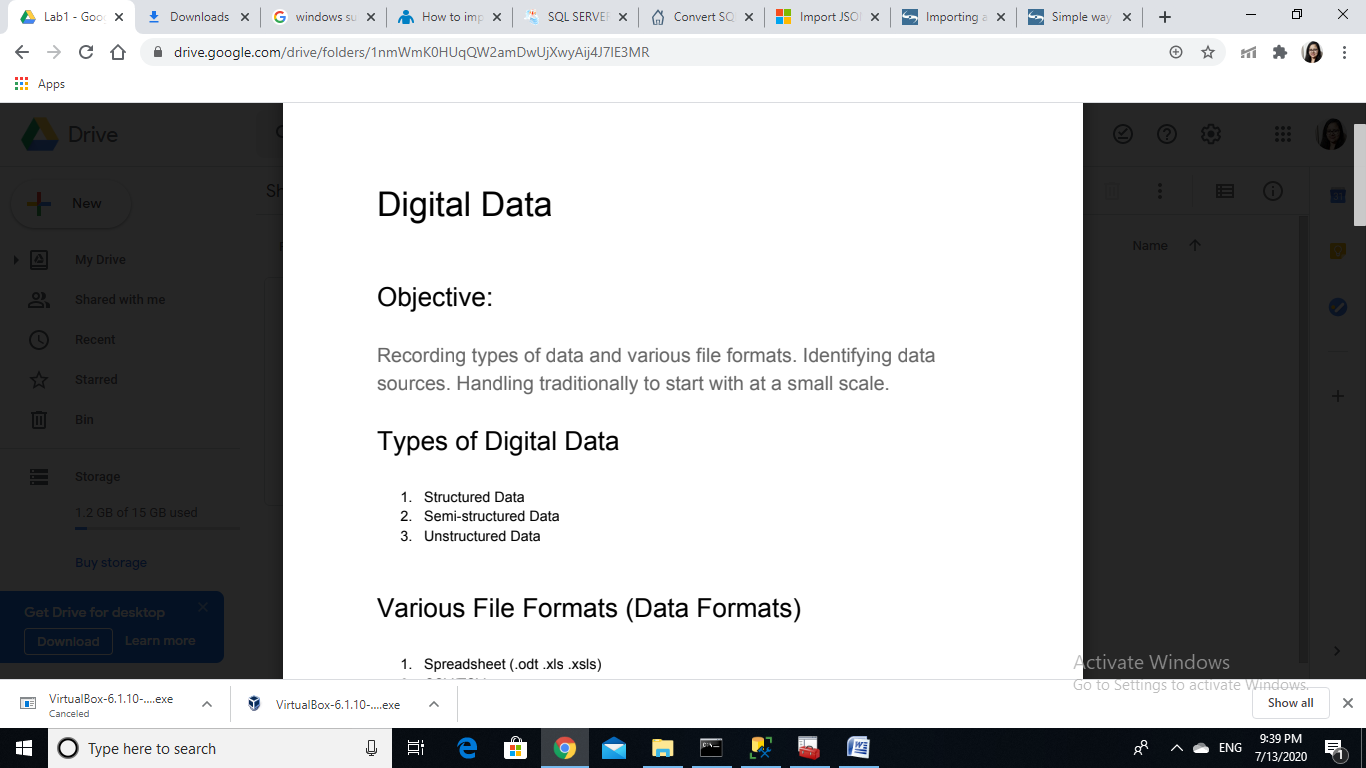
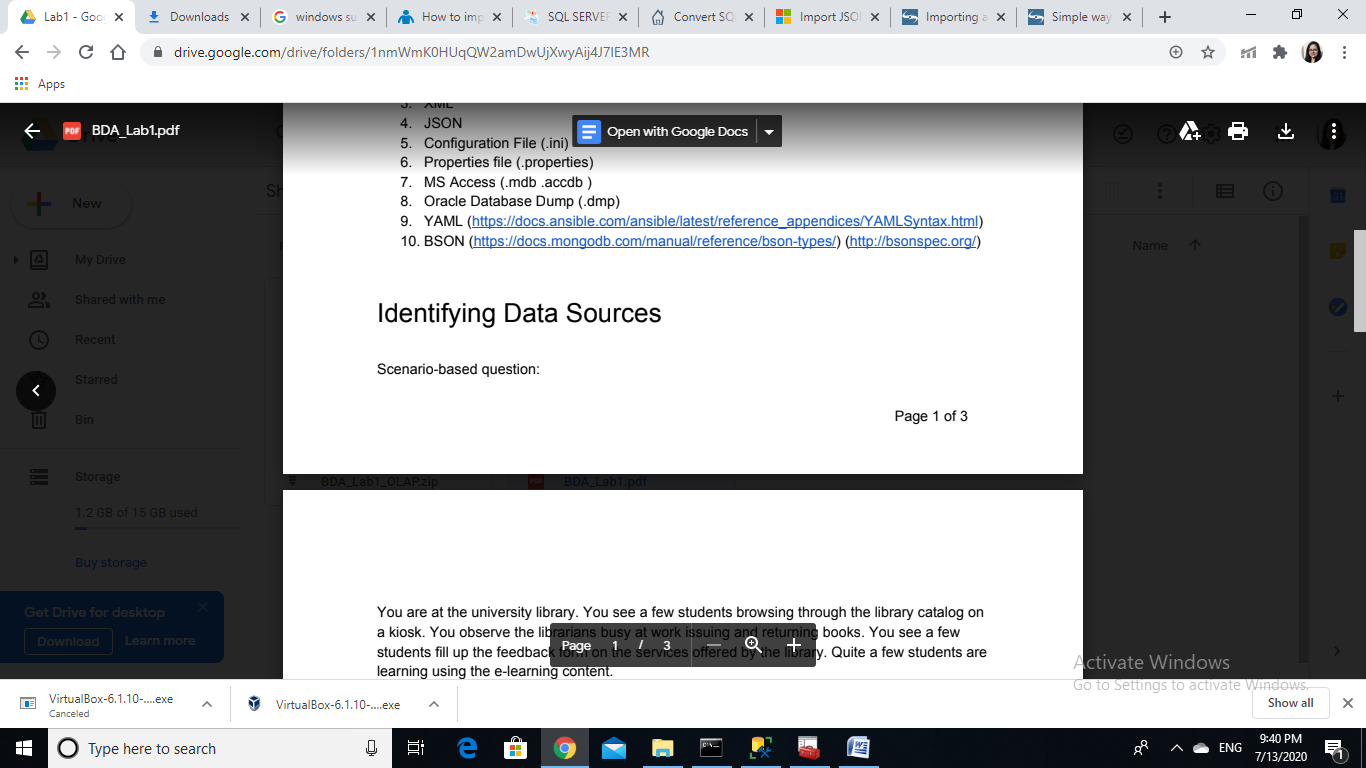
CE142 BDA LAB 1



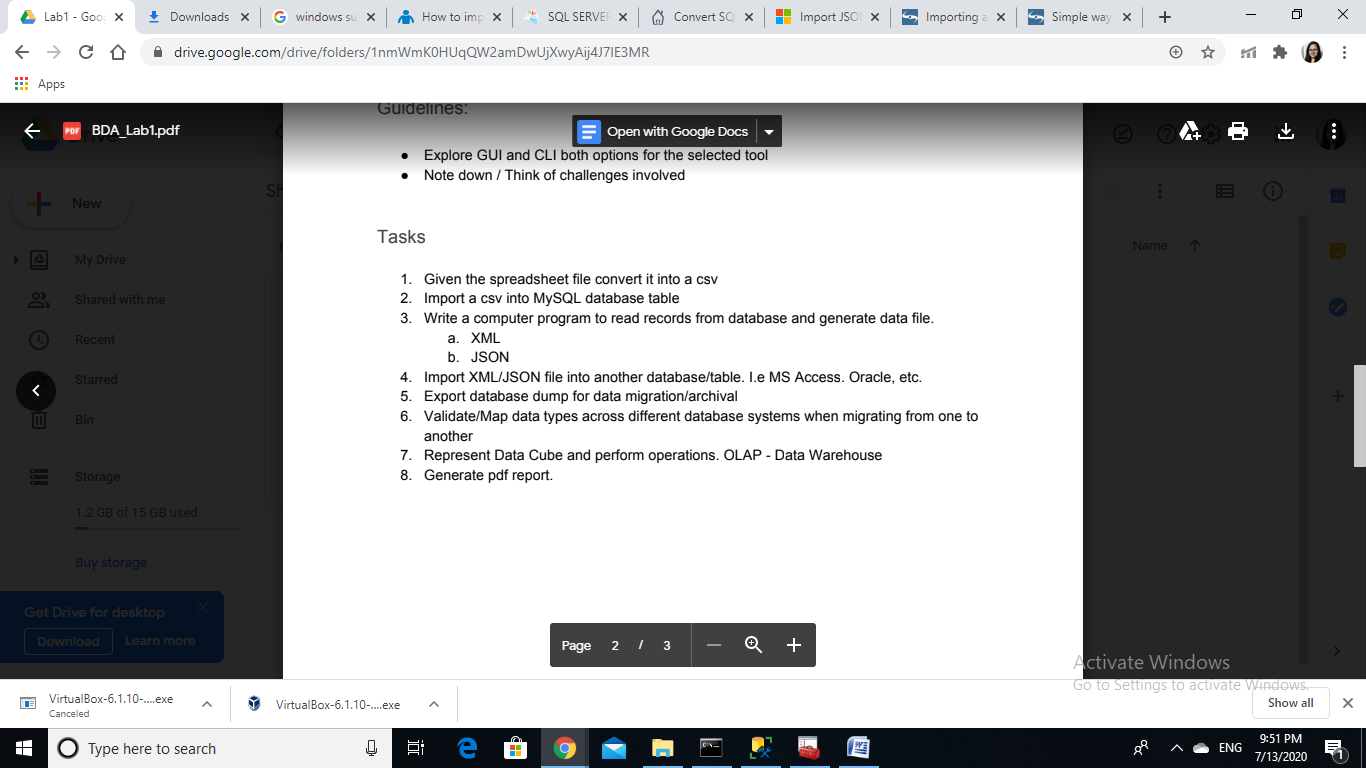


Kiosk: text, emails, web pages

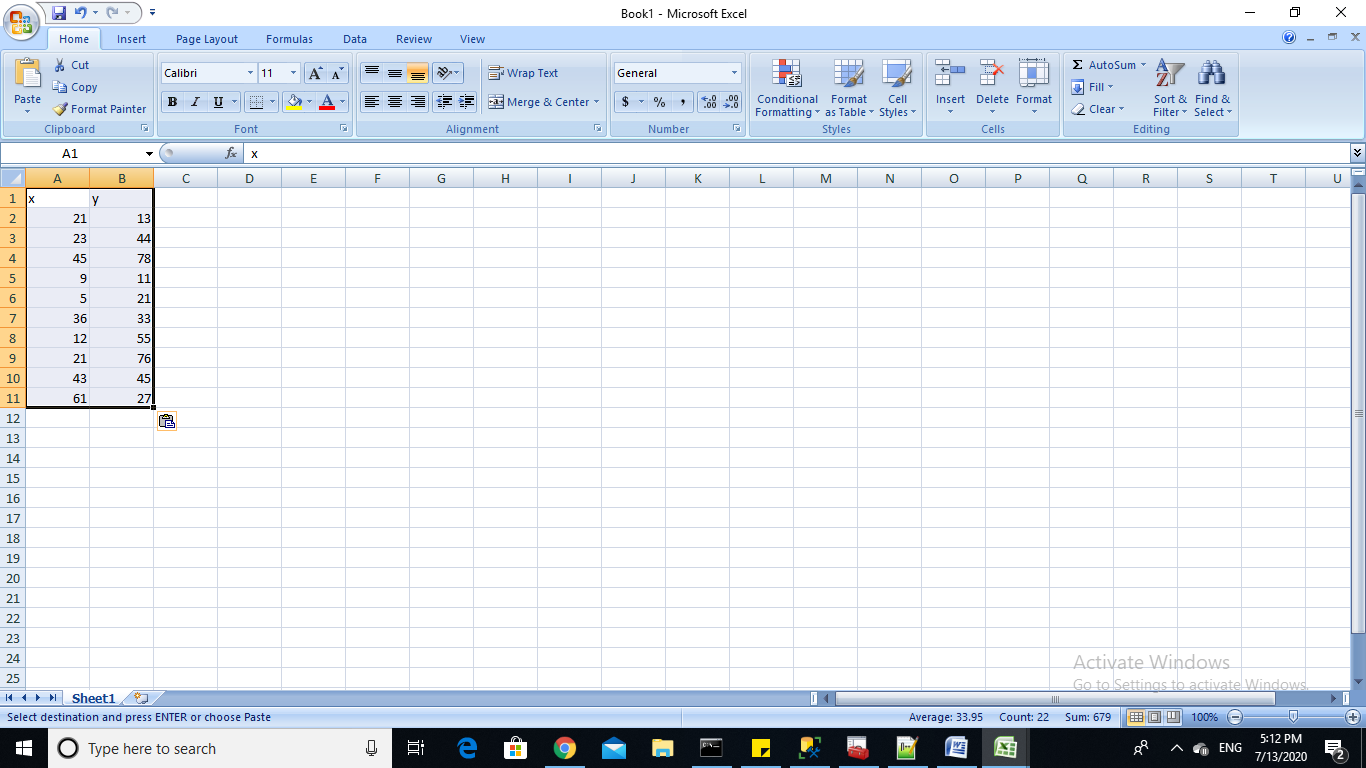
Library software: database files, Barcode scan related files, code files, executable files

