CREATE DATABASE rentalVCD;

use rentalVCD;

CREATE TABLE RatingVCD(

RatingID VARCHAR (20) PRIMARY KEY NOT NULL,

RatingName VARCHAR (20) NOT NULL,

AgeRequirements INT NOT NULL

);

INSERT INTO RatingVCD(RatingID, RatingName, AgeRequirements)

VALUES

('P', 'Infant', 2),

('A', 'Kids', 7),

('R', 'Teen', 13),

('D', 'Adult', 18);

CREATE TABLE GenreVCD(

GenreCode VARCHAR (20) PRIMARY KEY NOT NULL,

GenreName VARCHAR (20) NOT NULL

);

INSERT INTO GenreVCD (GenreCode, GenreName)

VALUES

('GN001', 'Action'),

('GN002', 'Romance'),

('GN003', 'Mystery'),

('GN004', 'Superhero'),

('GN005', 'Comedy'),

('GN006', 'Sci-Fi'),

('GN007', 'Fantasy'),

('GN008', 'Horror'),

('GN009', 'Family'),

('GN010', 'Historical'),

('GN011', 'Sports'),

('GN012', 'Music');

CREATE TABLE Customer(

CustomerCode VARCHAR (20) PRIMARY KEY NOT NULL,

CustomerName VARCHAR (50) NOT NULL,

CustomerAddress VARCHAR (50) NOT NULL,

CustomerPhone VARCHAR (20) NOT NULL,

CustomerEmail VARCHAR (30) NOT NULL,

CustomerDOB DATE NOT NULL

);

INSERT INTO Customer(CustomerCode, CustomerName, CustomerAddress, CustomerPhone, CustomerEmail, CustomerDOB)

VALUES

('CUT00001','John Doe' , '123 Main St, Anytown, USA' , '555-123-4567', 'john.doe@example.com' , '1990-05-15'),

('CUT00002','Jane Smith' , '456 Elm St, Another Town, USA' , '555-987-6543', 'jane.smith@example.com' , '1985-12-10'),

('CUT00003','Michael Johnson' , '79 Oak St, Yet Another Town, USA', '555-567-8901', 'michael.j@example.com' , '1995-02-28'),

('CUT00004','Sarah Brown' , '234 Pine St, Different Town, USA', '555-345-6789', 'sarah.b@example.com' , '1980-07-20'),

('CUT00005','David Lee' , '987 Birch St, Nearby Town, USA' , '555-234-5678', 'david.lee@example.com' , '1998-09-05'),

('CUT00006','Emily Davis' , '567 Cedar St, Faraway Town, USA' , '555-456-7890', 'emily.d@example.com' , '1987-04-18'),

('CUT00007','Christopher White' , '321 Maple St, Distant Town, USA' , '555-789-1234', 'chris.white@example.com' , '1993-10-30'),

('CUT00008','Jennifer Wilson' , '111 Walnut St, Remote Town, USA' , '555-012-3456', 'jennifer.w@example.com' , '1989-03-25'),

('CUT00009','Matthew Harris' , '654 Redwood St, Tiny Town, USA' , '555-890-1234', 'matthew.h@example.com' , '1996-06-12'),

('CUT00010','Linda Turner' , '876 Spruce St, Small Town, USA' , '555-456-7890', 'linda.t@example.com' , '1986-01-08'),

('CUT00011','Alice Johnson' , '123 Oak St, Anytown, USA' , '555-123-4567', 'alice@example.com' , '1990-05-15'),

('CUT00012','Bob Smith' , '456 Maple Dr, Another Town, USA' , '555-987-6543', 'bob@example.com' , '1985-12-10'),

('CUT00013','Charlie Davis' , '789 Pi Rd, Yet Another City, USA', '555-567-8901', 'charlie@example.com' , '1995-02-28'),

('CUT00014','David Wilson' , '234 Elm St, Differentville, USA' , '555-345-6789', 'david@example.com' , '1980-07-20'),

('CUT00015','Eve White' , '987 Birch Ln, Nearbyville, USA' , '555-234-5678', 'eve@example.com' , '1998-09-05'),

('CUT00016','Fiona Turner' , '567 Cedar Rd, Faraway City, USA' , '555-456-7890', 'fiona@example.com' , '1987-04-18'),

('CUT00017','George Brown' , '321 Walnut Dr, Remoteville, USA' , '555-789-1234', 'george@example.com' , '1993-10-30'),

('CUT00018','Helen Johnson' , '111 Maple Ln, Outskirt City, USA', '555-012-3456', 'helen@example.com' , '1989-03-25'),

('CUT00019','Ian Miller' , '654 Redwood Rd, Countryside, USA', '555-890-1234', 'ian@example.com' , '1996-06-12'),

('CUT00020','Jenny Lee' , '876 Spruce St, Smalltown, USA' , '555-456-7890', 'jenny@example.com' , '1986-01-08');

CREATE TABLE Employee(

EmployeeCode VARCHAR (20) PRIMARY KEY NOT NULL,

EmployeeName VARCHAR (50) NOT NULL,

EmployeeAddress VARCHAR (50) NOT NULL,

EmployeePhone VARCHAR (20) NOT NULL,

EmployeeEmail VARCHAR (30) NOT NULL

);

INSERT INTO Employee(EmployeeCode, EmployeeName, EmployeeAddress, EmployeePhone, EmployeeEmail)

