**Project Scripts**

**Create Database:**

CREATE TABLE customer(

customer\_id DECIMAL(10) NOT NULL,

customer\_fname VARCHAR (50) NOT NULL,

customer\_lname VARCHAR (50) NOT NULL,

customer\_balance DECIMAL(9, 2),

customer\_street VARCHAR(50) NOT NULL,

customer\_city VARCHAR(50) NOT NULL,

customer\_state VARCHAR(2) NOT NULL,

customer\_ZIP DECIMAL(10) NOT NULL,

PRIMARY KEY (customer\_id));

INSERT INTO customer VALUES(1, 'John', 'Smith', 0, '450 Branham Way', 'Mt Washington', 'KY', 40047);

INSERT INTO customer VALUES(2, 'Mary', 'Berman', 0, '359 Miles Dr', 'Shepherdsville', 'KY', 40165);

INSERT INTO customer VALUES(3, 'Elizabeth', 'Johnson', 0, '2702 Industrial Dr', 'Bowling Green', 'KY', 42101);

INSERT INTO customer VALUES(4, 'Peter', 'Quigley', 0, '2602 Navajo Dr', 'Bowling Green', 'KY', 42101);

INSERT INTO customer VALUES(5, 'Stanton', 'Hurley', 0, '546 Ashland Ave', 'Louisville', 'KY', 40216);

INSERT INTO customer VALUES(6, 'Yvette', 'Presley', 0, '3200 Hebron Estates', 'Shepherdsville', 'KY', 40165);

INSERT INTO customer VALUES(7, 'Hilary', 'Marsh', 0, '345 Stringer ln', 'Mt Washington', 'KY', 40047);

CREATE TABLE Membership(

Membership\_id DECIMAL(10) NOT NULL,

membership\_name VARCHAR(50) NOT NULL,

membership\_description VARCHAR(300),

membership\_price DECIMAL(9,2) NOT NULL,

PRIMARY KEY (membership\_id));

INSERT INTO Membership VALUES(11, 'Broze', '1 at a time program', 14.99);

INSERT INTO Membership VALUES(12, 'Silver', '2 at a time program', 17.99);

INSERT INTO Membership VALUES(13, 'Gold', '3 at a time program', 19.99);

CREATE TABLE Membership\_transaction(

membership\_transaction\_id DECIMAL(10) NOT NULL,

customer\_id DECIMAL(10) NOT NULL,

membership\_id DECIMAL(10) NOT NULL,

date\_purchased DATE NOT NULL,

Active DECIMAL(2) NOT NULL,

PRIMARY KEY (membership\_transaction\_id),

FOREIGN KEY (membership\_id) REFERENCES Membership,

FOREIGN KEY (customer\_id) REFERENCES Customer);

INSERT INTO Membership\_transaction VALUES(501, 1,11, sysdate, 1);

INSERT INTO Membership\_transaction VALUES(502, 2, 12, sysdate, 1);

INSERT INTO Membership\_transaction VALUES(503, 3, 11, sysdate, 1);

INSERT INTO Membership\_transaction VALUES(504, 4, 12, sysdate, 1);

INSERT INTO Membership\_transaction VALUES(505, 5, 13, sysdate, 1);

INSERT INTO Membership\_transaction VALUES(506, 6, 11, sysdate, 1);

INSERT INTO Membership\_transaction VALUES(507, 7, 13, sysdate, 1);

CREATE TABLE Genre(

Genre\_id DECIMAL(10) NOT NULL,

Genre\_name VARCHAR(30) NOT NULL,

PRIMARY KEY (Genre\_id));

INSERT INTO Genre VALUES (400, 'Action');

INSERT INTO Genre VALUES (401, 'Adventure');

INSERT INTO Genre VALUES (402, 'Drama');

INSERT INTO Genre VALUES (403, 'Sci-Fi');

INSERT INTO Genre VALUES (404, 'Horror');

INSERT INTO Genre VALUES (405, 'Comedy');

INSERT INTO Genre VALUES (406, 'Romance');

CREATE TABLE Director(

Director\_id DECIMAL(10) NOT NULL,

Director\_fname VARCHAR(50) NOT NULL,

Director\_lname VARCHAR(50) NOT NULL,

PRIMARY KEY (Director\_id));

INSERT INTO Director VALUES(200, 'Andy and Lana', 'Wachowski');