Feedback form: google database files(assumed google form)

e-learning: pdf, images, videos, images, web pages etc.



1. **Spreadsheet to csv conversion**

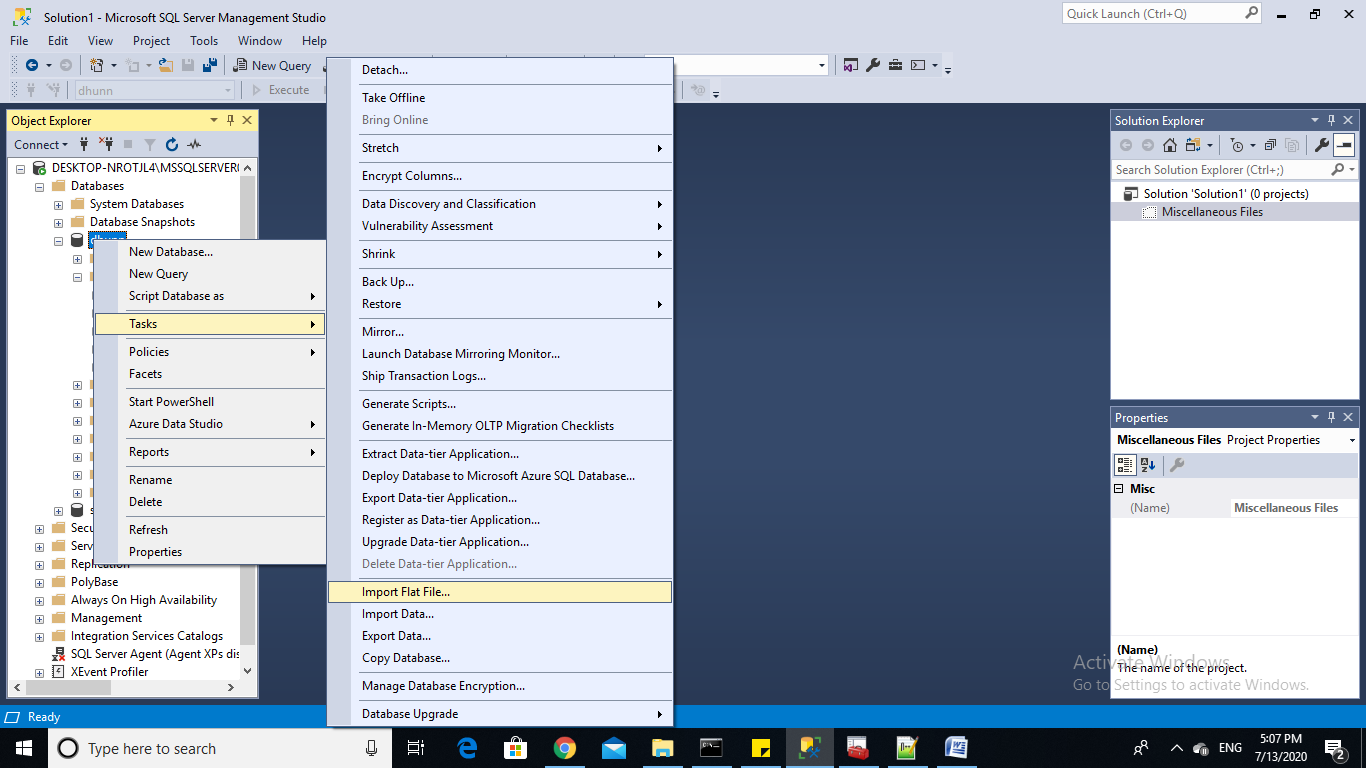
created demo spreadsheet.

While saving it choose csv file format, it will be saved in that particular format

Challenges: we have to care about labels and data types.

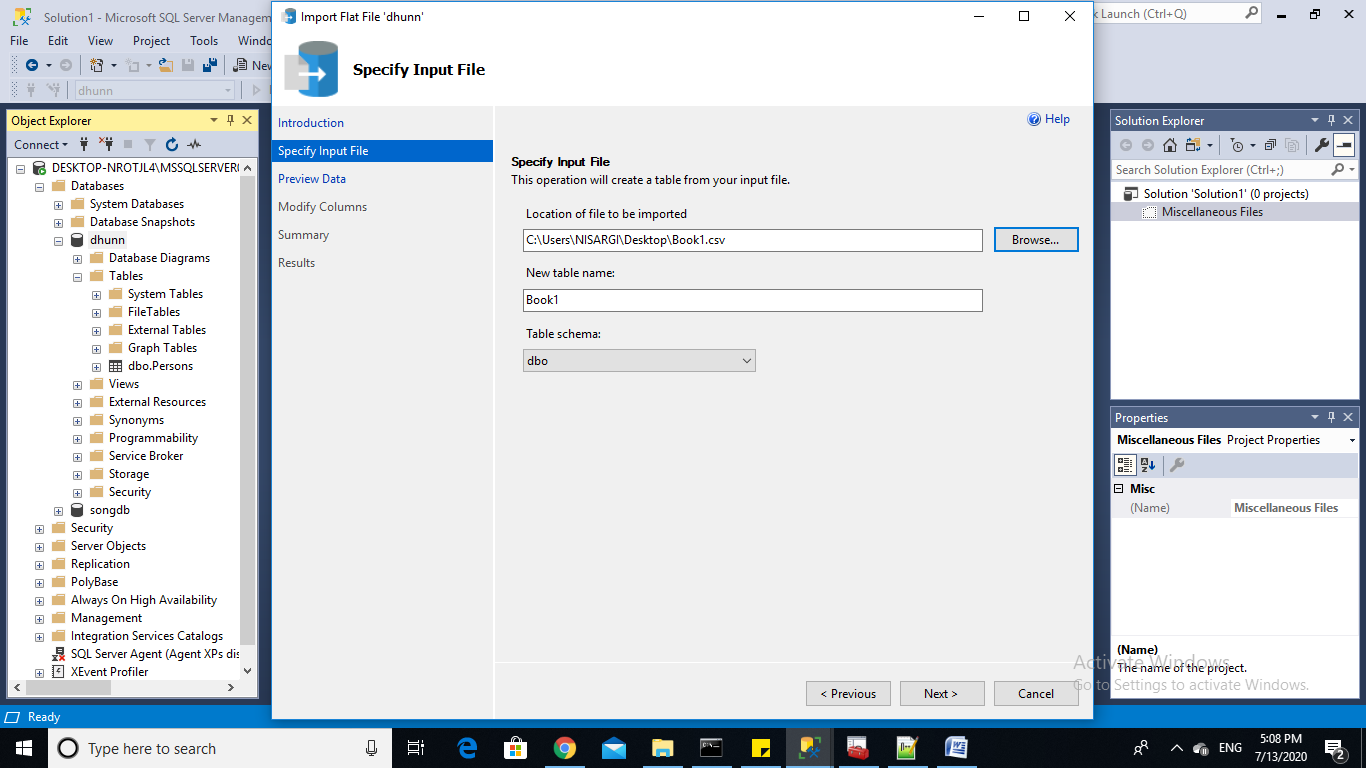
**(2)import csv to table**

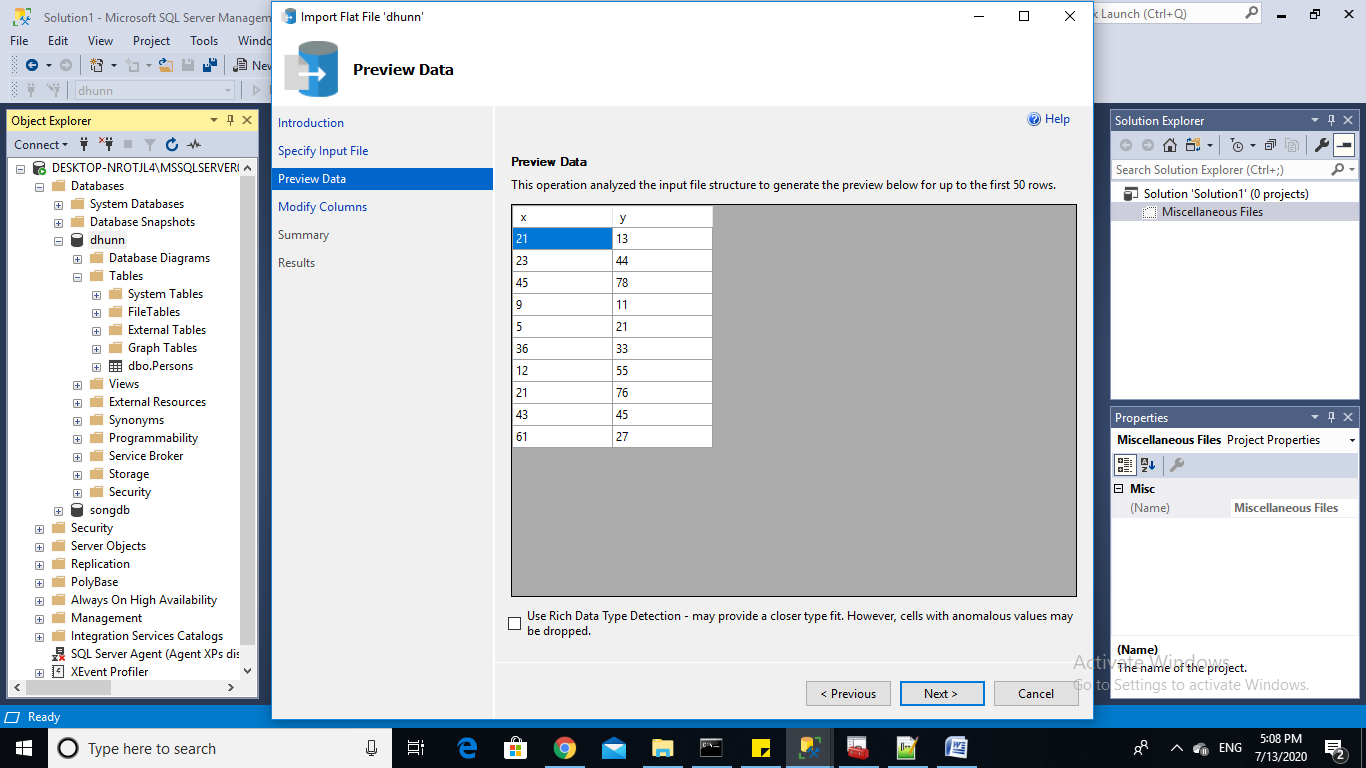
I have used sql server manager here.



