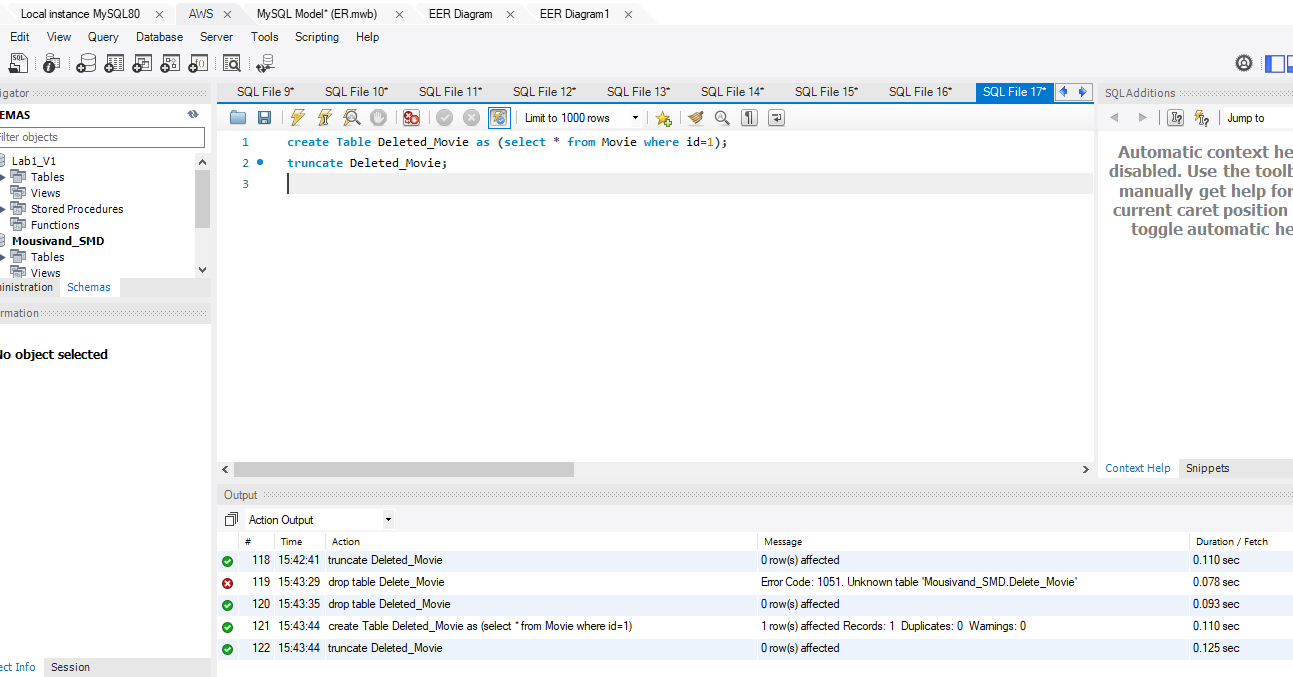
SQL Codes

**1, 2** : This is a fast way to create table for movie logging. The first query table creates the table and the second one clears the table.

**Codes:**

create Table Deleted\_Movie as (select \* from Movie where id=1);

truncate Deleted\_Movie;



