Data Science

Data Mining Techniques

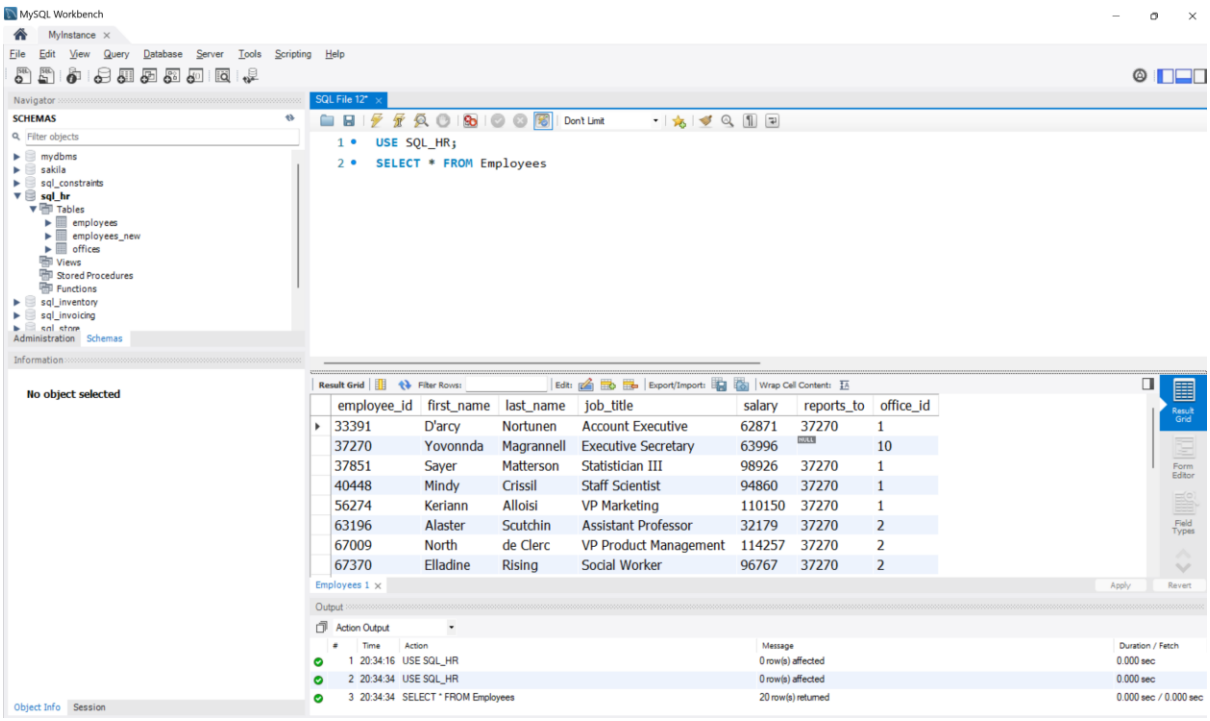
DDL Queries

Task 2:

Guidelines:

1. Use the appropriate database and table mentioned in the exercise to get the correct solution.
2. You need to download and install MySQL workbench.

   Sample MySQL Workbench:



3. You need to run the database scripts given, to create databases to work with the queries.
4. Query solution screenshots must be pasted in the same Word file.

DDL

i. Create a database with the name MYDBMS and create a 'PERSONS' table under MYDBMS database with the following columns.

PERSON_ID,

FIRSTNAME,

LASTNAME,

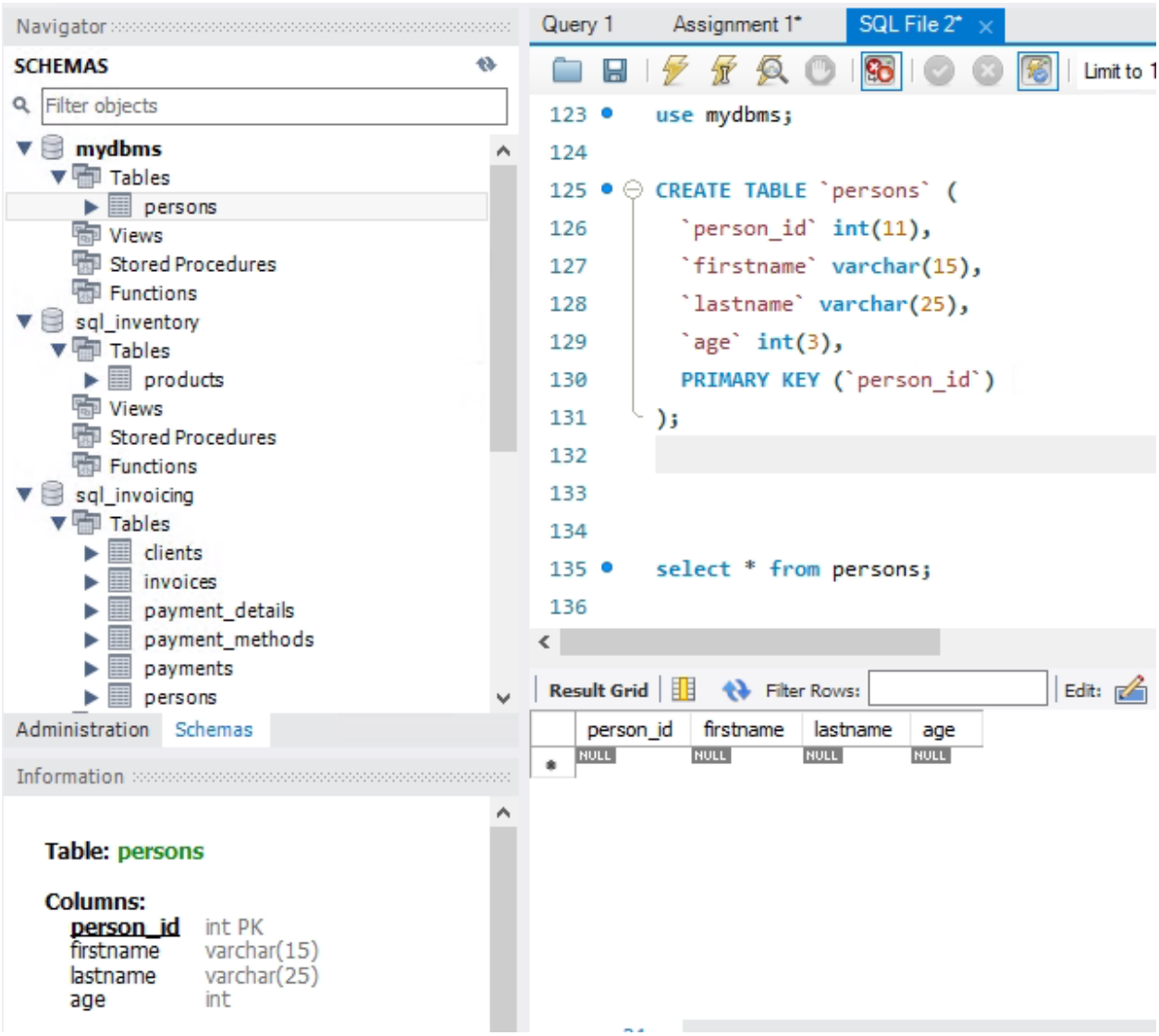
AGE.