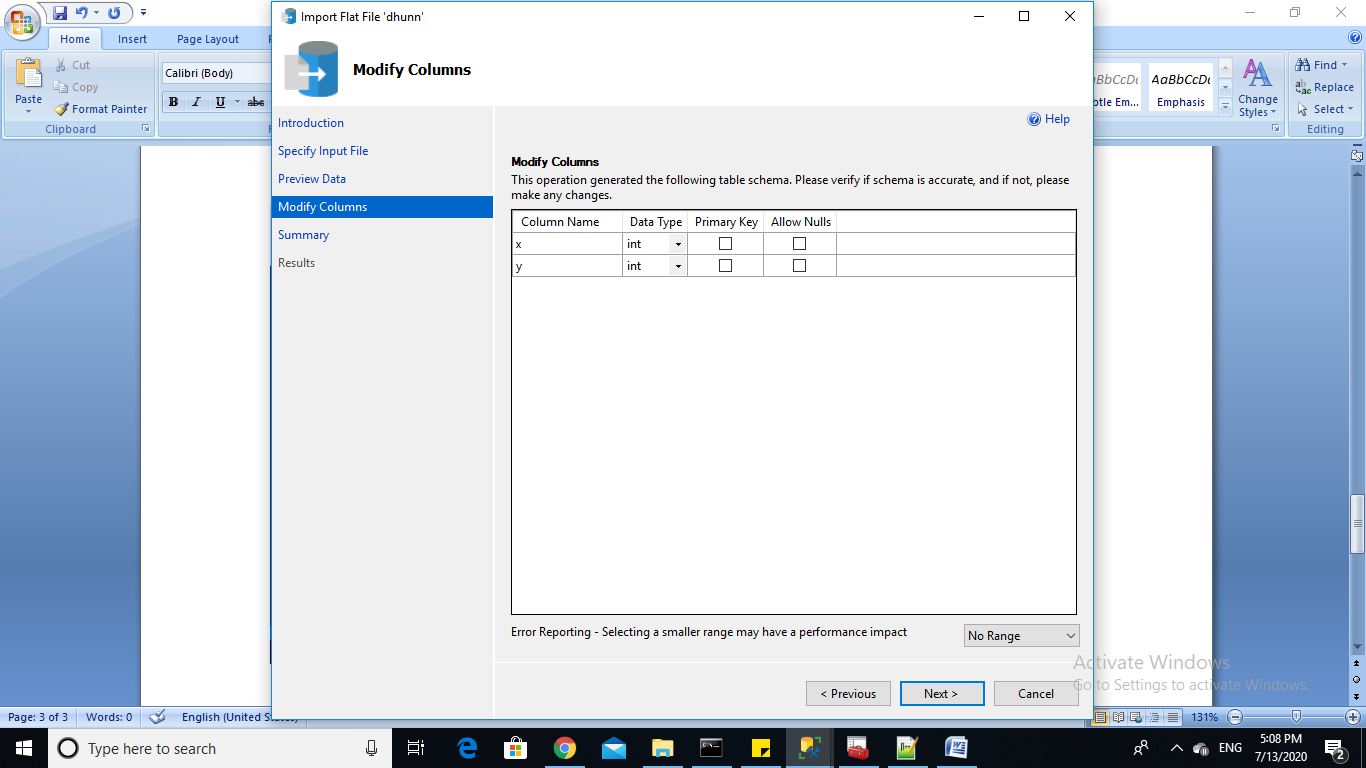
Steps: right click database ->tasks ->import flat file.

Select file from file browser.

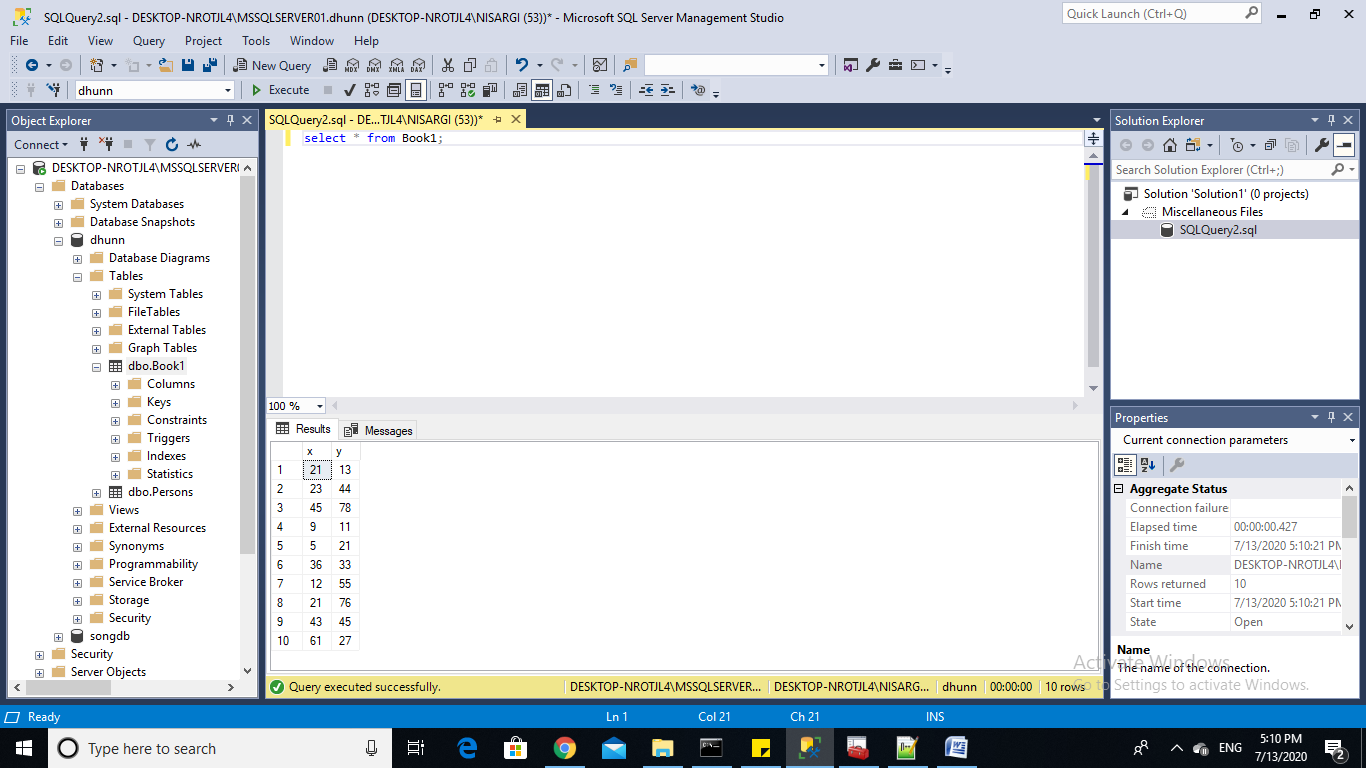




Confirm table details.



Output:

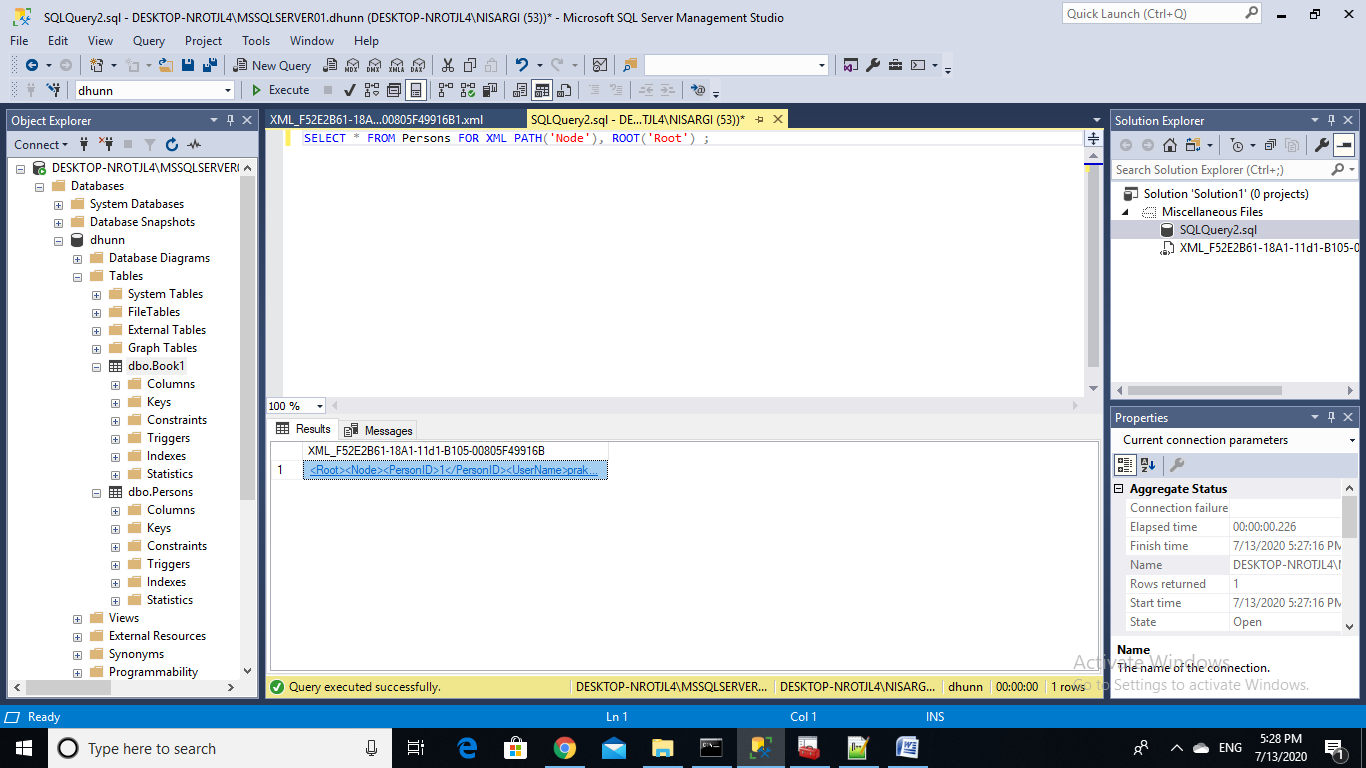


Challenges: data types and dependency should be taken care of.

