TERM PROJECT

CS669 Summer Semester 2016

Revision: 1.1

Author: Sylvester Amoah Addae

Date: August 2, 2016

The business rules

Customer 🡪 Membership Plan 1:1

Each customer signs up for one membership. 1: 1

Only one Membership Plan for each customer. 1:1

Customer 🡪 Queue 1:M

Each customer may have zero or many movies (up to 10 movies) in a Queue of 1: M

A Queue is created by one customer 1:1

Rent\_Record 🡪 Customer 1: M

A customer may have one or more Rent\_Record 1:M

A Rent\_Record belongs to a customer 1:1

Customer 🡪 Invoice 1: M

Each customer may receive one or more Invoices 1:M

An Invoice for one customer 1:1

Movies🡪 Queue 1:M

Queue receives orders of one or many Movies 1: M

A Movie is ordered into one Queue. 1: 1

Mail 🡪 Queue 1: M

A Mail contain one or Many Movies in the ordered queue 1: M

Next movie in Queue is contained in a Mail 1:1

Customer 🡪 Mail 1: M

A Customer receive one or many mails 1: M

A mail is sent to a customer 1: 1

Genre 🡪 Movie 1:M

Each Movie belong to one Genre 1:1

A Genre has many movie 1: M

Director 🡪 Movies 1: M

Each Movie is directed by one Directors 1: 1

A Director may direct one or many Movies 1:M

Customer 🡪 Account 1:1

Each Customer has one Bill\_payment\_Acc. 1:1

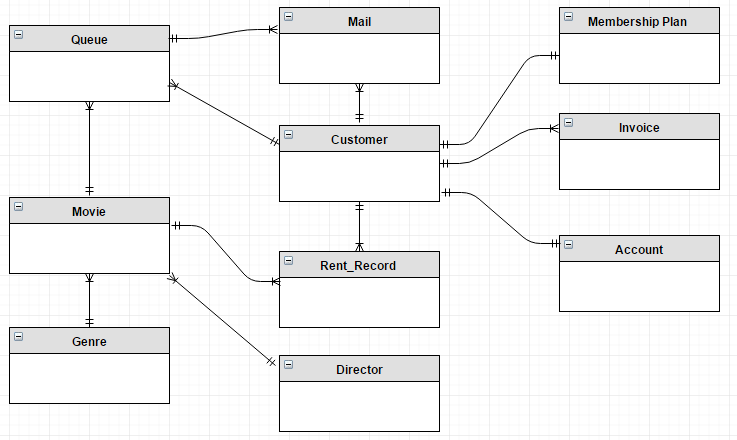
Bill\_payment\_Acc for one Customer 1:1

Movie 🡪 Rent\_Record 1:M

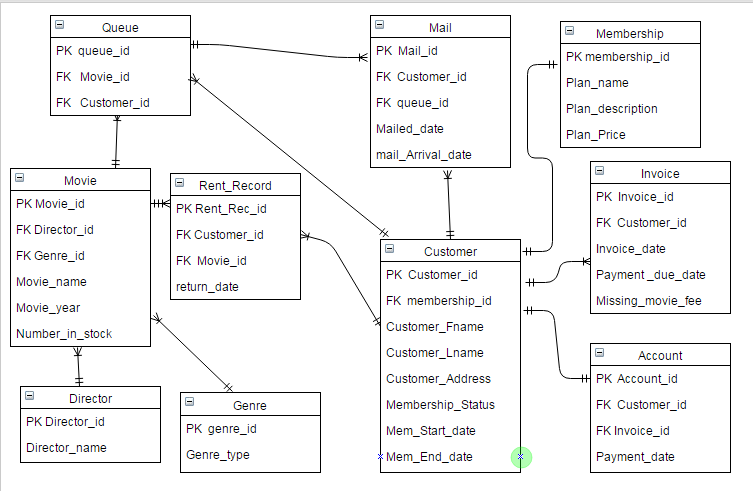
A movie may have many Rent\_Record. 1:M

Rent\_Record may have a Movie 1: 1

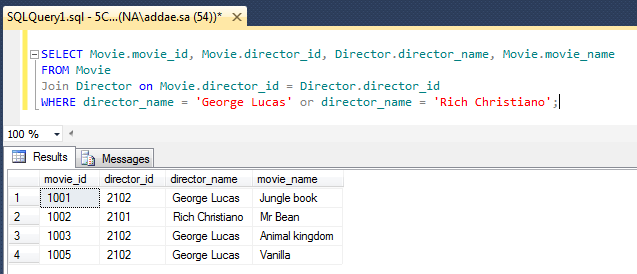
Conceptual ERD or EERD



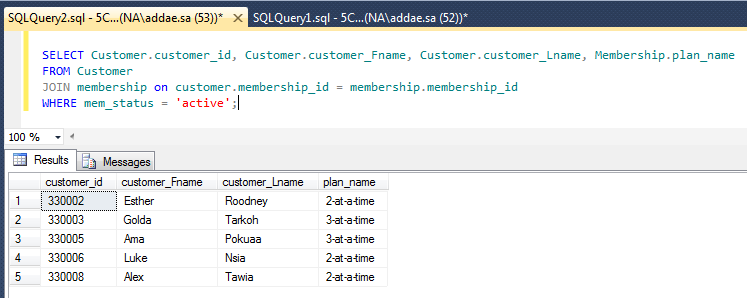
Logical ERD:



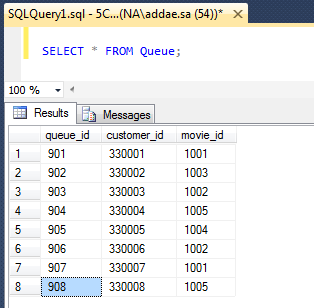
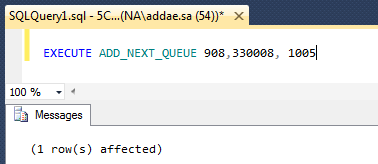
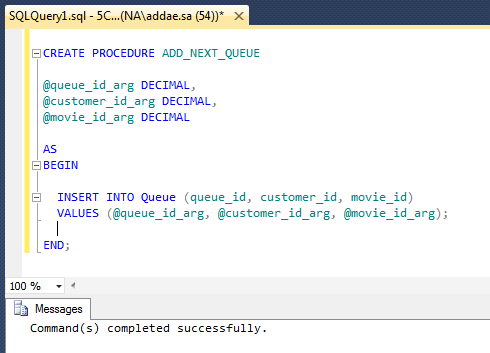
USE CASE 1



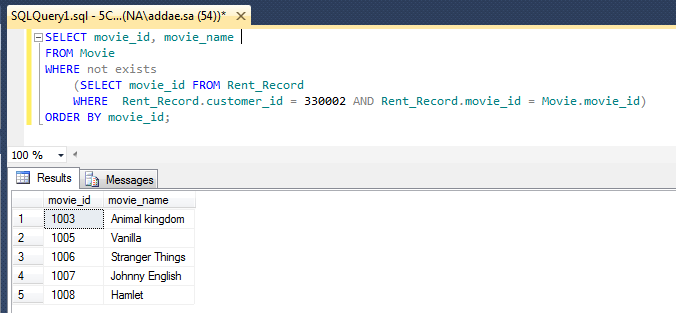
USE CASE 2:



USE CASE 3:

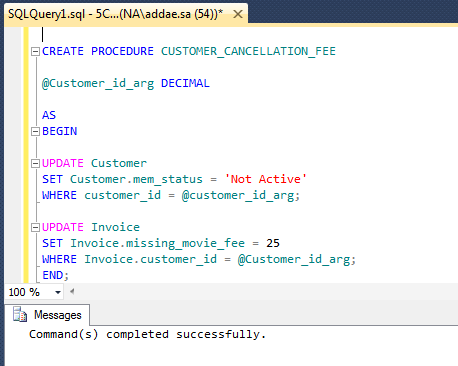


USE CASE 4:

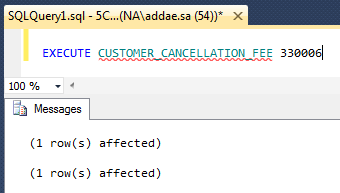


USE CASE 5:

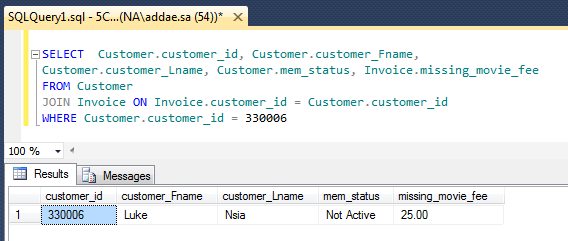
Stored Procedure



Execution/ Invocation

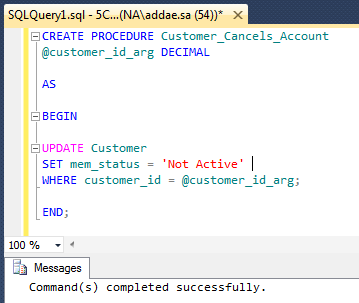


Verification

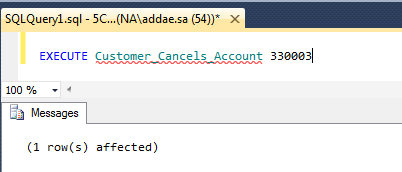


USE CASE 6:

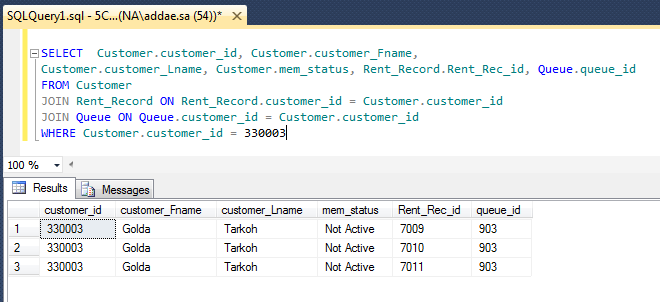
Stored Procedure Creation:



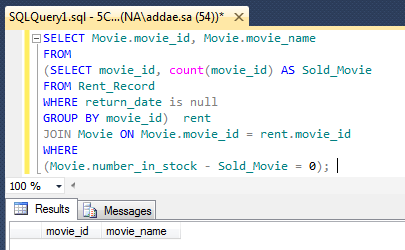
Executing the stored Procedure



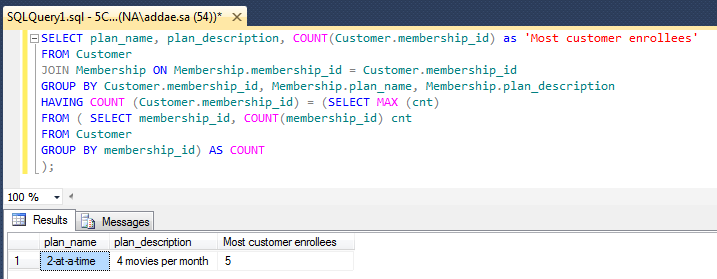
Verification of results



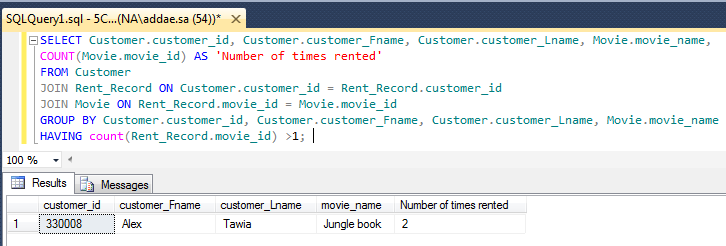
USE CASE 7.



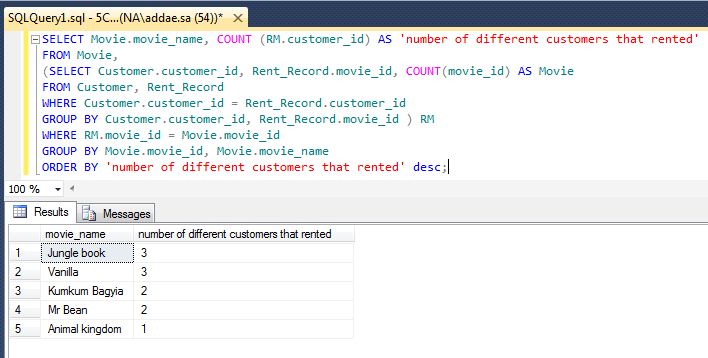
USE CASE 8



USE CASE 9.

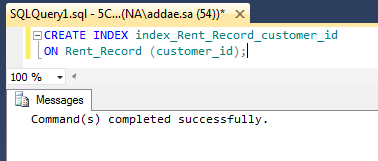


USE CASE 10



Index Creation

Index 1

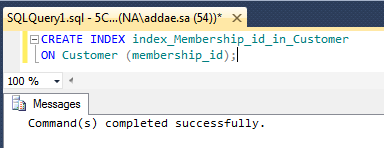


The customer\_id is a foreign key and it does carry over its primary key unique index status from Customer table so I will create non-unique index on it in the Rent\_Record table. This Index allows database application to find data fast; without reading the whole table. It will help speed up search from the database table. Rent\_Record.customer\_id is usually used in the WHERE and ORDER\_BY clauses;

Eg. SELECT movie\_id FROM Rent\_Record WHERE Rent\_Record.customer\_id = 330002 AND Rent\_Record.movie\_id = Movie.movie\_id

Eg2. WHERE Customer.customer\_id = Rent\_Record.customer\_id

Index 2



The membership\_id is a foreign in Customer table so I will create non-unique index on it in the Customer table. It will help speed up search from the database table

This Index allows database application to find data fast; without reading the whole table.

The membership\_id in Customer table is used in WHERE CLAUSES and ORDER BY clauses so creating index on it will improve efficieny.

Eg. GROUP BY Customer.membership\_id, Membership.plan\_name, Membership.plan\_description

HAVING COUNT (Customer.membership\_id) = (SELECT MAX (cnt)

FROM ( SELECT membership\_id, COUNT(membership\_id) cnt

FROM Customer

GROUP BY membership\_id) AS COUNT