Make ‘PERSON_ID’ as primary key and give the appropriate data type to each column.

Data types of the table “persons”:

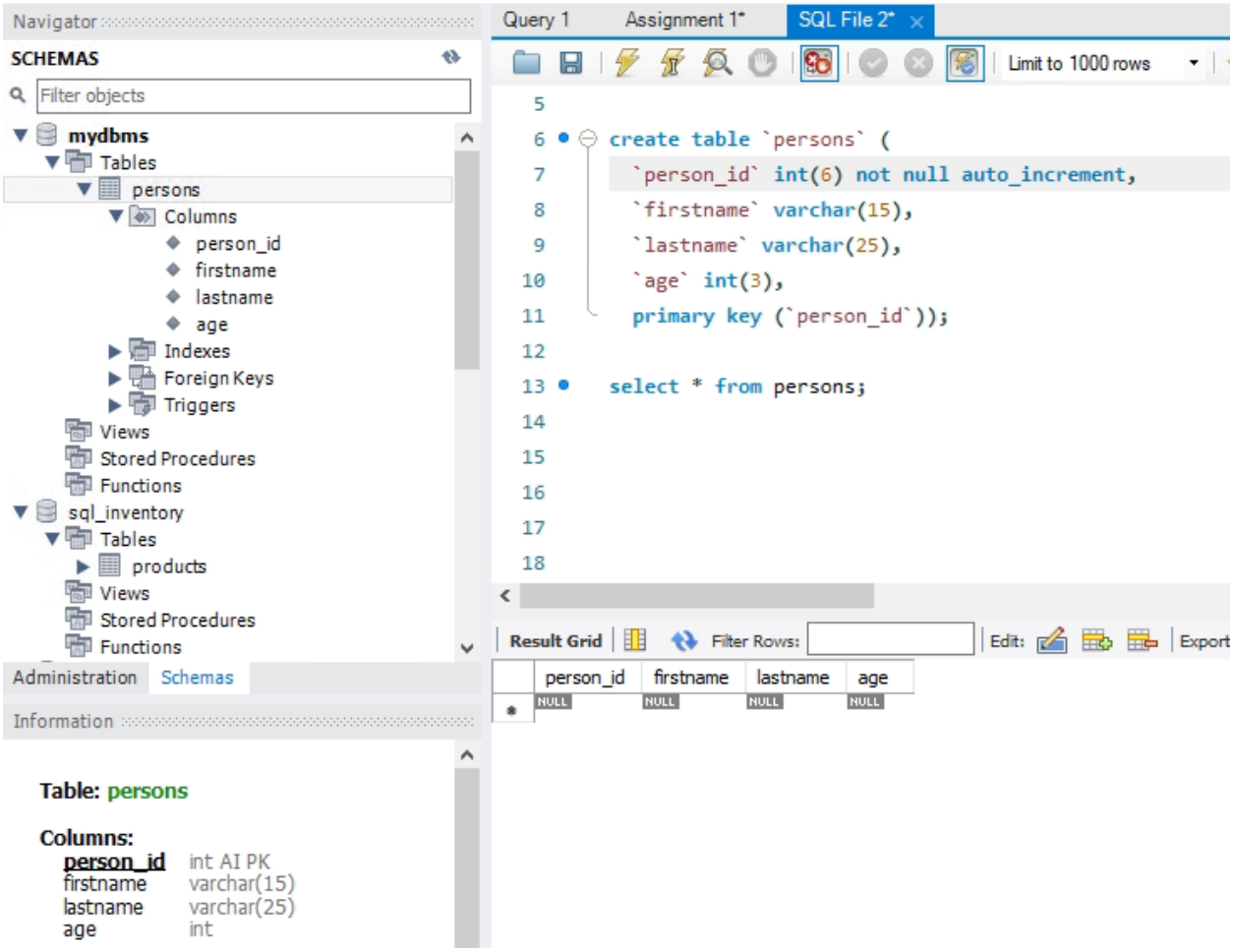
Column Name	Data type	Feature
person_id	Int(11)	The column can take, and show a value exceeding the length 11, but these values will not be prefixed with 0s
firstname	varchar(15)	The column allocates dynamically up to 16 bytes, (character strings) up to 15 for data and, at least, 1 additional byte to store the length of the data.
lastname	varchar(25)	The column allocates dynamically up to 26 bytes, (character strings) up to 25 for data and, at least, 1 additional byte to store the length of the data.
age	Int(3)	The column can take, and show a value exceeding the length 3, but these values will not be prefixed with 0s

A new database called mydbms and a table named "persons" are created with the following query.