INSERT INTO Director VALUES(201, 'Brian', 'Helgeland');

INSERT INTO Director VALUES(202, 'Bennett', 'Miller');

INSERT INTO Director VALUES(203, 'James', 'Gunn');

INSERT INTO Director VALUES(204, 'Christopher', 'Nolan');

INSERT INTO Director VALUES(205, 'Steven', 'Spielberg');

INSERT INTO DIRECTOR VALUES(206, 'George', 'Lucas');

INSERT INTO DIRECTOR VALUES(207, 'Rich', 'Christiano');

CREATE TABLE DVD(

DVD\_id DECIMAL(10) NOT NULL,

DVD\_name VARCHAR(30) NOT NULL,

Genre\_id DECIMAL(10) NOT NULL,

Director\_id DECIMAL(10) NOT NULL,

number\_available DECIMAL(10) NOT NULL,

number\_rented DECIMAL(10) NOT NULL,

number\_lost DECIMAL(10) NOT NULL,

PRIMARY KEY(DVD\_id),

FOREIGN KEY (Director\_id) REFERENCES Director,

FOREIGN KEY (Genre\_id) REFERENCES Genre);

INSERT INTO DVD VALUES(300, 'Matrix', 400, 200, 9, 1, 0);

INSERT INTO DVD VALUES(301, 'A Knights Tale', 401, 201, 0, 1, 0);

INSERT INTO DVD VALUES(302, 'Money Ball', 402, 202, 10, 0, 0);

INSERT INTO DVD VALUES(303, 'Guardians of the Galaxy', 403, 203, 10, 0, 0);

INSERT INTO DVD VALUES(304, 'Inception', 400, 204, 10, 0, 0);

INSERT INTO DVD VALUES(305, 'Star Wars Episode I', 403, 206, 10, 0, 0);

INSERT INTO DVD VALUES(306, 'Star Wars Episode II', 403, 206, 10, 0, 0);

INSERT INTO DVD VALUES(307, 'Time Changer', 403, 207, 10, 0, 0);

CREATE TABLE DVD\_que(

Que\_row\_id DECIMAL(10)NOT NULL,

Customer\_id DECIMAL(10) NOT NULL,

DVD\_id DECIMAL(10) NOT NULL,

Que\_position DECIMAL(10) NOT NULL,

PRIMARY KEY (Que\_row\_id),

FOREIGN KEY (Customer\_id) REFERENCES Customer,

FOREIGN KEY (DVD\_id) REFERENCES DVD);

INSERT INTO DVD\_que VALUES(600, 1, 300, 1);

INSERT INTO DVD\_que VALUES(601, 1, 301, 2);

INSERT INTO DVD\_que VALUES(602, 1, 302, 3);

INSERT INTO DVD\_que VALUES(603, 1, 303, 4);

INSERT INTO DVD\_que VALUES(604, 1, 304, 5);

INSERT INTO DVD\_que VALUES(605, 2, 302, 1);

INSERT INTO DVD\_que VALUES(606, 2, 301, 2);

INSERT INTO DVD\_que VALUES(607, 2, 304, 3);

INSERT INTO DVD\_que VALUES(608, 2, 303, 4);

CREATE TABLE Rental\_transaction(

Rental\_transaction\_id DECIMAL(10) NOT NULL,

Customer\_id DECIMAL(10) NOT NULL,

DVD\_id DECIMAL (10) NOT NULL,

Date\_rented DATE,

Date\_Returned DATE,

PRIMARY KEY(Rental\_transaction\_id),

FOREIGN KEY(customer\_id) REFERENCES Customer,

FOREIGN KEY(DVD\_id) REFERENCES DVD);

INSERT INTO Rental\_Transaction VALUES (900, 1, 300, SYSDATE, SYSDATE);

INSERT INTO Rental\_Transaction VALUES (901, 1, 301, SYSDATE, NULL);

INSERT INTO Rental\_Transaction VALUES (902, 1, 300, SYSDATE, NULL);

INSERT INTO Rental\_Transaction VALUES (903, 3, 303, SYSDATE, NULL);

INSERT INTO Rental\_Transaction VALUES (904, 3, 304, SYSDATE, NULL);

INSERT INTO Rental\_Transaction VALUES (905, 5, 303, SYSDATE, NULL);

