**Basic SQL Commands:**

DDL , DML  
Basic Operations using various DML.  
Limiting Results  
Sorting Results  
Skipping Results  
Case statements  
Aggregation  
Constraints  
Index and its uses.  
Alias  
GROUP BY and Having Clause

**DDL: Data Definition Language:**

CREATE, ALTER, DROP, TRUNCATE

**CREATE Table:**

MYSQL> create database bank;

MYSQL> use bank;

MYSQL> create table ACCOUNTS (acc\_number int NOT NULL, cust\_name varchar(25), age int, acc\_balance int, acc\_type varchar(20), branch varchar(20), acc\_startdate date);

0 row(s) affected

0.157 sec

**ALTER Table:**

MYSQL> alter table ACCOUNTS ADD email varchar(50);

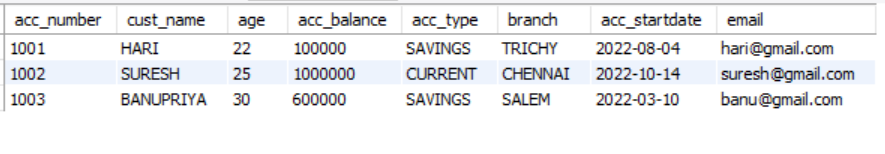
0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0

MYSQL> alter table ACCOUNTS rename to ACCOUNT\_DETAILS;

0 row(s) affected

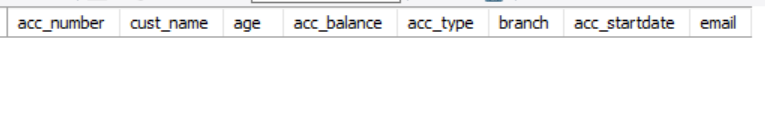
**TRUNCATE:**

Before Truncate:



MYSQL> truncate ACCOUNTS;

After Truncate:



**DROP:**

MYSQL> drop table ACCOUNTS;

The table rows as well as the schema of the table is totally dropped and the table is deleted.

MYSQL> select \* from ACCOUNTS;

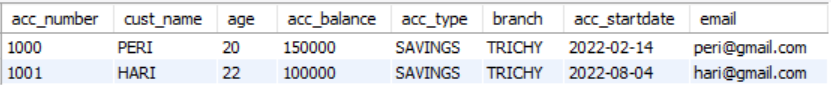
Error Code: 1146. Table 'bank.accounts' doesn't exist

**DML: Data Manipulation Language**

INSERT, UPDATE, DELETE

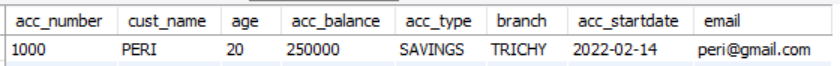
**INSERT into Table:**

MYSQL> insert into ACCOUNTS values (1001, 'HARI', 22, 100000, 'SAVINGS', 'TRICHY', '2022-08-04','hari@gmail.com');



**UPDATE Row:**

MYSQL> update ACCOUNTS set acc\_balance = acc\_balance + 100000 where acc\_number = 1000;

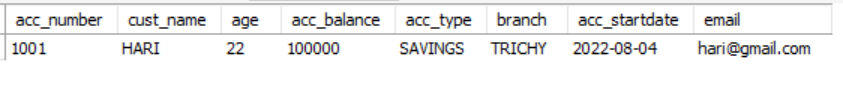


**DELETE Row:**

MYSQL> DELETE from ACCOUNTS where cust\_name = 'Peri';

Row Deleted

1 row(s) affected



**Basic Operations using various DML:**

**INSERT Commands:**

MYSQL> insert into ACCOUNTS values (1000, 'PERI', 20, 150000, 'SAVINGS', 'TRICHY', '2022-02-14','peri@gmail.com')

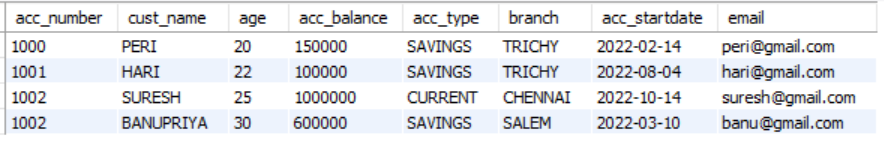
MYSQL> insert into ACCOUNTS values (1001, 'HARI', 22, 100000, 'SAVINGS', 'TRICHY', '2022-08-04','hari@gmail.com')

MYSQL> insert into ACCOUNTS values (1002, 'SURESH', 25, 1000000, 'CURRENT', 'CHENNAI', '2022-10-14','suresh@gmail.com')

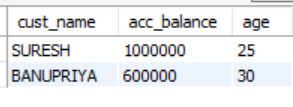
MYSQL> insert into ACCOUNTS values (1003, 'BANUPRIYA', 30, 600000, 'SAVINGS', 'SALEM', '2022-03-10','banu@gmail.com');

**SELECT Statement**:

MYSQL> select \* from ACCOUNTS;



MYSQL> select cust\_name, acc\_balance, age from accounts where age >= 25;



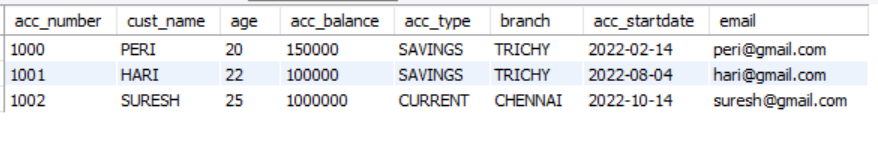
**LIMITING RESULTS:**

Limits and fetches only top 3 rows. All returns the same output.