**3**:This query is to make sure that no user can have more than 5 accounts and adds a constraint

**Code**:

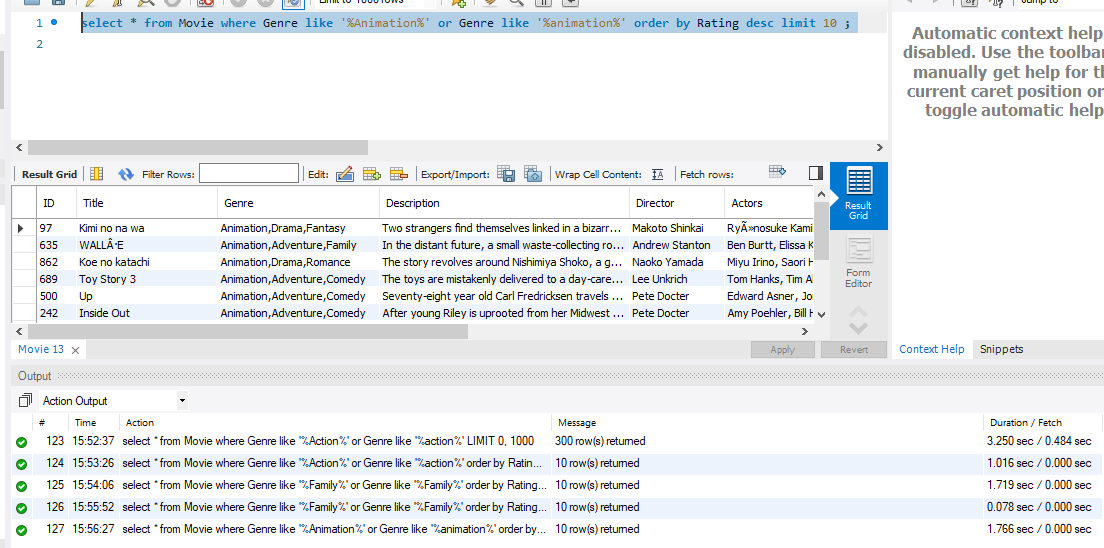
ALTER TABLE Account

ADD CHECK (Account\_Number<=5);



**4**: This query gives everything about 10 animation movies with highest Rating in the descending order

**Code:** select \* from Movie where Genre like '%Animation%' or Genre like '%animation%' order by Rating desc limit 10 ;



**5**: Here by “set autocommit” , “commit” and “rollback”, transaction is created to have either user added and account number one dedicated to him , or neither the user is added nor the first account is created for him .

**Code:**

set autocommit=0;

insert ignore into User values ('Roberto\_Baggio18' , 'Roberto', 'baggio', '3435112','18, Rome St, Milan, Italy');

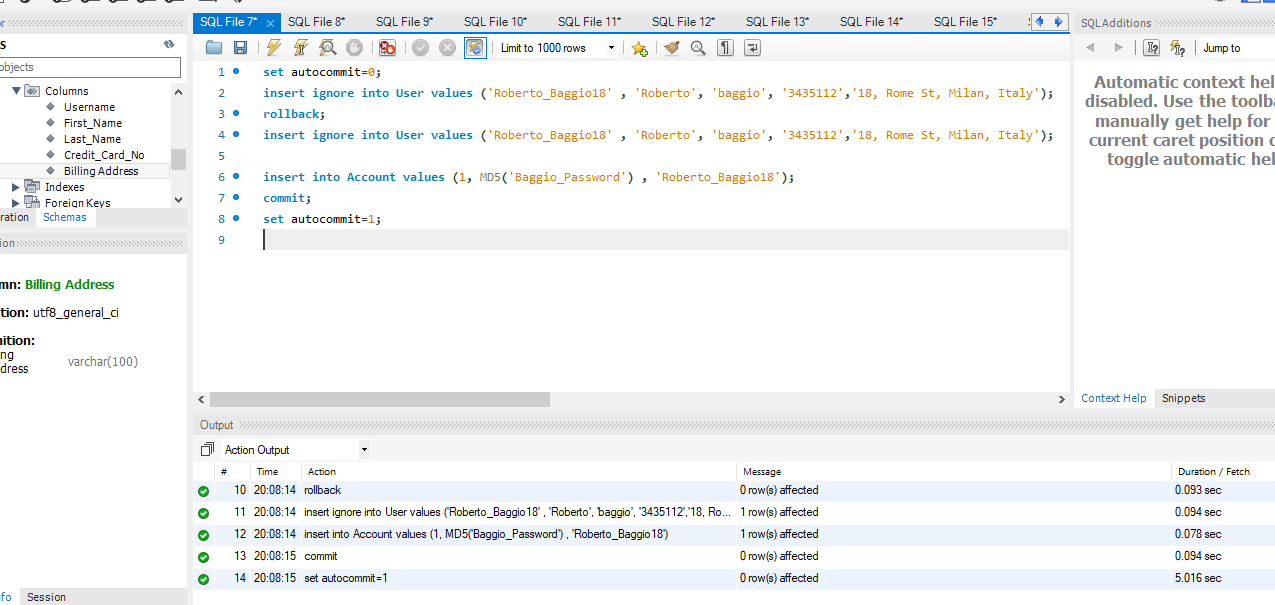
rollback;