INSERT INTO Rental\_Transaction VALUES (906, 5, 304, SYSDATE, NULL);

**Queries/Stored Procedures:**

**1.)**

SELECT DVD\_name, Director\_fname, Director\_lname

FROM DVD JOIN Director ON DVD.Director\_id = Director.Director\_id

WHERE Director.Director\_lname = 'Lucas' AND (Director.Director\_fname = 'George')

OR Director.Director\_lname = 'Christiano' AND (Director.Director\_fname = 'Rich');

**2.)**

SELECT Customer\_fname, customer\_lname, membership\_name

FROM Customer JOIN Membership\_transaction on Customer.customer\_id = membership\_transaction.customer\_id

JOIN membership on membership.membership\_id = membership\_transaction.membership\_id

WHERE membership\_transaction.active = 1;

**3.)**

CREATE OR REPLACE PROCEDURE Front\_of\_que(

Que\_row\_id\_arg IN DECIMAL,

Customer\_id\_arg IN DECIMAL,

DVD\_id\_arg IN DECIMAL,

Que\_position\_arg IN DECIMAL)

IS

BEGIN

UPDATE DVD\_que

SET que\_position = Que\_position\_arg

WHERE customer\_id\_arg = customer\_id and que\_position = 1;

INSERT INTO DVD\_Que

(Que\_row\_id, Customer\_id, DVD\_id, Que\_position)

VALUES(Que\_row\_id\_arg, Customer\_id\_arg, DVD\_id\_arg, 1);

END;

BEGIN

Front\_of\_que(609, 1, 305, 6);

END;

/

**4.)**

SELECT DVD.DVD\_name

FROM DVD

left join Rental\_Transaction on Rental\_transaction.dvd\_id = dvd.dvd\_id

WHERE NOT EXISTS (SELECT dvd.dvd\_name from dvd WHERE rental\_transaction.customer\_id = 1);

**5.)**

CREATE OR REPLACE PROCEDURE Fee(

customer\_id\_arg IN DECIMAL)

IS

BEGIN

UPDATE Customer

SET customer\_balance = +25

WHERE customer\_id\_arg = customer\_id;

END;

Begin

FEE(4);

END;

/

**6.)**

CREATE OR REPLACE PROCEDURE DEACTIVATE(

Customer\_id\_arg IN DECIMAL)

IS

BEGIN

UPDATE membership\_transaction

SET Active = 0

WHERE customer\_id\_arg = customer\_id;

END;

BEGIN

DEACTIVATE(2);

END;

/

**7.)**

SELECT dvd.dvd\_name

from DVD

WHERE Number\_available = 0;

**8.)**

SELECT membership.membership\_id, membership.membership\_name,

membership.membership\_description, COUNT(membership\_transaction.membership\_id) AS NR\_Enrollees

FROM Membership

JOIN membership\_transaction on membership.membership\_id = membership\_transaction.membership\_id

WHERE Active = 1

GROUP BY membership.membership\_id, membership.membership\_name, membership.membership\_description

having count(membership\_transaction.membership\_id) = (select max(cnt)

from (select membership\_transaction.membership\_id, count(membership\_transaction.membership\_id) as cnt

from membership\_transaction

group by membership\_Transaction.membership\_id

) rc

);

**9.)**

SELECT customer.customer\_fname, customer.customer\_lname, dvd.dvd\_name,

count(rental\_Transaction.dvd\_id) AS NR\_Rented

FROM Customer

JOIN rental\_transaction on customer.customer\_id = rental\_transaction.customer\_id

JOIN DVD on dvd.dvd\_id = rental\_transaction.dvd\_id

GROUP BY customer.customer\_fname, customer.customer\_lname, dvd.dvd\_name

HAVING COUNT(rental\_transaction.dvd\_id) > 1;

**10.)**

SELECT dvd.dvd\_name, count(distinct rental\_Transaction.customer\_id) AS NR\_Times\_Rented

FROM rental\_transaction

join DVD on DVD.dvd\_id = rental\_Transaction.dvd\_id

group by dvd.dvd\_name

ORDER BY NR\_Times\_Rented DESC;

**Creating Indexes**

CREATE INDEX dvd\_id\_idx ON Rental\_Transaction(dvd\_id);

CREATE INDEX Membership\_id\_idx ON Membership\_Transaction(membership\_id);