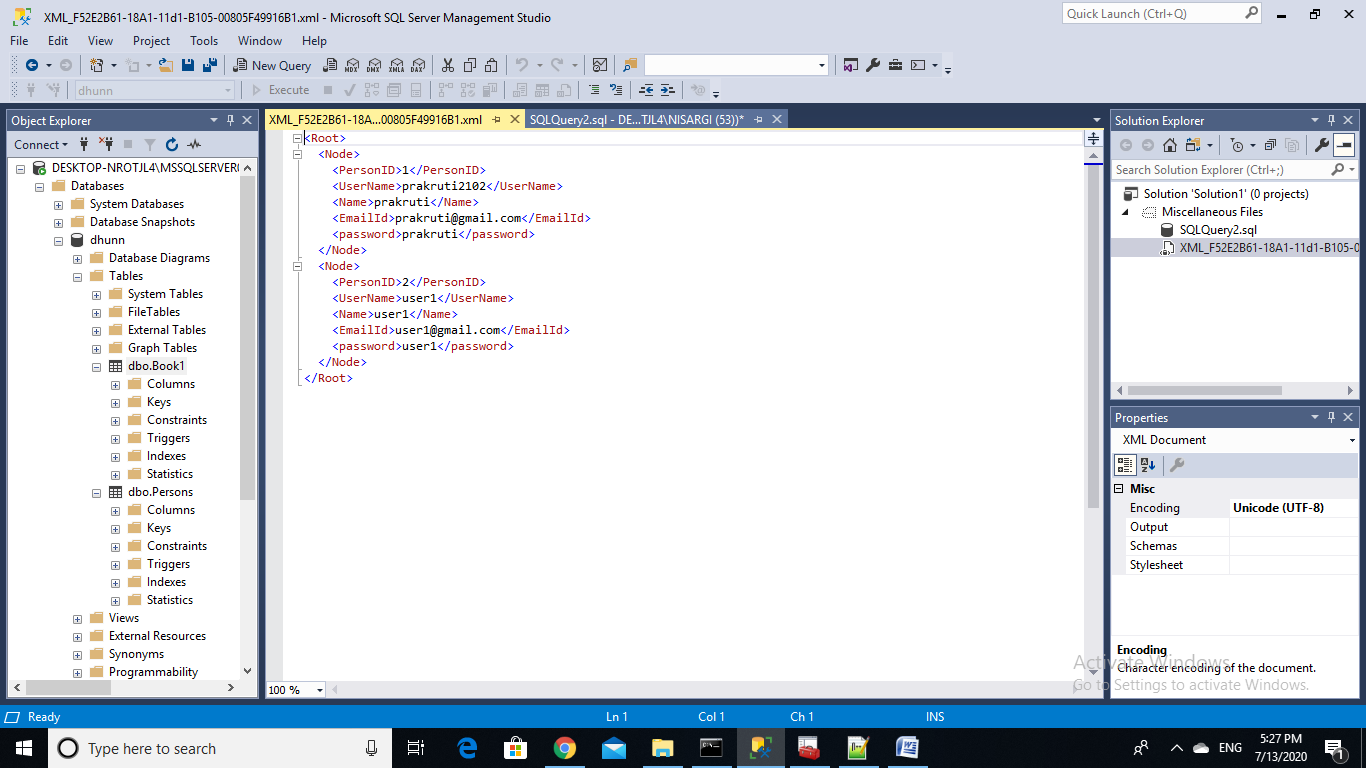
(**3) generate data file from database table**

(a) generate xml

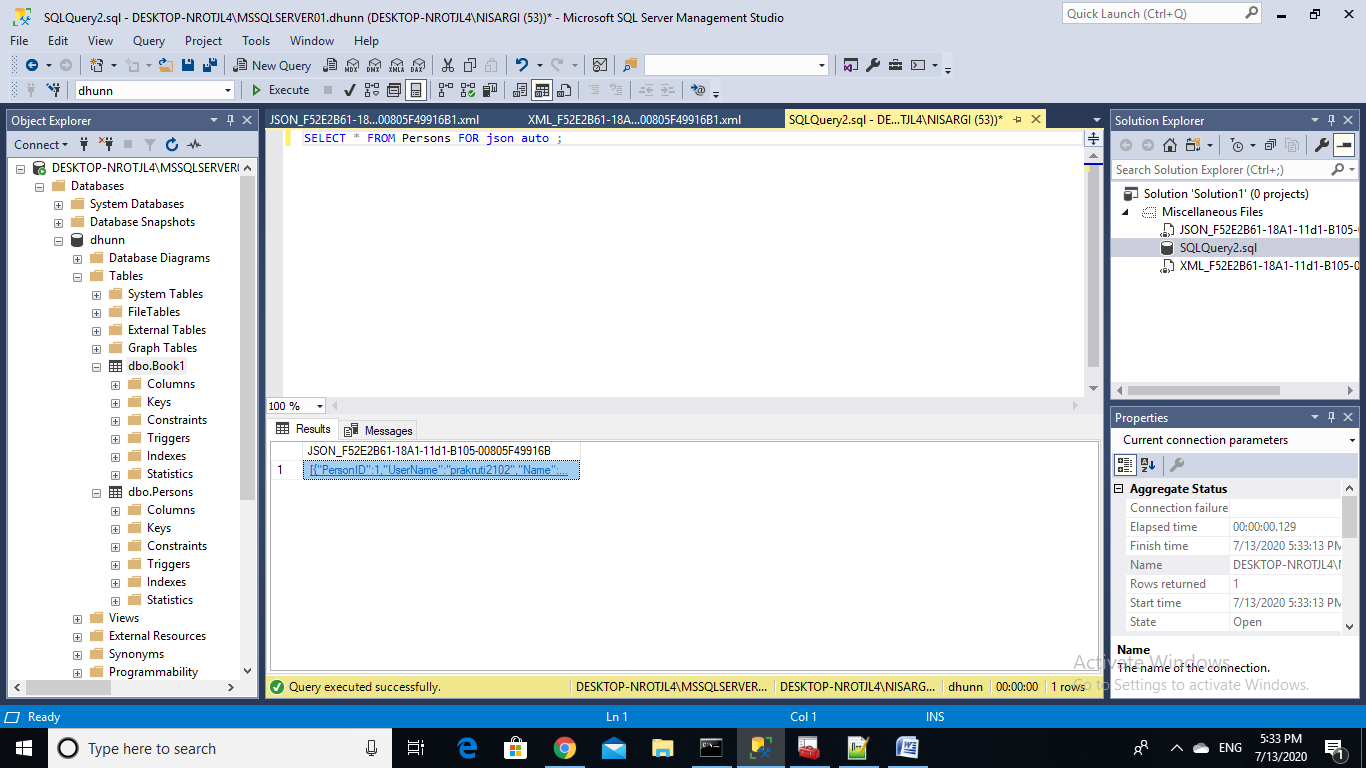
Execute following query

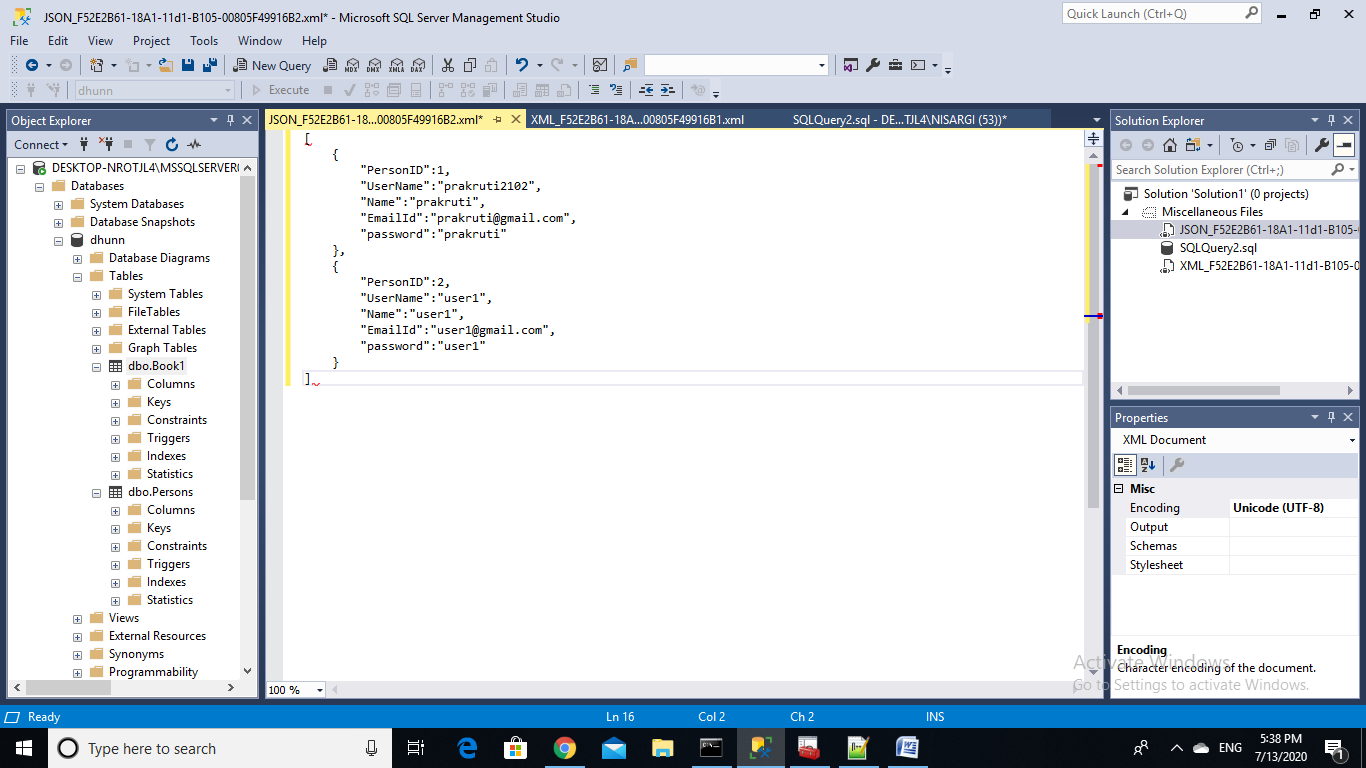
.

Output: save it at desired place.



(b) json file :

 run this query

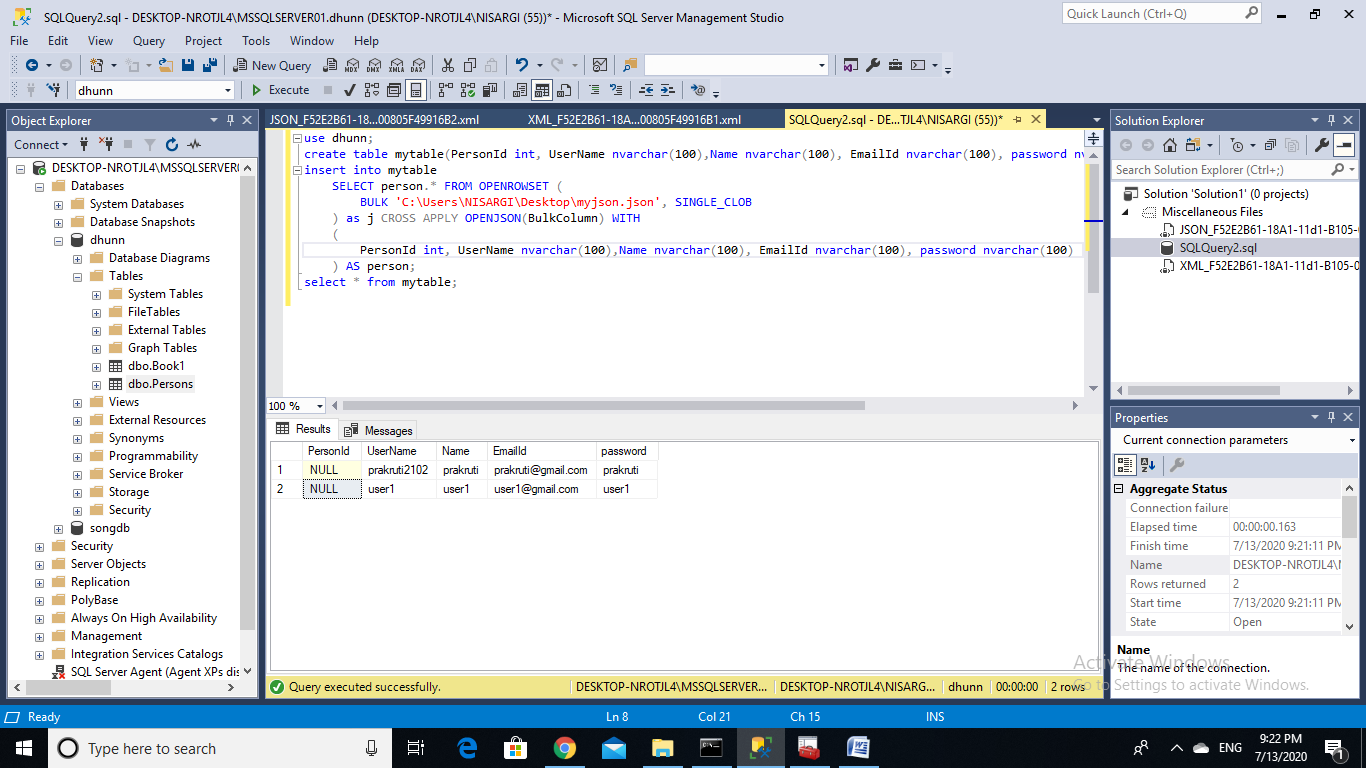


save this output file at desired place.