insert ignore into User values ('Roberto\_Baggio18' , 'Roberto', 'baggio', '3435112','18, Rome St, Milan, Italy');

insert into Account values (1, MD5('Baggio\_Password') , 'Roberto\_Baggio18');

commit;

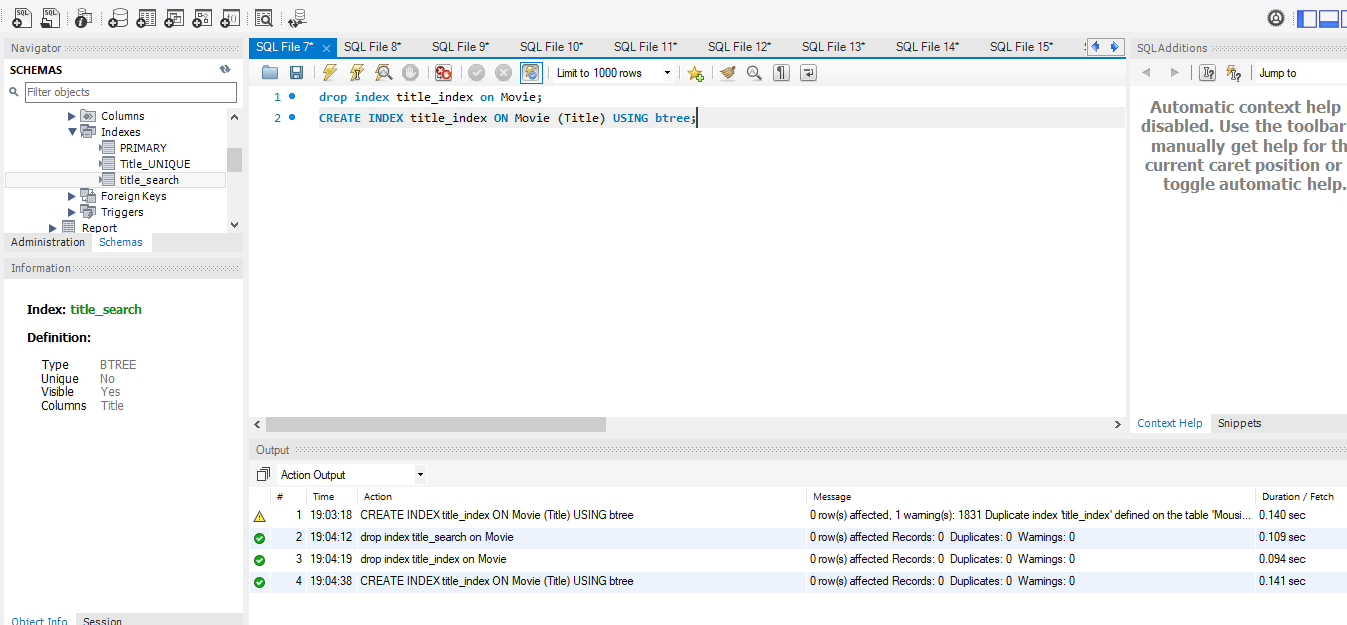
set autocommit=1;



**6**: This Query creates an index to improve speed searching for a movie based on its title using Btree data structure and algorithms. First it removes the created index, the defined it again with adding Btree as the algorithm.

drop index title\_index on Movie;