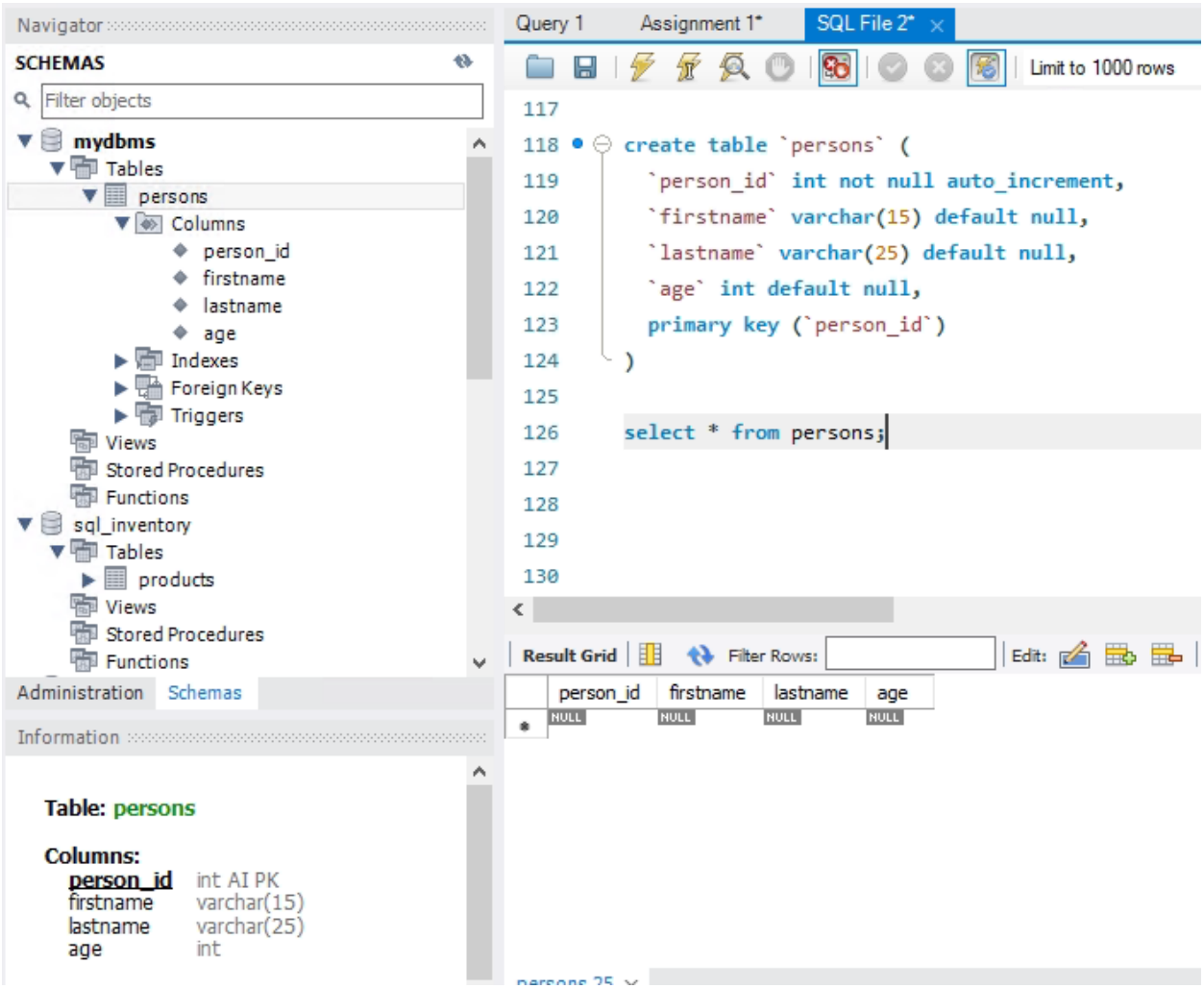


By applying the auto_increment key word in the MySQL query, it is also possible to use the auto increment feature on the person_id column. The person_id does not need to be manually inserted in that scenario. The only drawback is that this auto increment functionality won't help if the user needs to keep his own numbering scheme because the person_id might not always be in sequence mode. However, the user can still insert id numbers as they consider by utilizing the column name (person_id) in the insert sql query.

Furthermore, as per the below sql query, null values are not permitted in the “person_id” column.



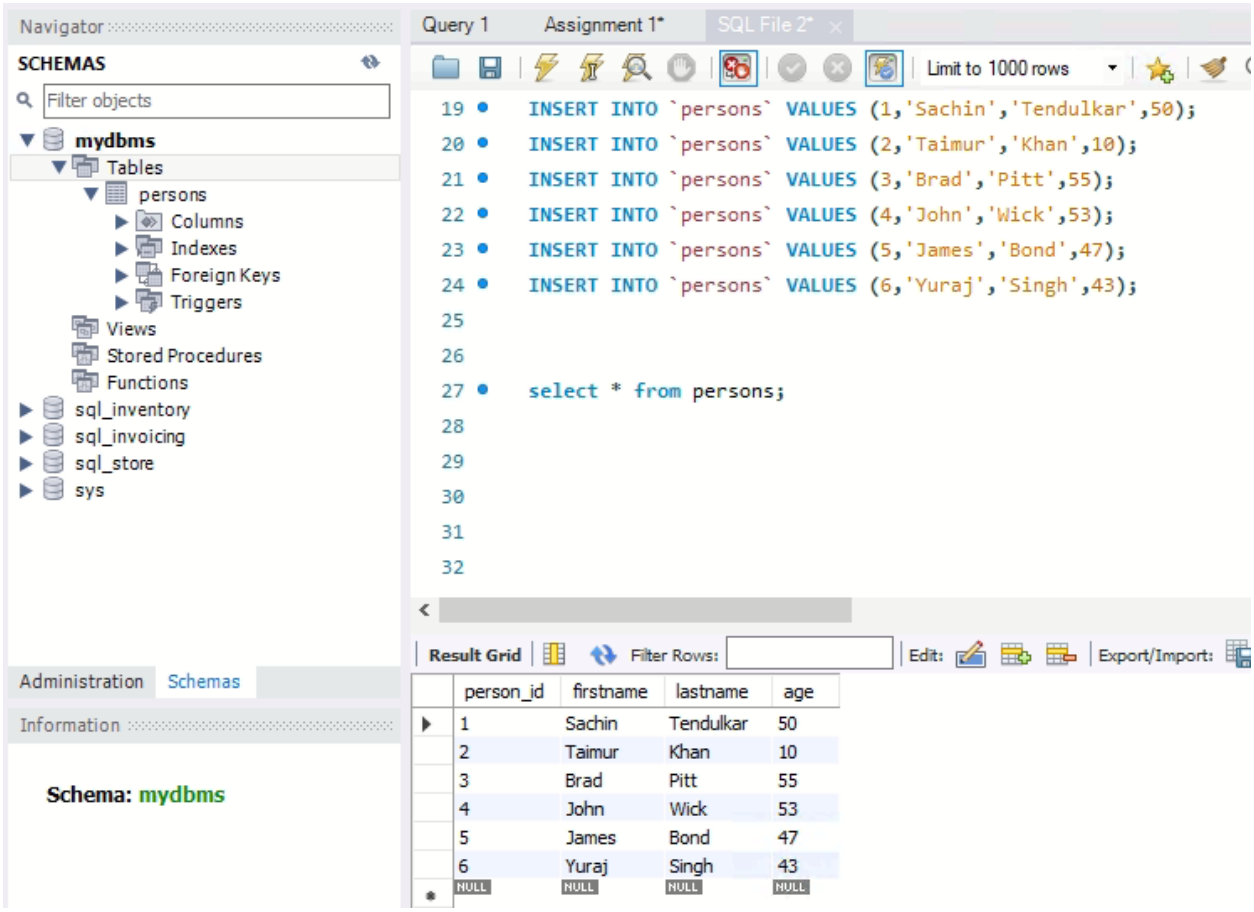
In addition, column constraints can be explicitly defined by using the "default null" key words to indicate that the column can contain null values by default. If not, the MySQL query management system will automatically define it. Therefore, the column constraint must be defined as "not null" when data is required.