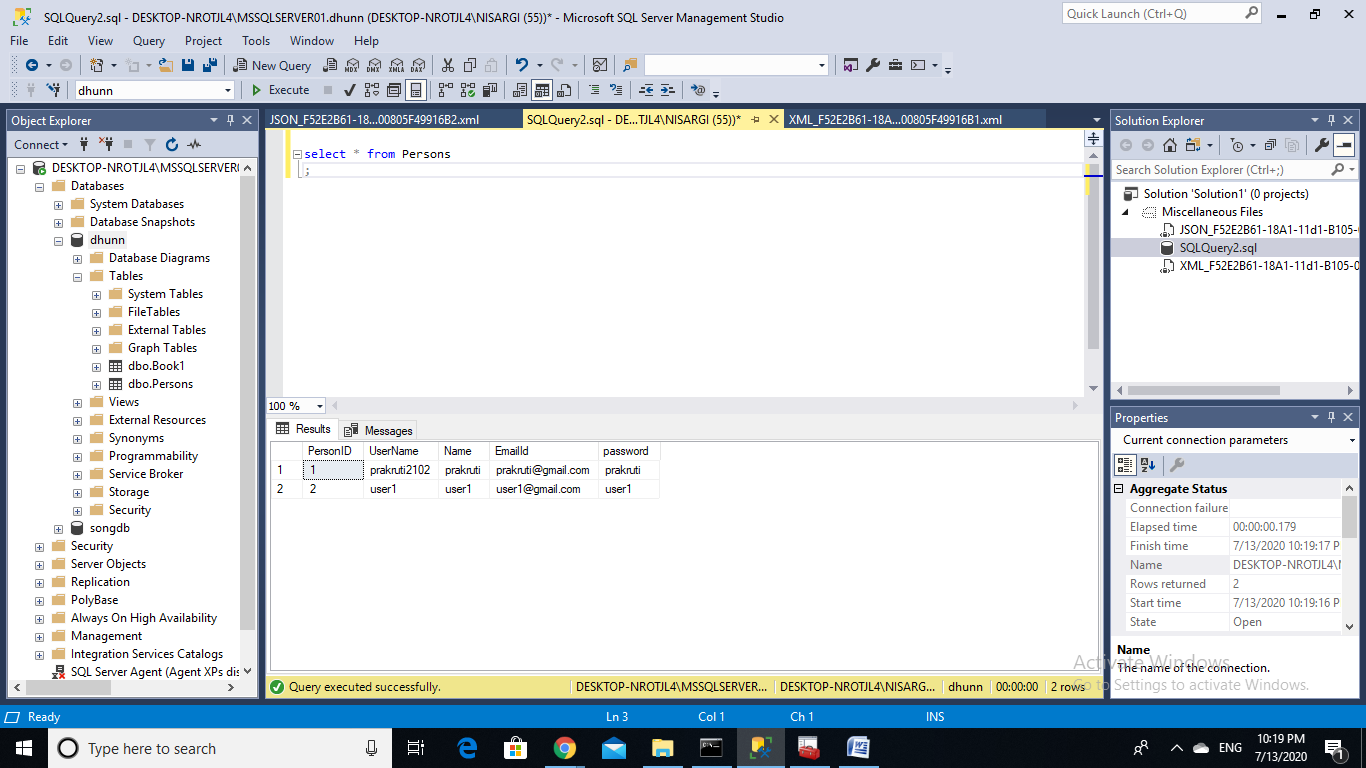
Challenges: schema and constraints related issues should be taken care of.

**(4) import xml/ json into database tables:**

JSON:

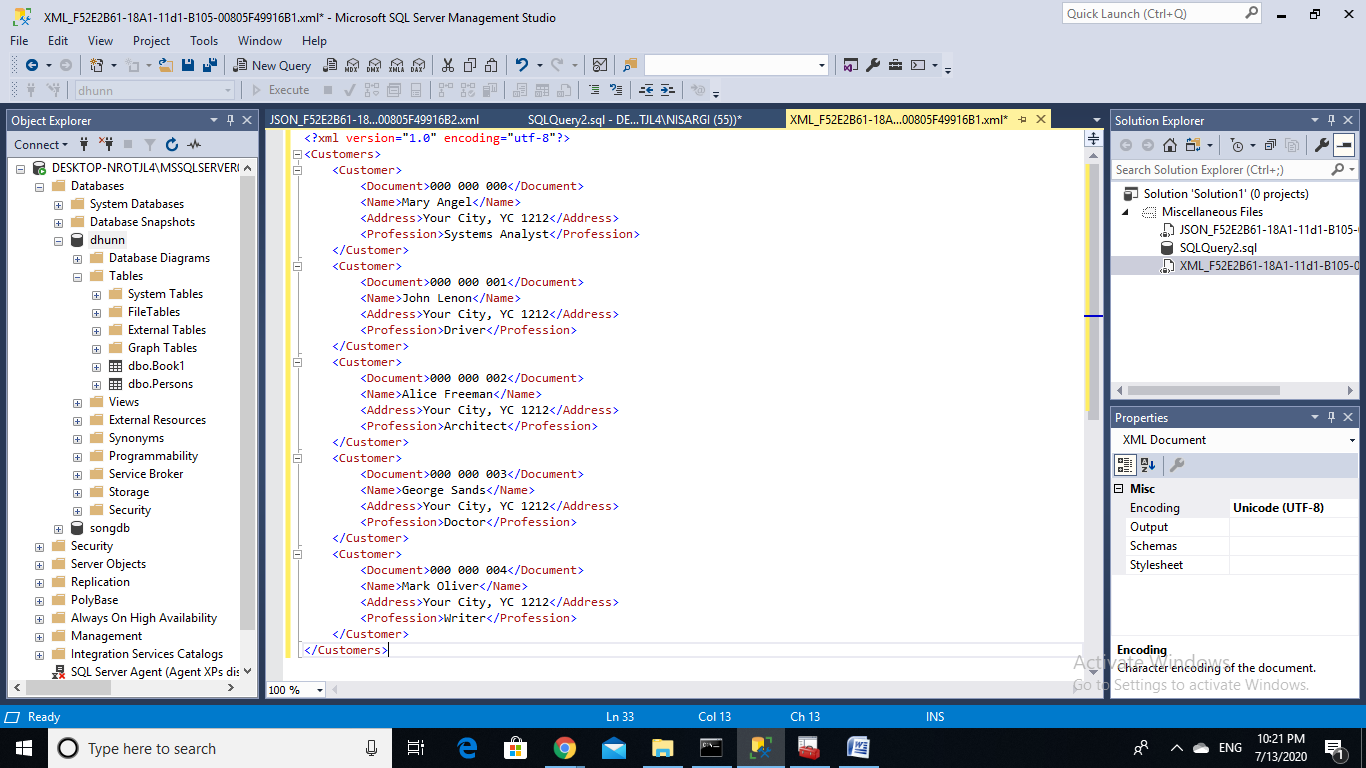


Output:

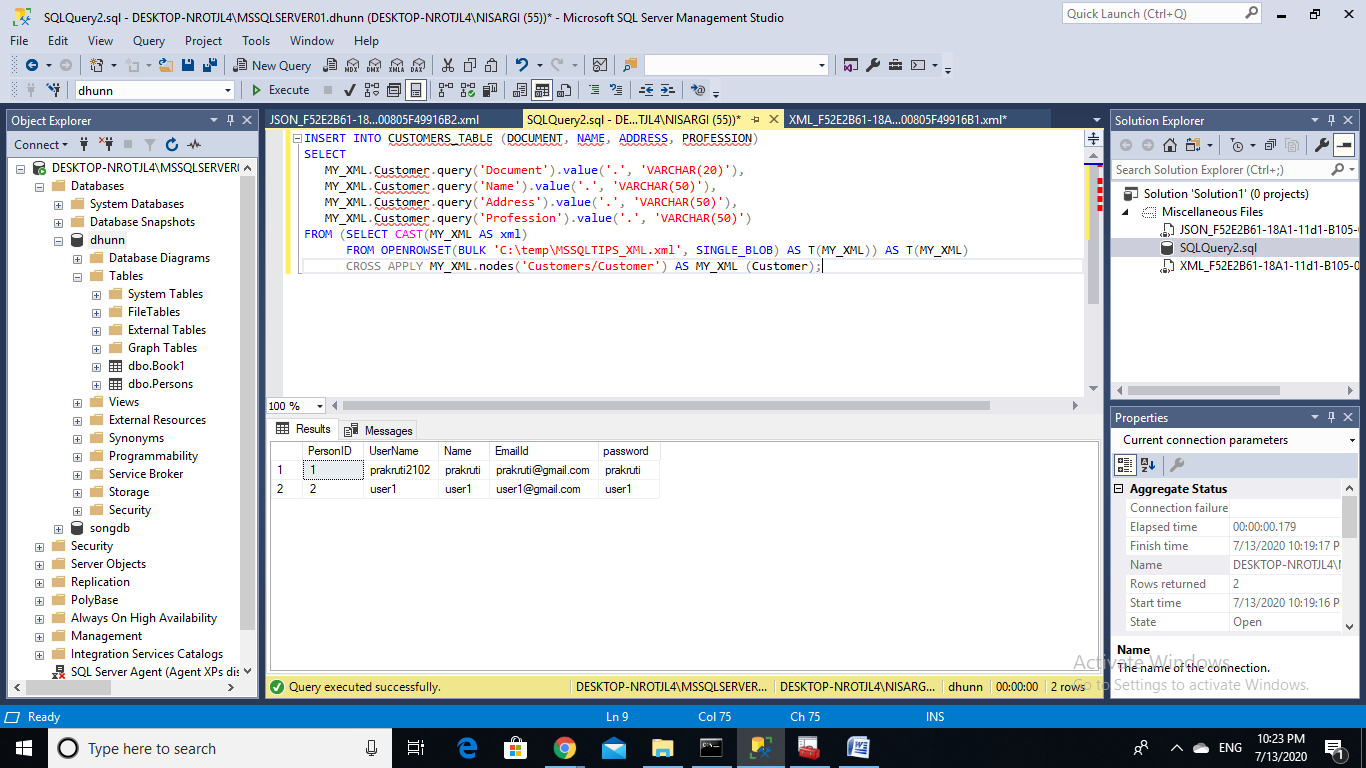


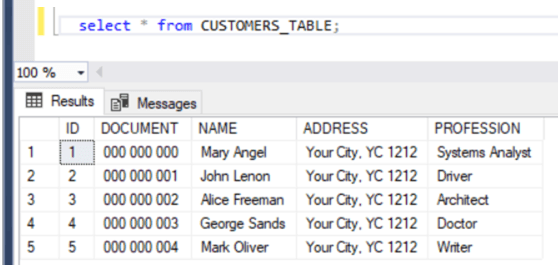
Challenges : columns should be known in this approach.

XML:



Run this query, you can observe output:

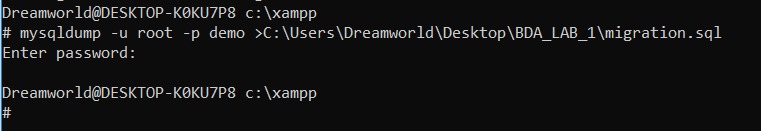




Challenges: schema and columns should be known, dependency should be preserved.