MYSQL> SELECT \* FROM ACCOUNTS LIMIT 3;

MYSQL> SELECT \* FROM ACCOUNTS FETCH FIRST 3 ROWS ONLY;

MYSQL> SELECT TOP 3 \* FROM ACCOUNTS;

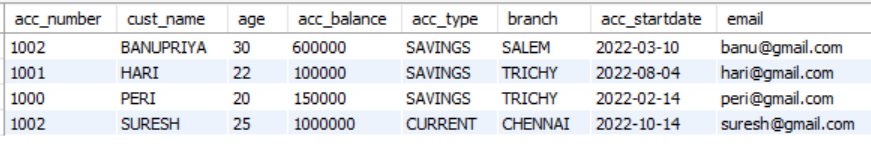


**SORTING Results:**

**ORDERBY:**

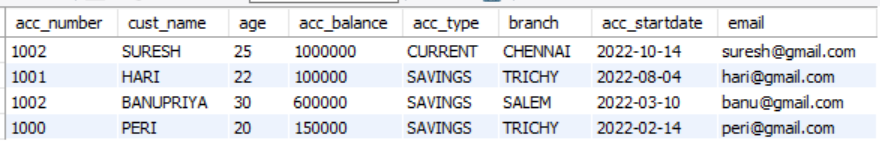
We sort the rows with specific row using ORDERBY Clause.

MYSQL> SELECT \* FROM ACCOUNTS ORDER BY cust\_name;



MYSQL> SELECT \* FROM ACCOUNTS ORDER BY acc\_startdate DESC;

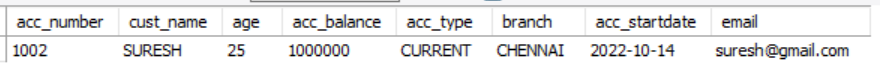
Order by in descending order.



**SKIPPING RESULTS:**

MYSQL> SELECT \* FROM ACCOUNTS LIMIT 2, 1; -- offset, limit

This SQL query Skips first 2 rows and fetches one row after that. (i.e, 3rd Row)



**CASE Statements:**

Case statement is similar to an If-Else Statement

MYSQL> SELECT acc\_number, cust\_name,

CASE

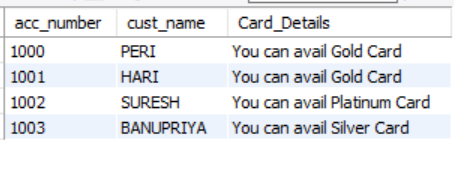
WHEN acc\_balance >= 200000 THEN 'You can avail Platinum Card'

WHEN acc\_balance >= 100000 and acc\_balance < 200000 THEN 'You can avail Gold Card'

ELSE 'You can avail Silver Card'

END AS Card\_Details

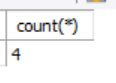
FROM ACCOUNTS;



**AGGREGATION:**

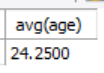
**COUNT:**

MYSQL> select count(\*) from ACCOUNTS;



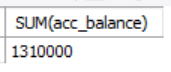
**AVERAGE:**

MYSQL> select avg(age) from ACCOUNTS;



**SUM:**

MYSQL> select SUM(acc\_balance) from ACCOUNTS;



**CONSTRAINTS:**

MYSQL> ALTER TABLE ACCOUNTS

MODIFY acc\_number int NOT NULL UNIQUE;

MYSQL> ALTER TABLE ACCOUNTS

MODIFY acc\_number int PRIMARY KEY;

**INDEXES and its USES:**

It is a schema object used by Oracle server to speed up the retrieval of rows by using a pointer. It is independent of the table it indexes. It can reduce disk I/O by using a rapid path access method to locate data quickly.

MYSQL> create index index\_bank on ACCOUNTS (cust\_name,branch);

Index is created.

MYSQL> show index from ACCOUNTS;



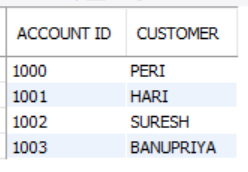
MYSQL> alter table ACCOUNTS drop index index\_bank;

Index is dropped.

**ALIAS Statements:**

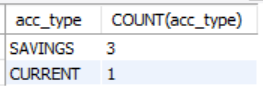
MYSQL> select acc\_number AS 'ACCOUNT ID', cust\_name AS 'CUSTOMER' from ACCOUNTS;

Gives an alias name for the column to display.



**GROUP BY and HAVING Clause:**

MYSQL> select acc\_type,COUNT(acc\_type) from ACCOUNTS GROUP BY acc\_type;



(With having clause):

MYSQL> select acc\_type,COUNT(acc\_type) from ACCOUNTS GROUP BY acc\_type HAVING count(acc\_type) > 1 ;