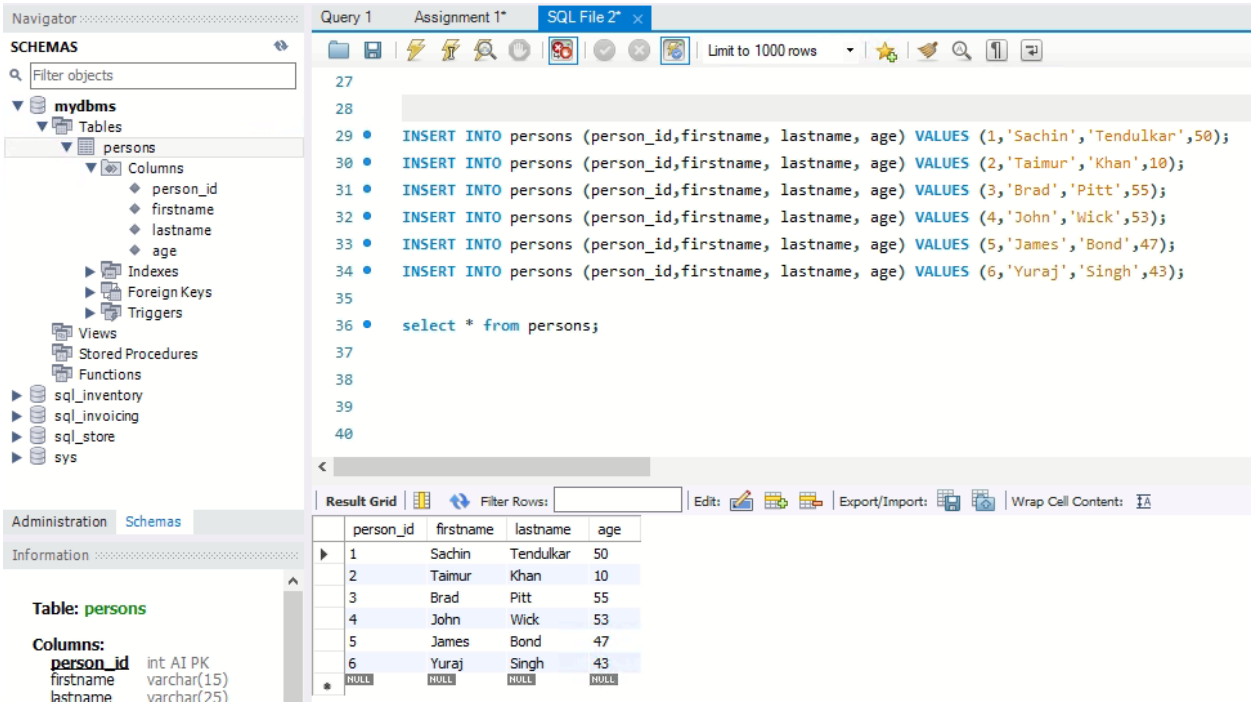
ii. Insert the following data into the 'PERSONS' created table.

```
-- Person_ID    FirstName    LastName    Age
-- 1            Sachin       Tendulkar   50
-- 2            Taimur       Khan        10
-- 3            Brad         Pitt        55
-- 4            John         Wick        53
-- 5            James        Bond        47
-- 6            Yuraj        Singh       43
```

Inserting values without using column names,



Inserting data using column names,



iii. Use the ‘SQL_INVOICING’ database.

Join the ‘payments’ table with the ‘clients’ table using the ‘CLINETS_ID’ column and then join the ‘payments’ table with the ‘PAYMENT_METHODS’ table using the ‘PAYMENT_METHOD’ column. Select the following columns.

PAYMENT_ID from the payments table,

Amount from the payments table,

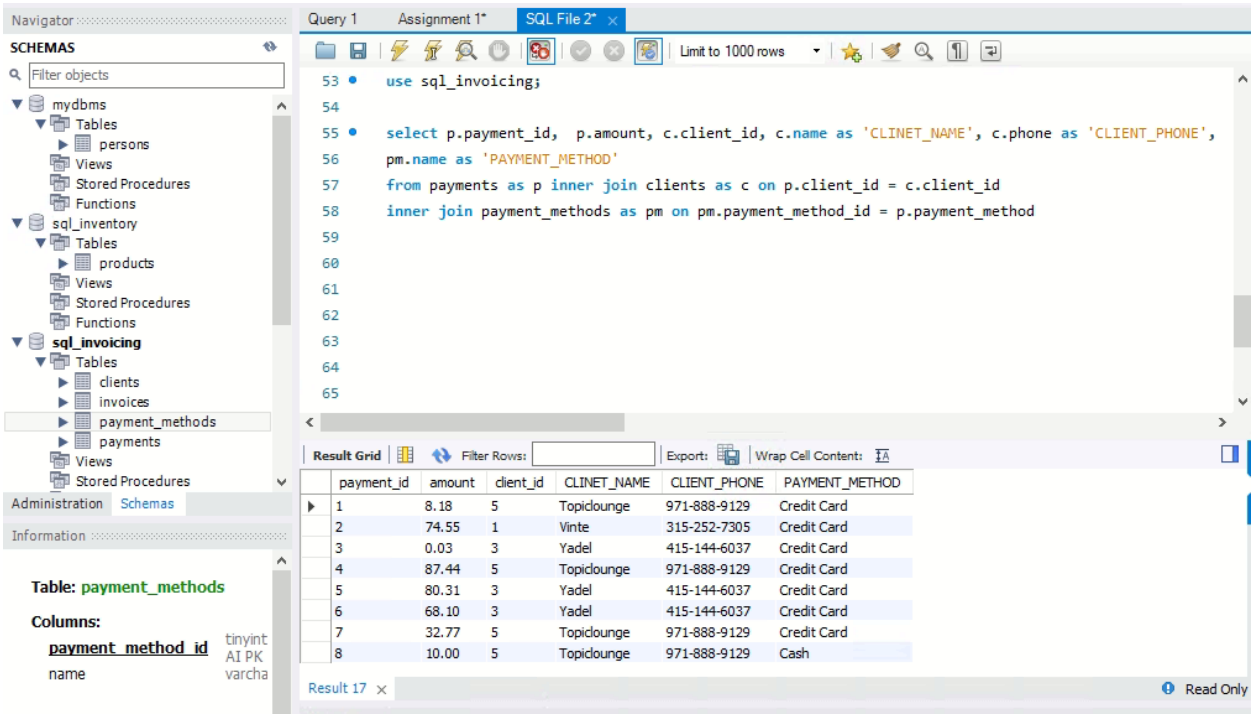
CLIENT_ID from the clients table,

Name as CLINET_NAME from the clients table,

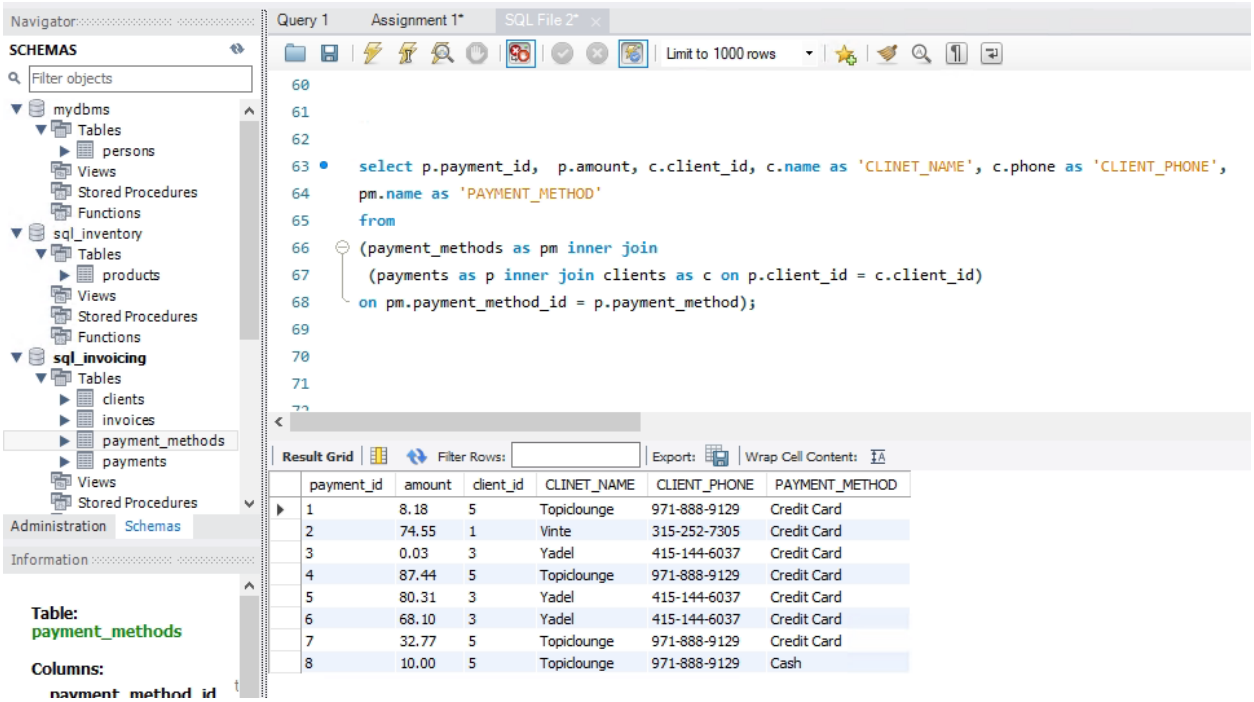
Phone as CLIENT_PHONE from the clients table,

Name as PAYMENT_METHOD from PAYMENT_METHODS table.