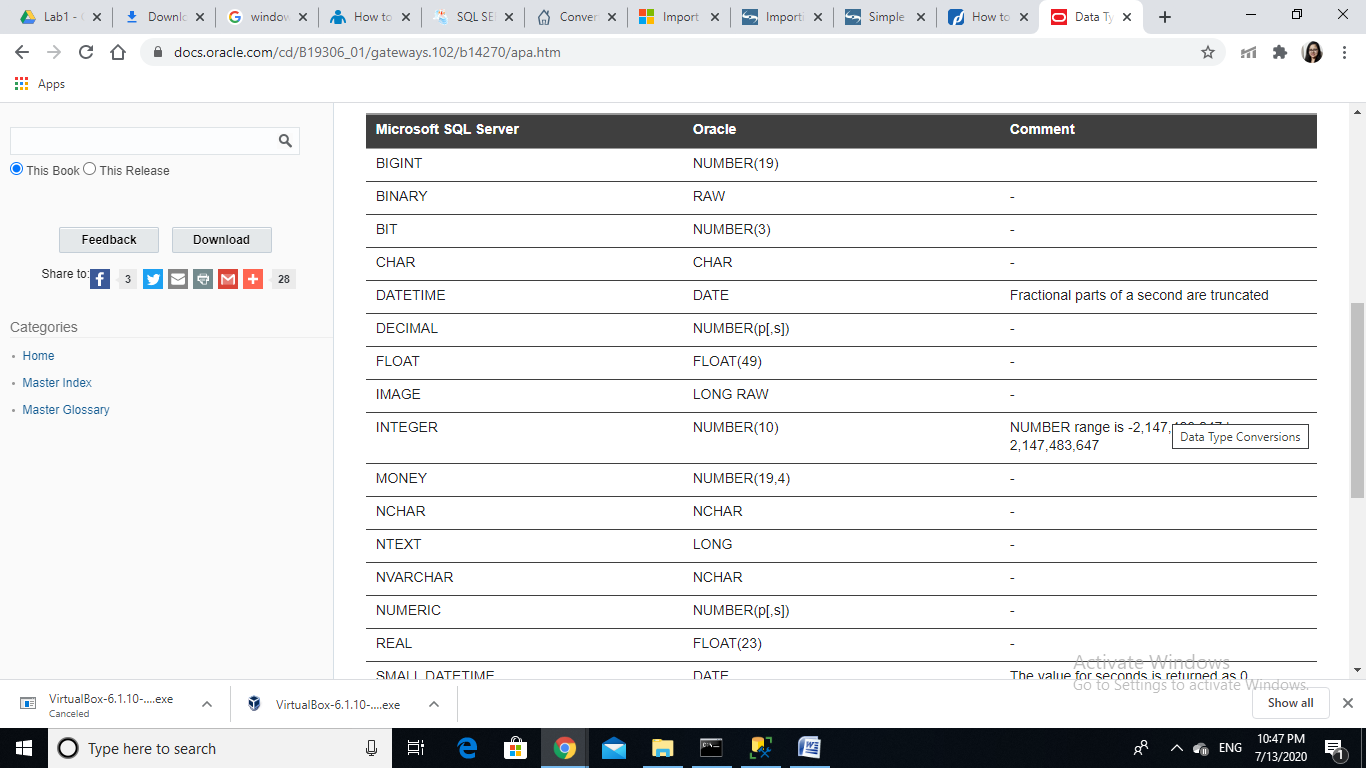
**(5) Export database dump for migration**

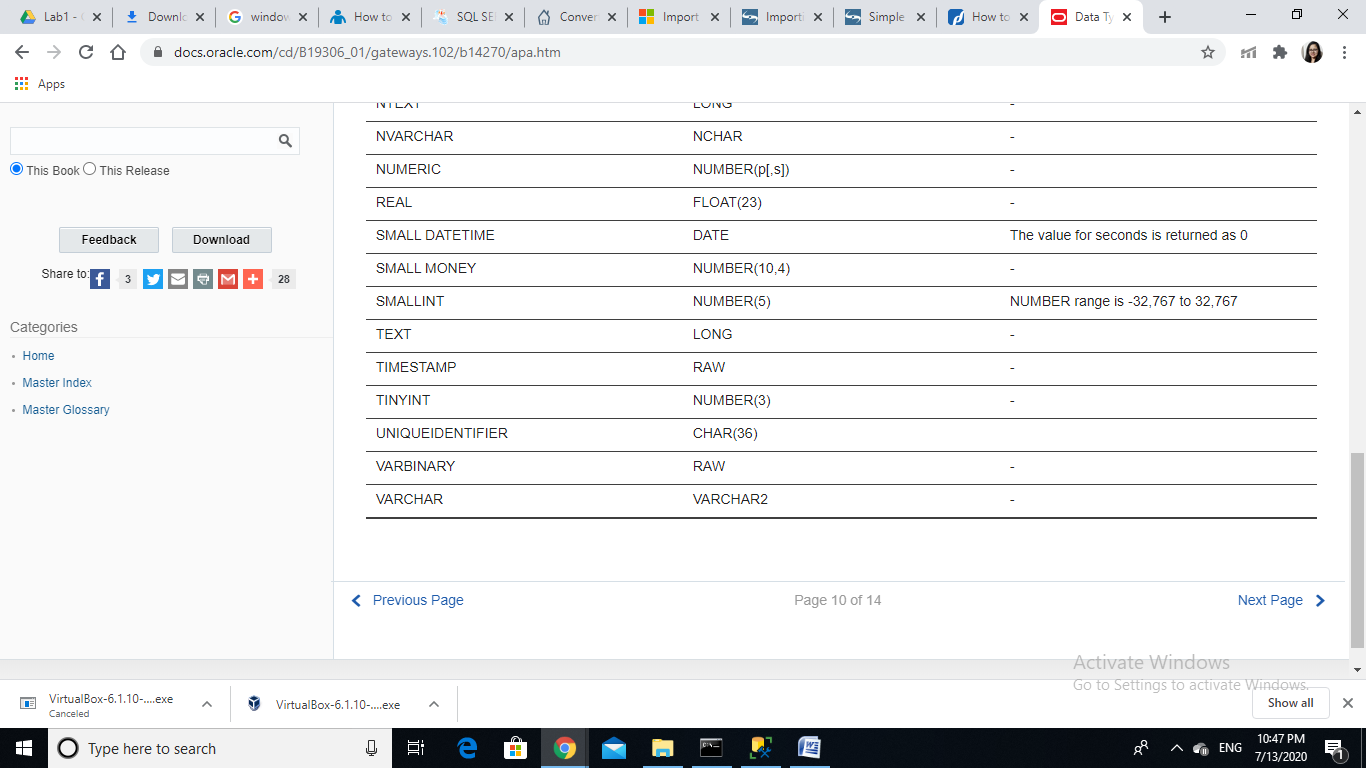
Run this command on xampp shell

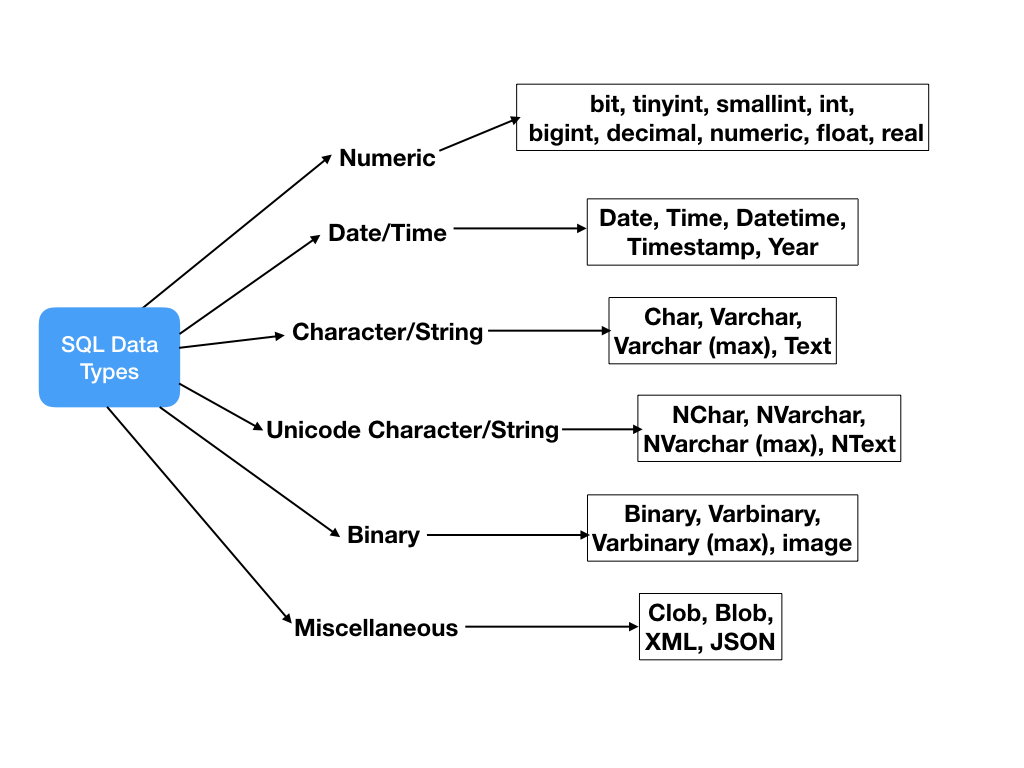


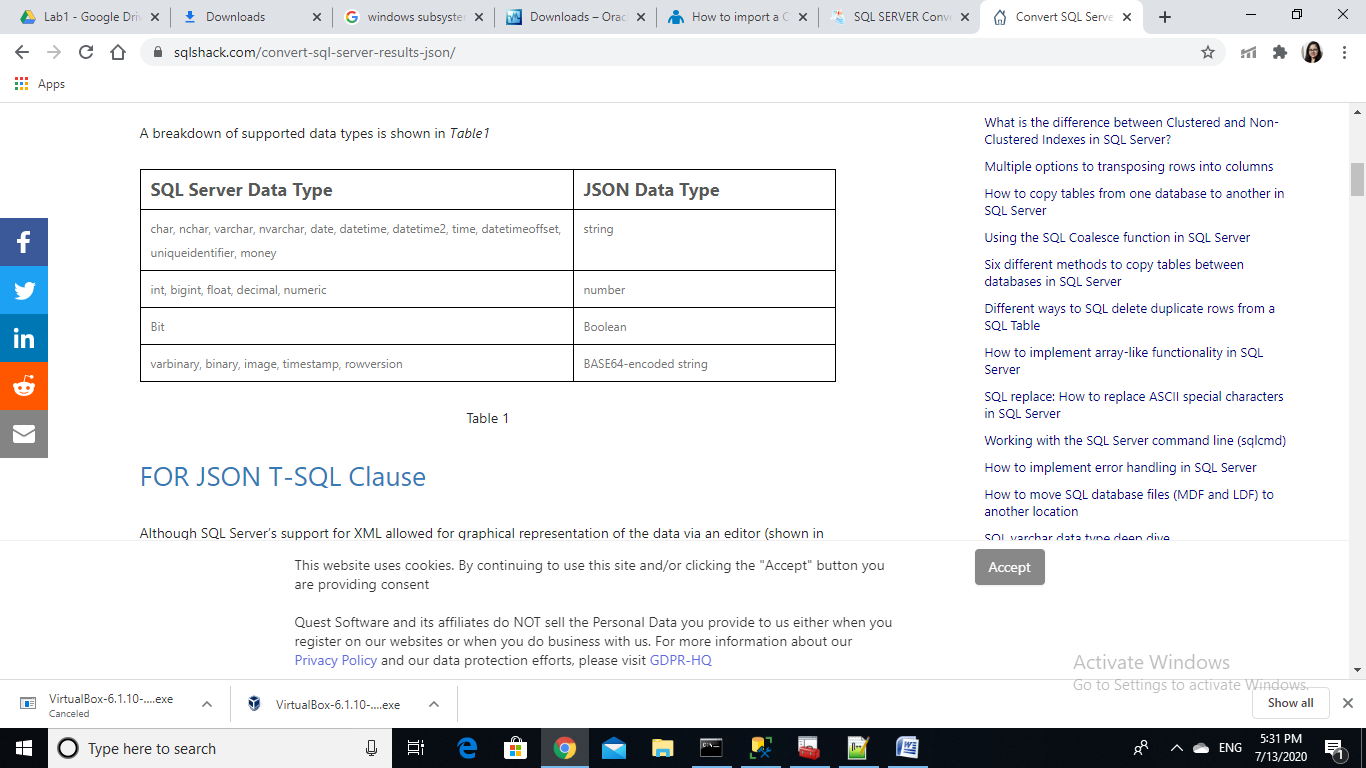
Your file will be created at desired destination

**(6) data types across different database systems**









NoSQL databases come in four core types — one for each type of data the database is expected to manage:

Columnar: Extension to traditional table structures. Supports variable sets of columns (column families) and is optimized for column‐wide operations (such as count, sum, and mean average).