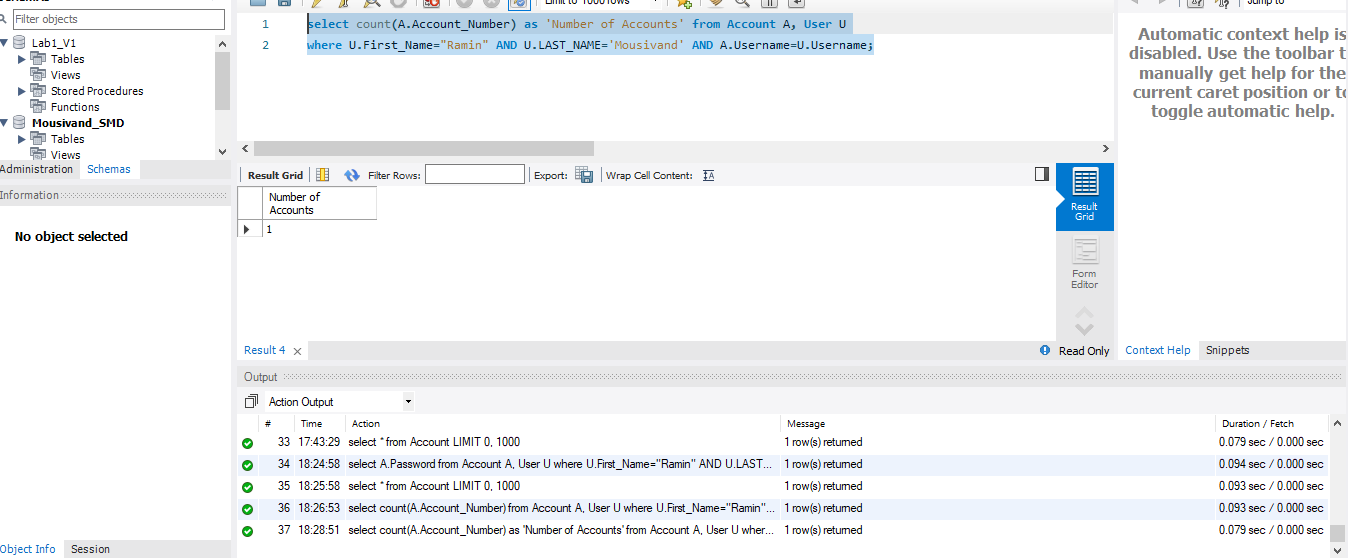
CREATE INDEX title\_index ON Movie (Title) USING btree;



**7**: This query shows the number of accounts associated with User (Ramin Mousivand).

**Code**: select count(A.Account\_Number) as 'Number of Accounts' from Account A, User U

where U.First\_Name="Ramin" AND U.LAST\_NAME='Mousivand' AND A.Username=U.Username;