The join query,



Another way of joining tables to get the same results,

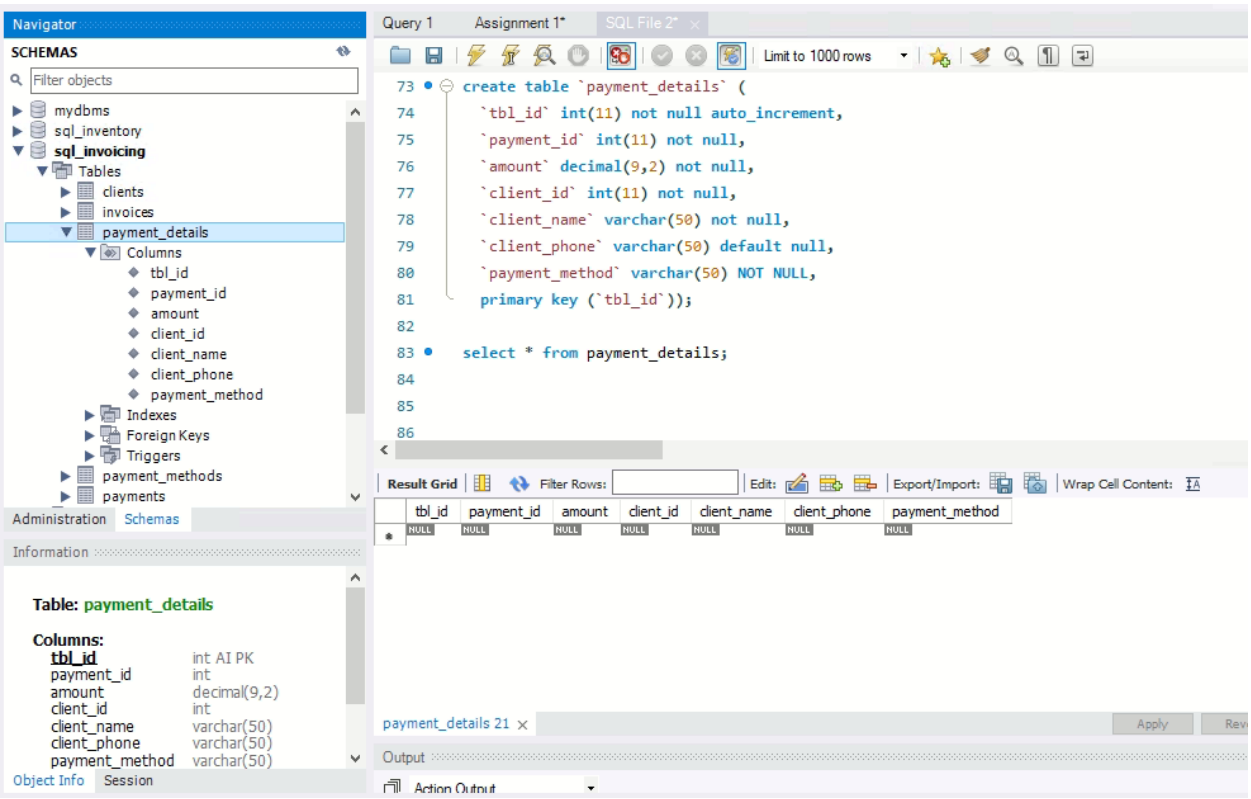


iv. Create a ‘PAYMENT_DETAILS’ table and insert records in it by executing the following join query.

Keeping an auto increment id field in the payment_details table.

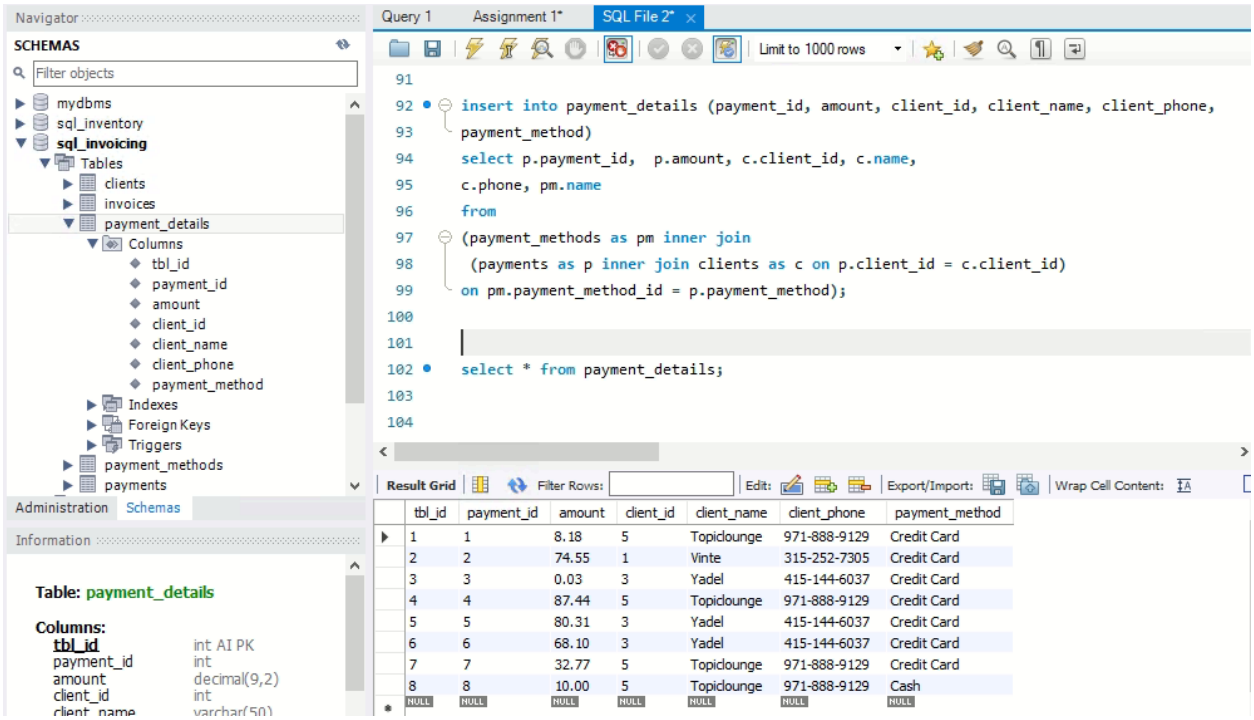
‘tbl_id’ int(11) not null auto_increment

Here, the columns have been chosen with the same data types depending on the tables on which the aforementioned join query was mentioned.