Key‐value: A very simple structure. Sets of named keys and their value(s), typically an uninterpreted chunk of data. Sometimes that simple value may in fact be a JSON or binary document.

Triple: A single fact represented by three elements:

The subject you’re describing

The name of its property or relationship to another subject

The value — either an intrinsic value (such as an integer) or the unique ID of another subject (if it’s a relationship)

For example, Adam likes Cheese. Adam is the subject, likes is the predicate, and Cheese is the object.

Document: XML, JSON, text, or binary blob. Any treelike structure can be represented as an XML or JSON document, including things such as an order that includes a delivery address, billing details, and a list of products and quantities.

Some document NoSQL databases support storing a separate list (or document) of properties about the document, too.

**(7) Dicing and Slicing**

OLTP and OLAP both are the online processing systems. OLTP is a transactional processing while OLAP is an analytical processing system. The basic difference between OLTP and OLAP is that OLTP is an online database modifying system, whereas, OLAP is an online database query answering system. We can utilize OLAP operations on our data in data warehouse to produce beneficial and fruitful results that can be used for strategic planning in future.

1) We need to create the dimensions table, i.e. those tables which store the basic descriptive information about the entity.

Example: Student Details Data Warehouse

Dimension 1 – Branch

CREATE TABLE Branch( id INT(6) AUTO\_INCREMENT PRIMARY KEY, category VARCHAR(255) NOT NULL, stream VARCHAR(255) NOT NULL );

Dimension 2 – Student

CREATE TABLE Student( id INT(6) AUTO\_INCREMENT PRIMARY KEY, gender VARCHAR(255) NOT NULL, category VARCHAR(255) NOT NULL );

Dimension 3 – Period

CREATE TABLE Period( id INT(6) AUTO\_INCREMENT PRIMARY KEY, type VARCHAR(255) NOT NULL, year VARCHAR(255) NOT NULL );

2) create a fact table that stores actual analytical data that we need.

Example:

create table fact\_Admission (

Branch\_ID integer references d1\_Branch,

Fellow\_ID integer references d2\_Student,

Period\_ID integer references d3\_Period,

Fact\_No\_Adm integer,

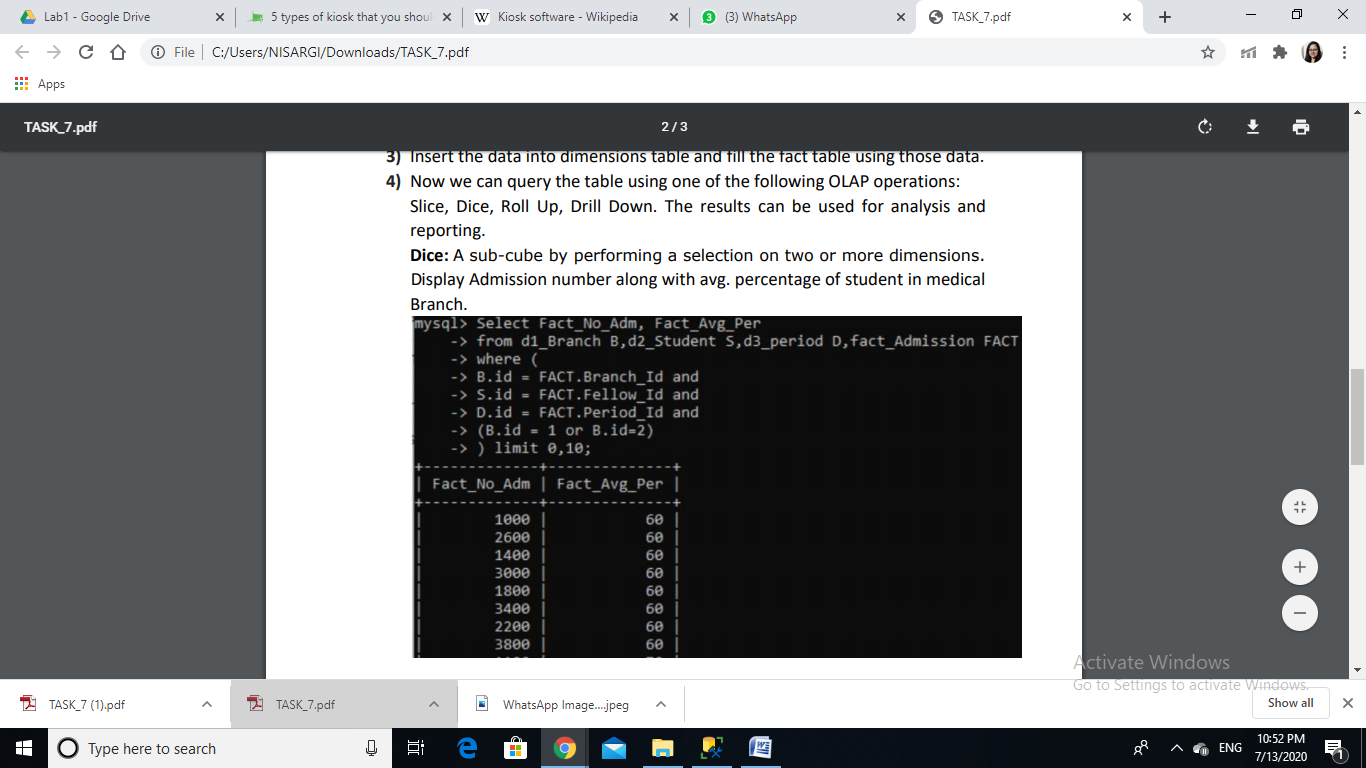
Fact\_Avg\_Per integer,

primary key (Branch\_Id,Fellow\_Id,Period\_Id));

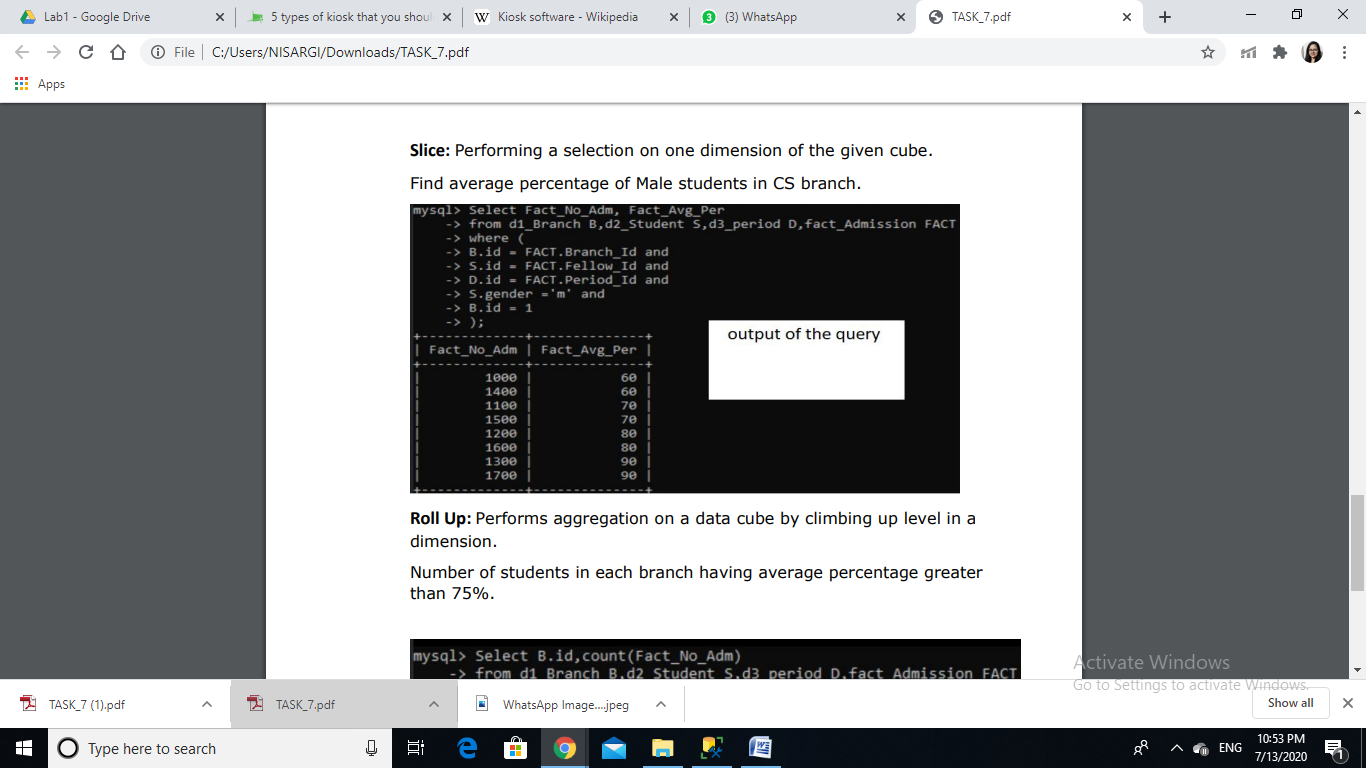
3) Insert the data into dimensions table and fill the fact table using those data.

4) OLAP operations: Slice, Dice, Roll Up, Drill Down

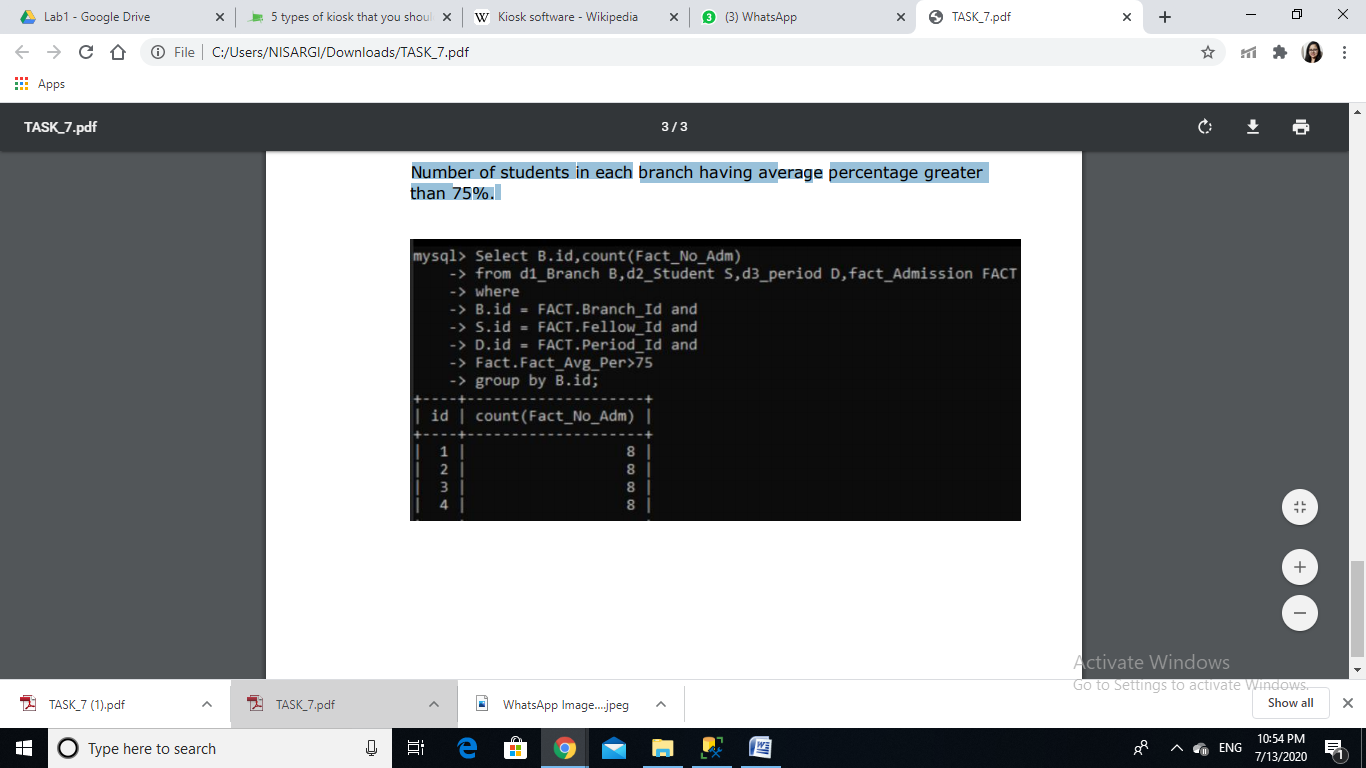
Dice: A sub-cube by performing a selection on two or more dimensions. Display Admission number along with avg. percentage of student in medical Branch.