**8:** This query creates a view that holds username and credit cards of users in California

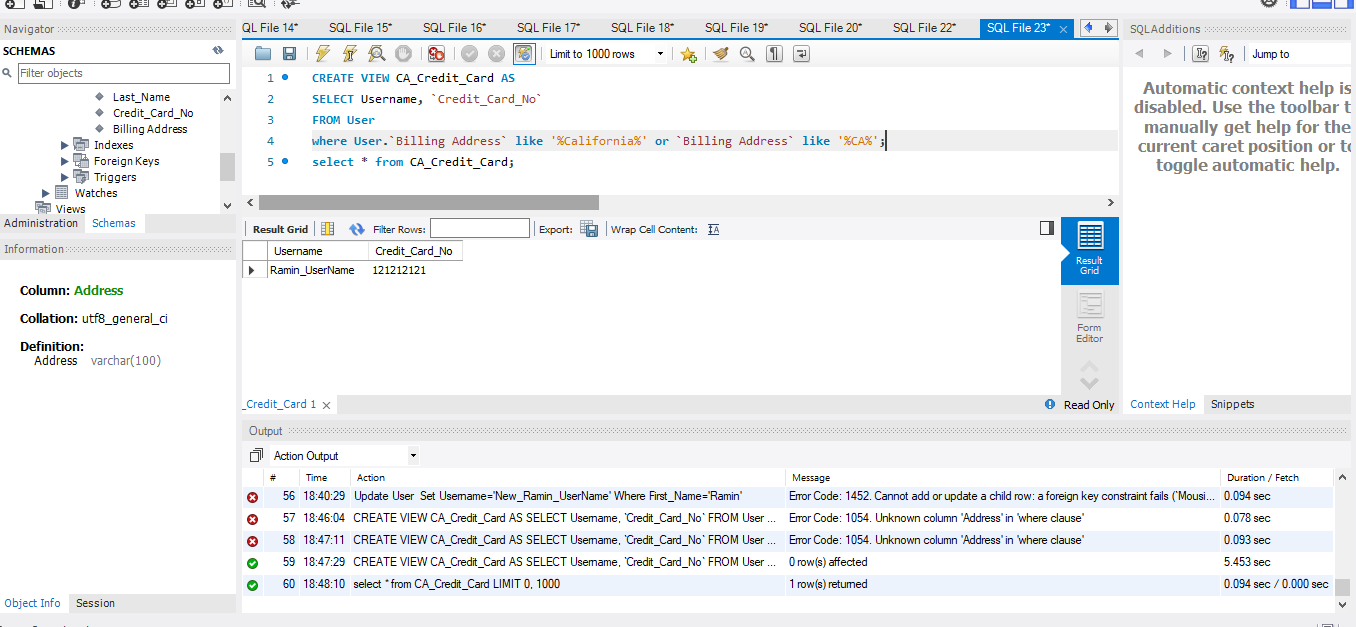
CREATE VIEW CA\_Credit\_Card AS

SELECT Username, `Credit\_Card\_No`

FROM User

where User.`Billing Address` like '%California%' or `Billing Address` like '%CA%';

select \* from CA\_Credit\_Card;