The below query shows the insertion of the outcome of the aforementioned join query into the recently created payment_details table.

Here, the join query's chosen columns have been properly mapped to the columns of the payment_details table.

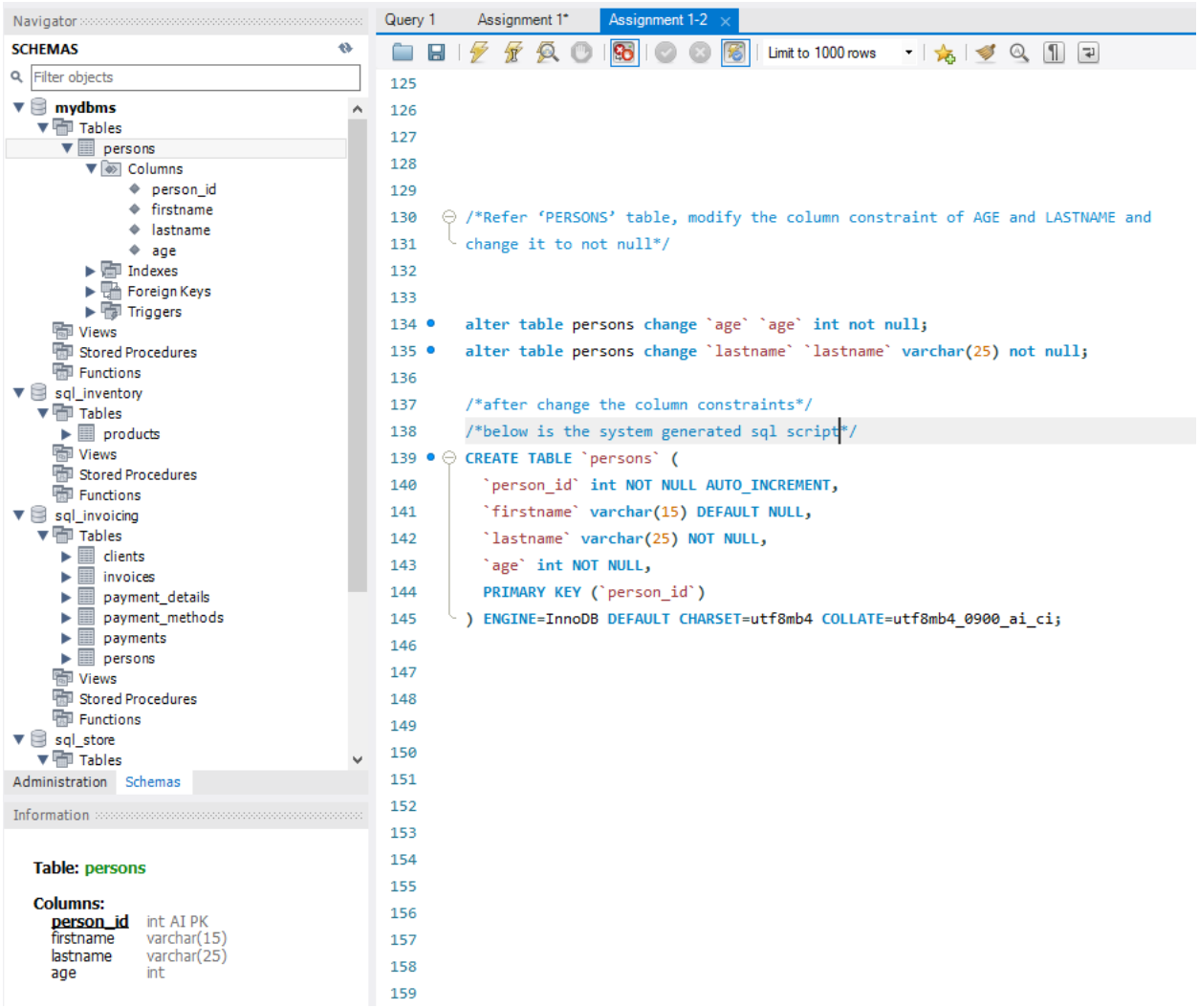


DML

v. Refer 'PERSONS' table, modify the column constraint of AGE and LASTNAME and change it to not null.

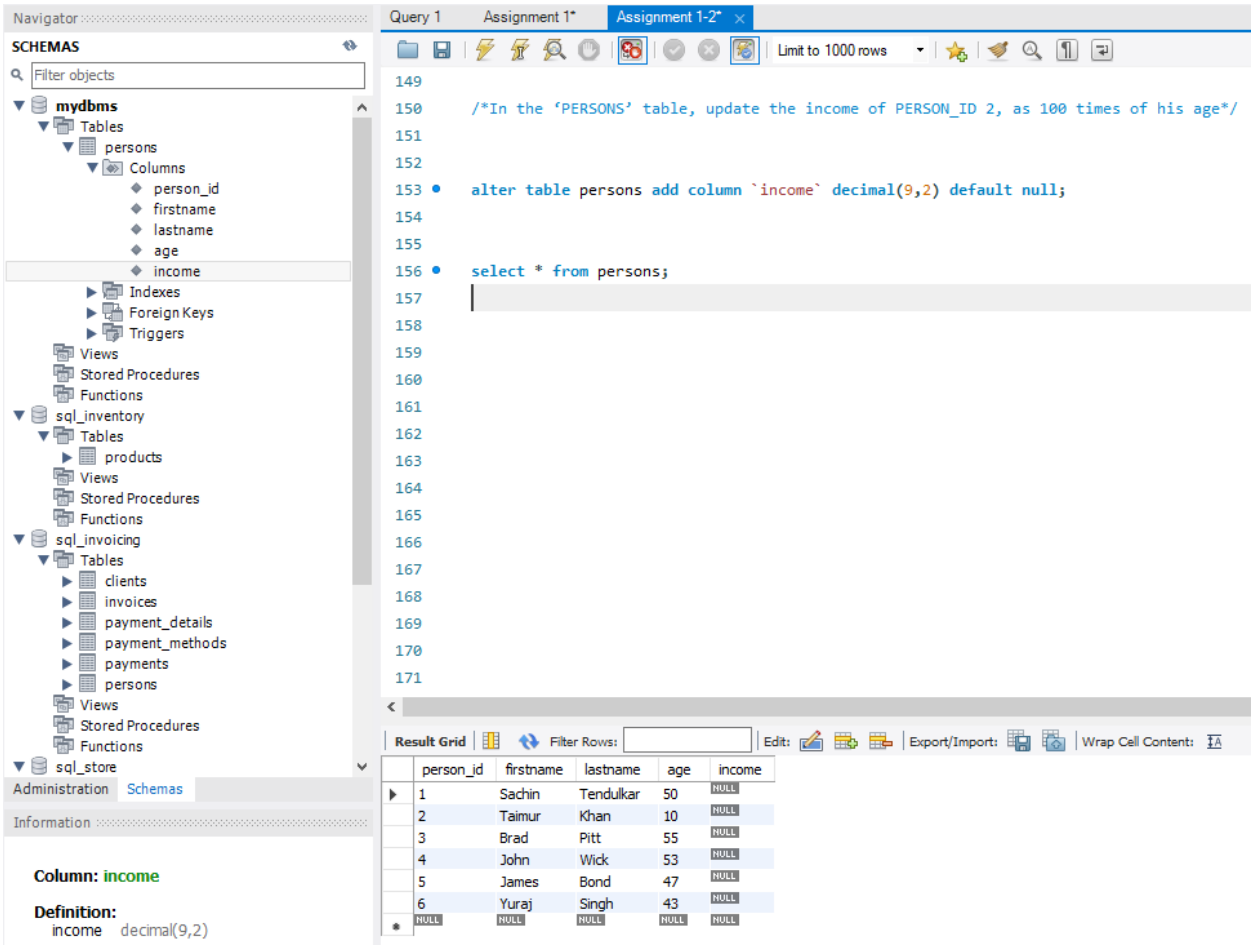
The below syntax can be used to change NULL into NOT NULL constraint in MySQL.
ALTER TABLE table_name MODIFY column_name column_definition;