Slice: Performing a selection on one dimension of the given cube. Find average percentage of Male students in CS branch.



Number of students in each branch having average percentage greater than 75%.



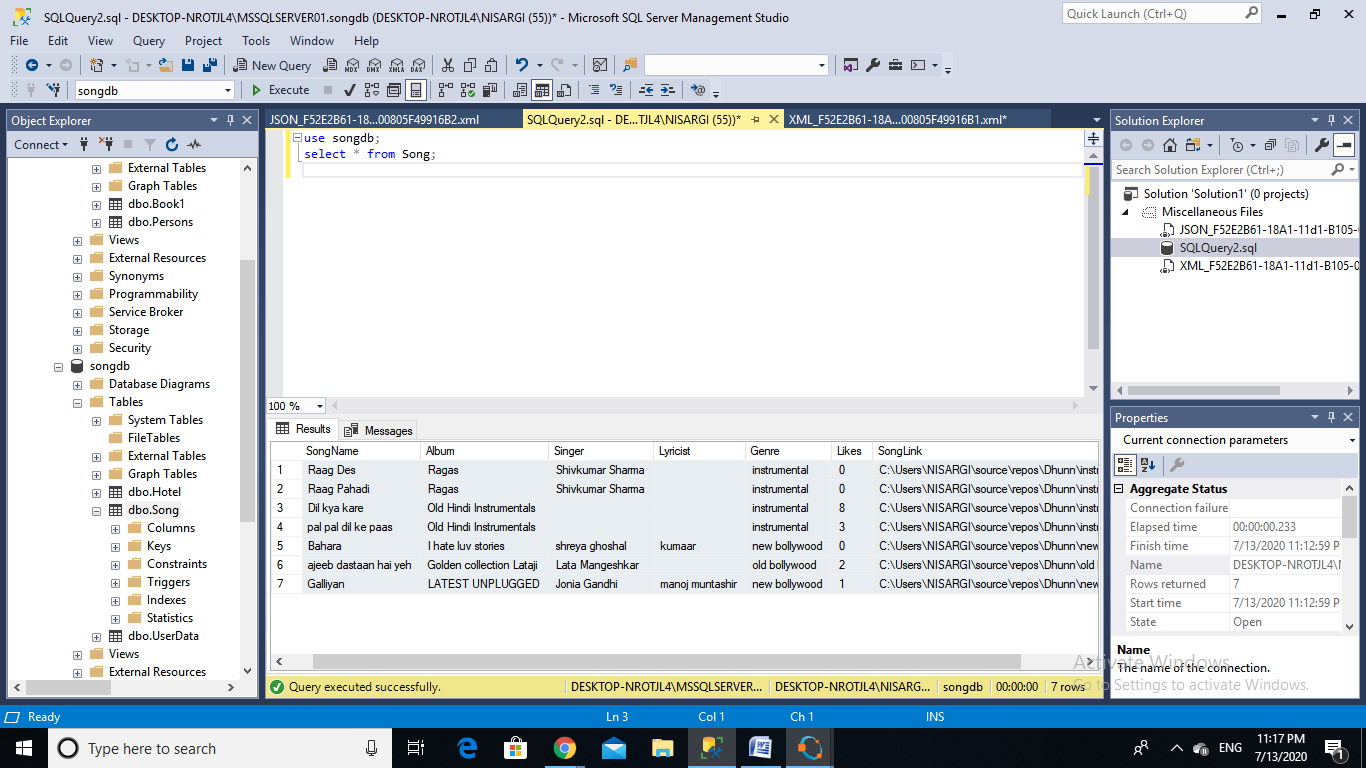
**(8) Report**

Songs table:

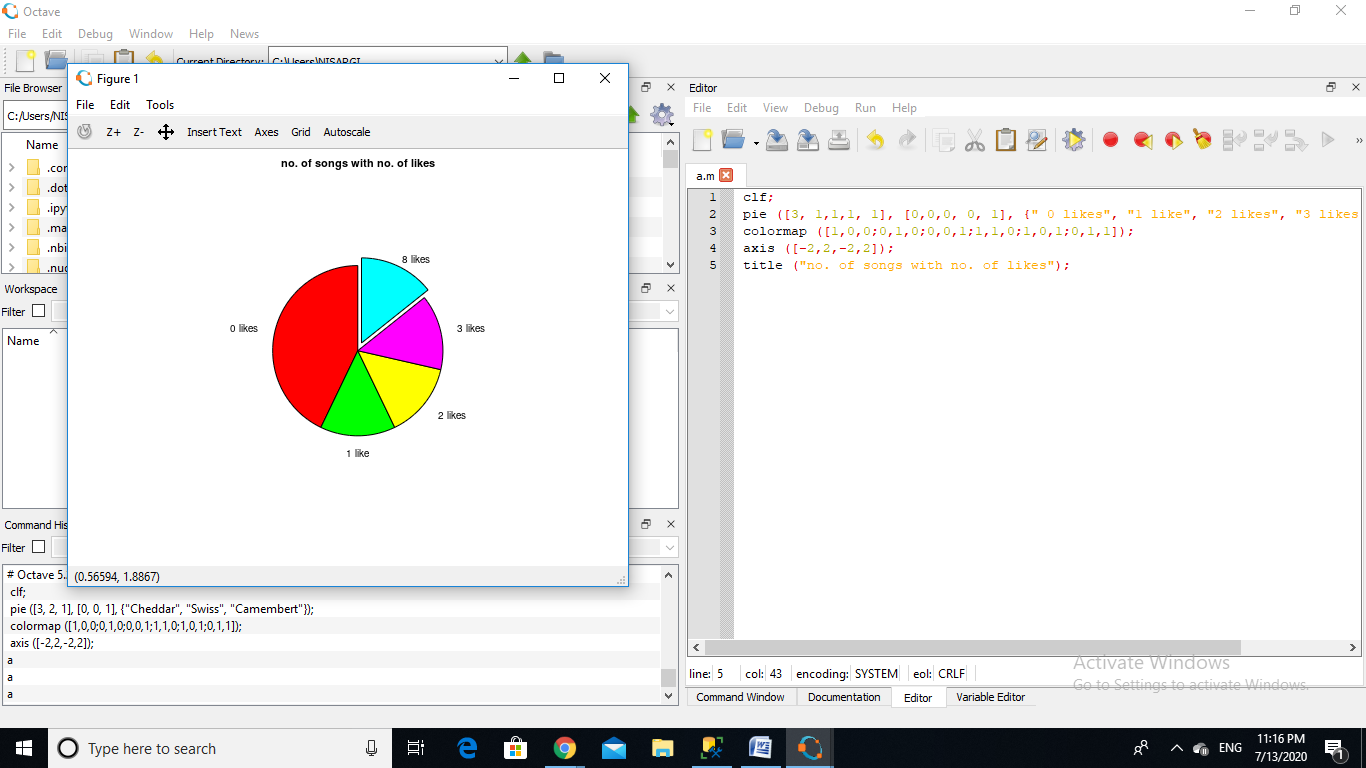
Primary key: id

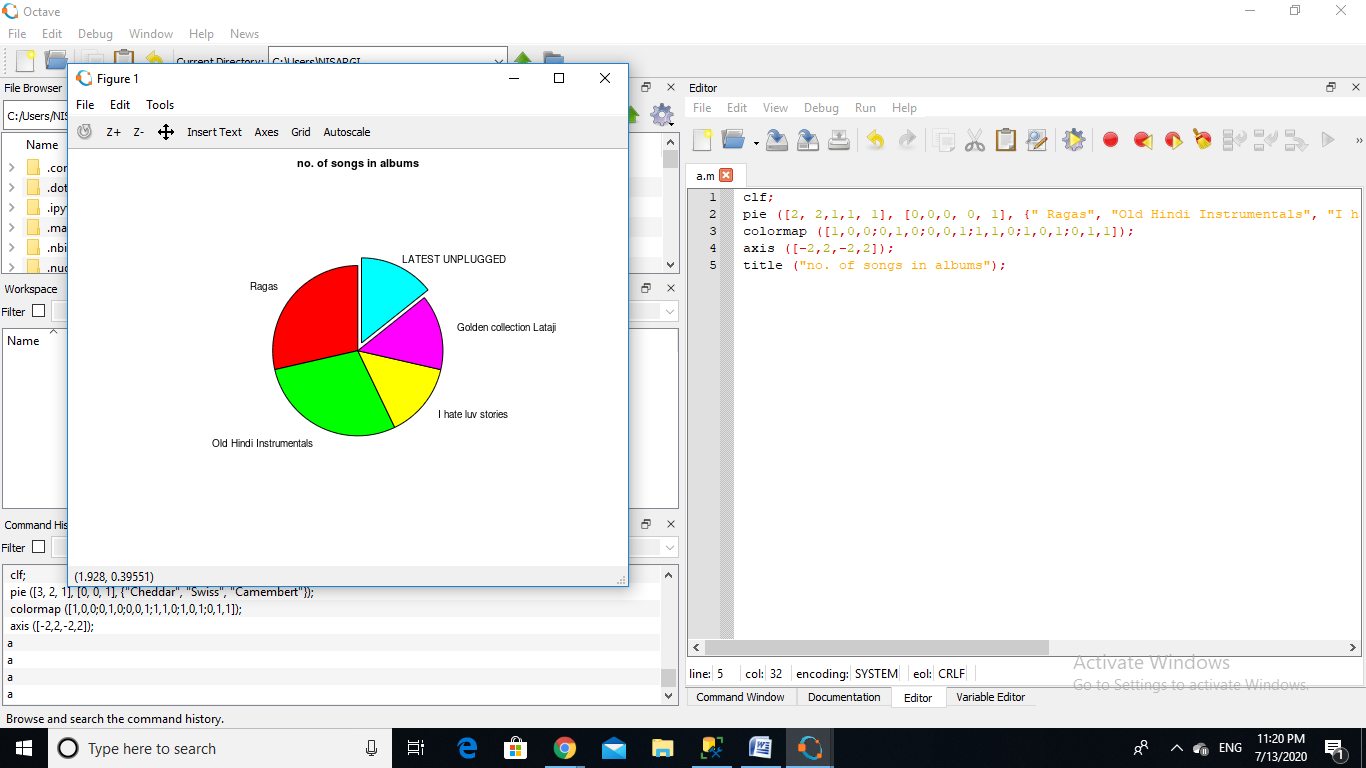
Containes 7 songs form various generes and albums:

Analysis of it is as below.

****

Pi chart for songs table:

****

****