**9**: This query is used to update User (Ramin Mousivand)’s billing address.

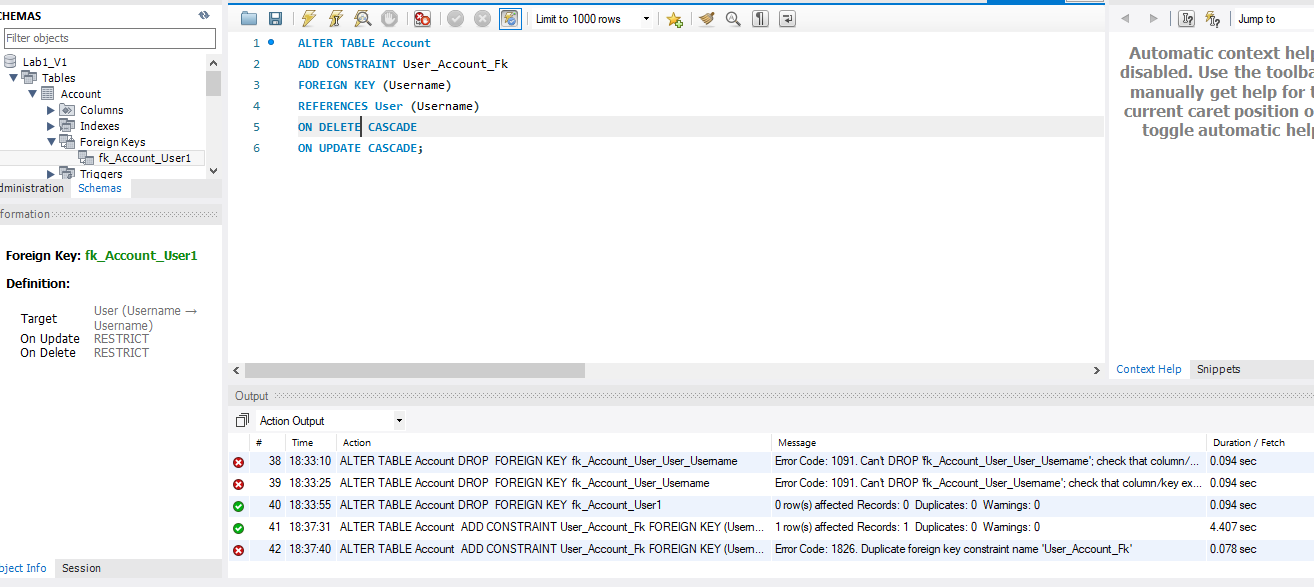
update User

set `Billing Address`='232 Sunset St, San Francicso, CA, US'

where Username='Ramin\_Username';

select \* from User

where Username="Ramin\_Username";



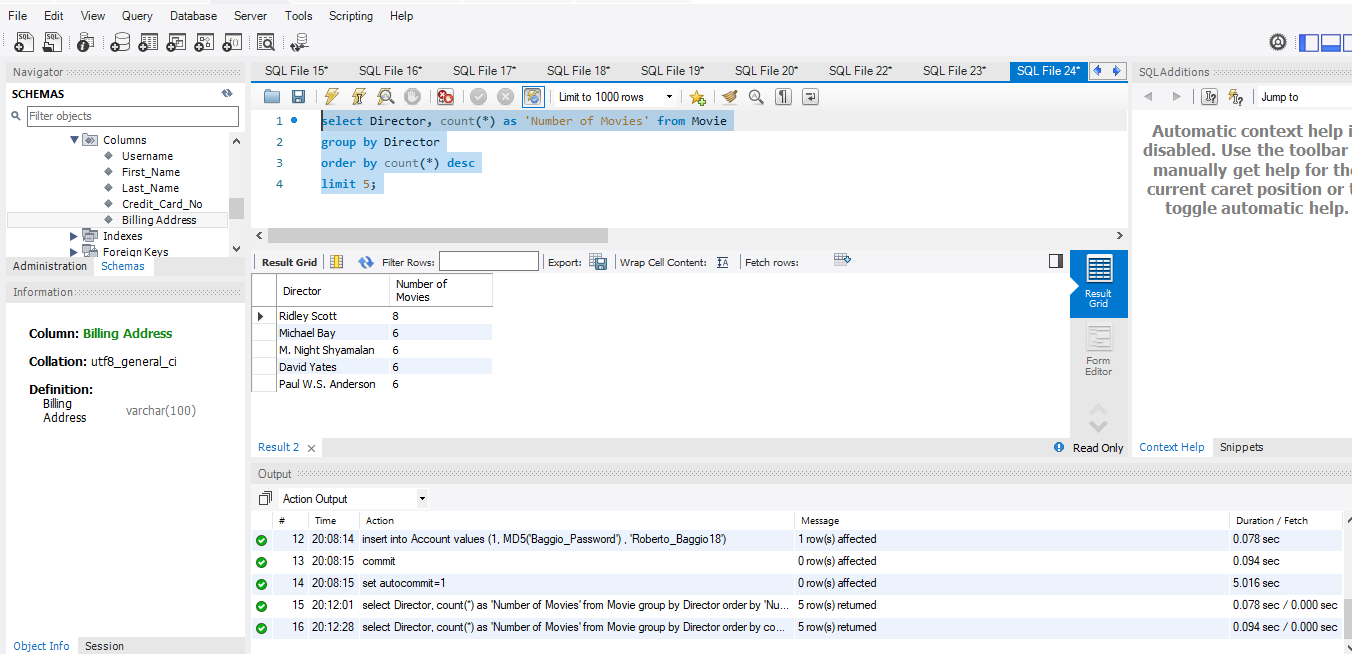
**10**: Find 5 directors with most number of works.

**Code:** select Director, count(\*) as 'Number of Movies' from Movie

group by Director

order by count(\*) desc

limit 5;



**11:** This is to get info regarding the table Deleted\_Credit\_card.

**Code:** describe table Deleted\_Credit\_card;