According to the snapshot below, the constraints for the "lastname" and "age" columns have been modified from NULL to NOT NULL.
It indicates that from now onwards, the user is unable to enter null values in these two columns.

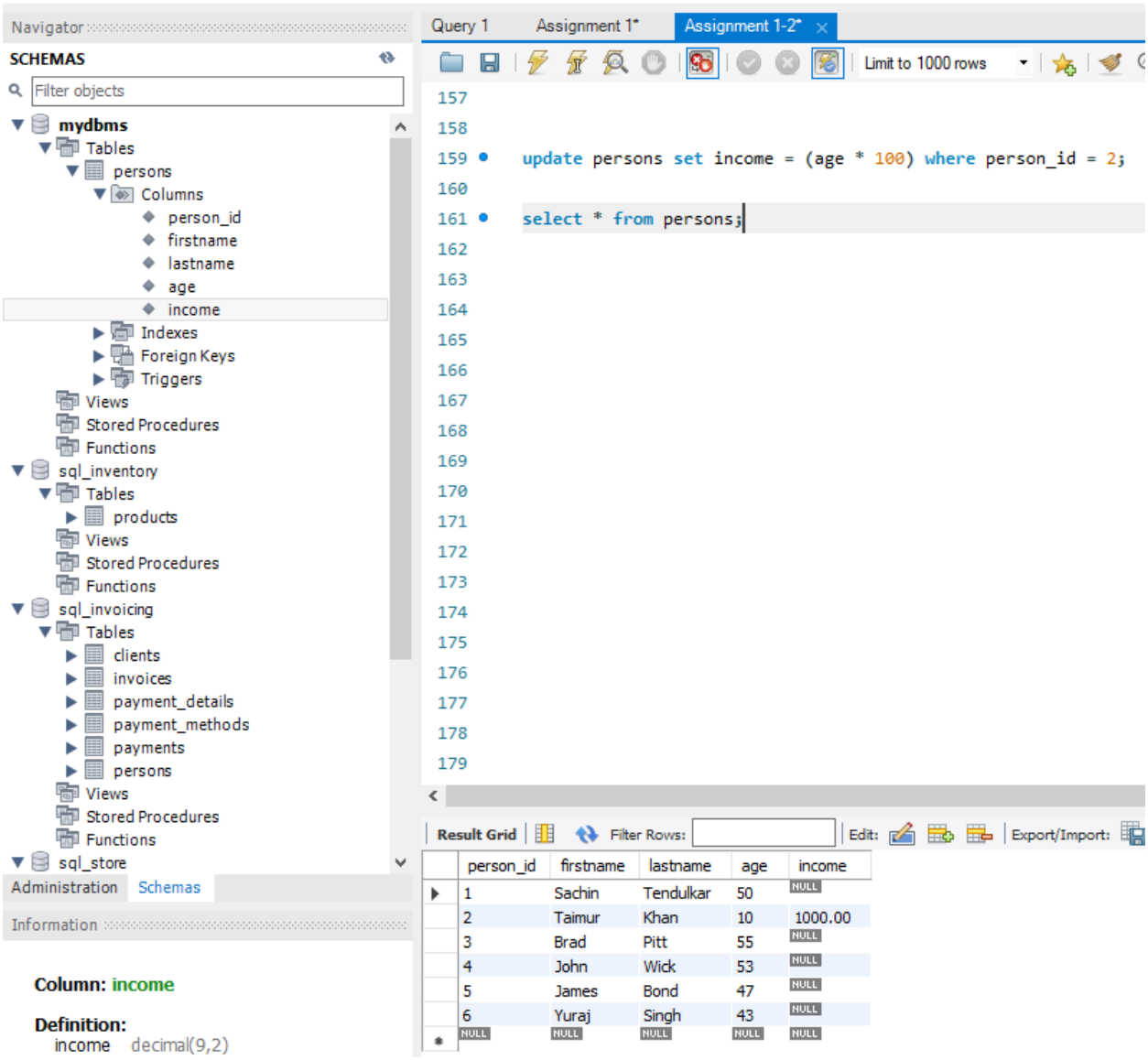


vi. In the 'PERSONS' table, update the income of PERSON_ID 2, as 100 times of his age.

Adding a new column called "income," using decimal data type, since income is usually expressed in monitory terms.

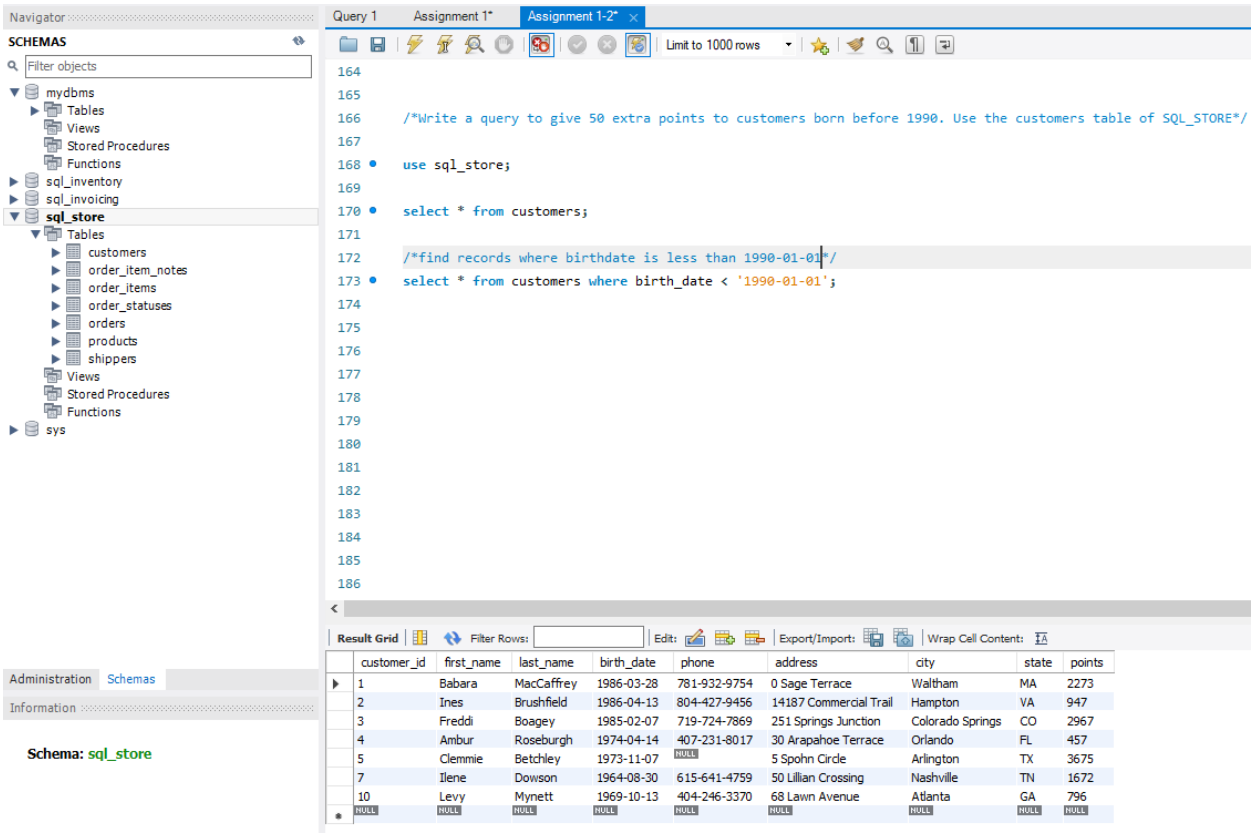


Updating the 'income' column just for the person with ID 2 by multiplying the value of the 'age' column by 100.

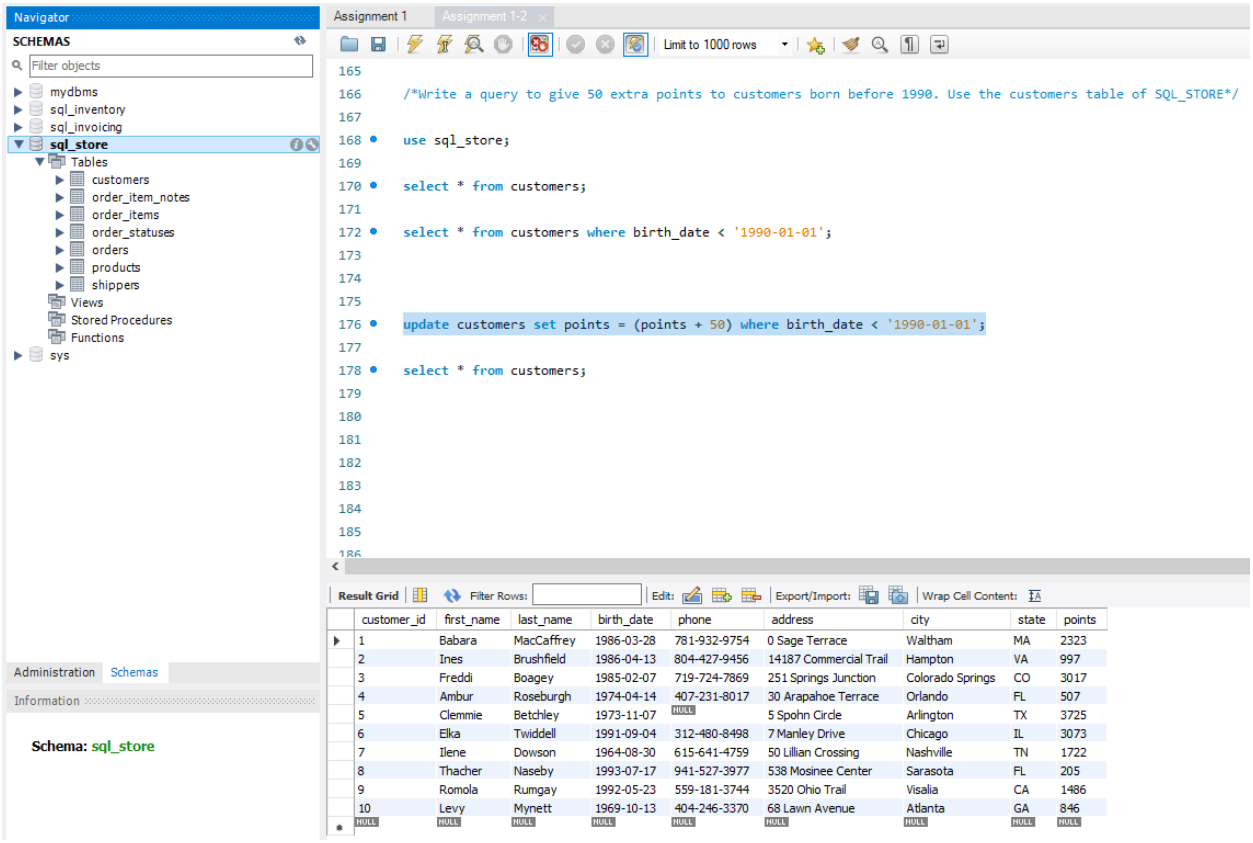


vii. Write a query to give 50 extra points to customers born before 1990. Use the customers table of SQL_STORE.

Identifying customers whose birth dates are less than January 1, 1990



Updating the points column,



References

Abld Ali Awan. (2023, Sep). *What is Named Entity Recognition (NER)? Methods, Use Cases, and Challenges*. Retrieved from datacamp: https://www.datacamp.com/blog/what-is-named-entity-recognition-ner

Akash Bokade. (2023, Aug). *NLP workbook + Amazon Review Sentiment analysis*. Retrieved from kaggle: https://www.kaggle.com/code/akashbokade/nlp-workbook-amazon-review-sentiment-analysis

Chetna Khanna. (2021, Feb 10). *Text pre-processing: Stop words removal using different libraries*. Retrieved from Medium: https://towardsdatascience.com/text-pre-processing-stop-words-removal-using-different-libraries-f20bac19929a

Kevin Arvai. (2020, Jul 20). *K-Means Clustering in Python: A Practical Guide*. Retrieved from realpython: https://realpython.com/k-means-clustering-python/

Kevin Arvai. (n.d.). *K-Means Clustering in Python: A Practical Guide*. Retrieved from realpython: https://realpython.com/k-means-clustering-python/

Martin Porter. (2006, Jan). *The Porter Stemming Algorithm*. Retrieved from tartarus: https://www.tartarus.org/~martin/PorterStemmer/

Neri. (2023, Nov 03). *How To Implement Intent Classification In NLP [7 ML & DL Models] With Python Example*. Retrieved from spotintelligence: https://spotintelligence.com/2023/11/03/intent-classification-nlp/

Prateek Majumder. (2023, Sep 13). *Named Entity Recognition (NER) in Python with Spacy*. Retrieved from analyticsvidhya: https://www.analyticsvidhya.com/blog/2021/06/nlp-application-named-entity-recognition-ner-in-python-with-spacy/

Sadrach Pierre & Matthew Urwin. (2017, Oct 14). *builtin*. Retrieved from How to Form Clusters in Python: Data Clustering Methods: https://builtin.com/data-science/data-clustering-python

What is named entity recognition? (n.d.). Retrieved from ibm: https://www.ibm.com/topics/named-entity-recognition