VALUES

('EM001','Amanda Johnson' , '543 Willow Ln, Smallville, USA' , '555-234-5678', 'amanda.j@example.com') ,

('EM002','Robert Miller' , '789 Oak Ave, Suburbia, USA' , '555-123-4567', 'robert.m@example.com') ,

('EM003','Lisa Anderson' , '456 Pine St, Quaint Town, USA' , '555-567-8901', 'lisa.a@example.com') ,

('EM004','William Davis' , '321 Elm Rd, Ruralville, USA' , '555-789-1234', 'william.d@example.com') ,

('EM005','Melissa Martinez' , '876 Birch Dr, Countryside, USA' , '555-345-6789', 'melissa.m@example.com') ,

('EM006','Daniel Taylor' , '234 Cedar Dr, Villageville, USA' , '555-678-9012', 'daniel.t@example.com') ,

('EM007','Sarah Harris' , '987 Redwood Dr, Hometown, USA' , '555-456-7890', 'sarah.h@example.com') ,

('EM008','John Wilson' , '543 Maple Dr, Peaceful Town, USA', '555-012-3456', 'john.w@example.com') ,

('EM009','Emily Lewis' , '111 Walnut Dr, Sereneville, USA' , '555-890-1234', 'emily.l@example.com') ,

('EM010','Michael Clark' , '654 Spruce Dr, Tranquiltown, USA', '555-890-5678', 'michael.c@example.com') ;

CREATE TABLE VCD(

VCDCode VARCHAR (20) PRIMARY KEY NOT NULL,

VCDName VARCHAR (30) NOT NULL,

VCDRating VARCHAR (20) NOT NULL,

GenreCode VARCHAR (20) NOT NULL,

VCDStock INT CHECK (VCDStock >= 0),

FOREIGN KEY(VCDRating)

REFERENCES RatingVCD (RatingID)

ON UPDATE CASCADE

ON DELETE CASCADE,

FOREIGN KEY(GenreCode)

REFERENCES GenreVCD (GenreCode)

ON UPDATE CASCADE

ON DELETE CASCADE

);

INSERT INTO VCD(VCDCode, VCDName, VCDRating, GenreCode, VCDStock)

VALUES

('VCD001', 'The Action Hero' , 'A', 'GN001', 10),

('VCD002', 'Romantic Rendezvous' , 'R', 'GN002', 10),

('VCD003', 'Mystery Mansion' , 'D', 'GN003', 10),

('VCD004', 'Superhero Saga' , 'A', 'GN004', 10),

('VCD005', 'Comedy Club' , 'R', 'GN005', 10),

('VCD006', 'Sci-Fi Spectacle' , 'A', 'GN006', 10),

('VCD007', 'Fantasy Quest' , 'R', 'GN007', 10),

('VCD008', 'Horror House' , 'D', 'GN008', 10),

('VCD009', 'Family Funtime' , 'R', 'GN009', 10),

('VCD010', 'Historical Journey' , 'D', 'GN010', 10),

('VCD011', 'Sports Showdown' , 'A', 'GN011', 10),

('VCD012', 'Music Mania' , 'R', 'GN012', 10),

('VCD013', 'Political Pioneers' , 'D', 'GN001', 10),

('VCD014', 'Adventure Awaits' , 'A', 'GN002', 10),

('VCD015', 'Animation Adventure' , 'R', 'GN003', 10),

('VCD016', 'Documentary Dive' , 'D', 'GN004', 10),

('VCD017', 'War Warriors' , 'A', 'GN005', 10),

('VCD018', 'Zombie Zone' , 'A', 'GN006', 10),

('VCD019', 'Spy Secrets' , 'R', 'GN007', 10),

('VCD020', 'Rom-Com Romance' , 'R', 'GN008', 10),

('VCD021', 'Western Wonders' , 'A', 'GN009', 10),

('VCD022', 'Post-Apocalyptic Paradox' , 'D', 'GN010', 10),

('VCD023', 'Mockumentary Madness' , 'R', 'GN011', 10),

('VCD024', 'Biographical Bliss' , 'D', 'GN012', 10),

('VCD025', 'Fantasy Funland' , 'A', 'GN001', 10);

CREATE TABLE TransactionHeader(

TransactionCode VARCHAR (20) PRIMARY KEY NOT NULL,

CustomerCode VARCHAR (20) NOT NULL,

EmployeeCode VARCHAR (20) NOT NULL,

RentDate DATE NOT NULL,

DueDate DATE NOT NULL,

RentStatus INT,

FOREIGN KEY (CustomerCode)

REFERENCES Customer(CustomerCode)

ON UPDATE CASCADE

ON DELETE CASCADE,

FOREIGN KEY (EmployeeCode)

REFERENCES Employee (EmployeeCode)

ON UPDATE CASCADE

ON DELETE CASCADE

);

INSERT INTO TransactionHeader(TransactionCode, CustomerCode, EmployeeCode, RentDate, DueDate, RentStatus)

VALUES

('TXN00001', 'CUT00001', 'EM001', '2023-10-01', '2023-10-10', 1),

('TXN00002', 'CUT00002', 'EM002', '2023-10-02', '2023-10-11', 0),

('TXN00003', 'CUT00003', 'EM003', '2023-10-03', '2023-10-12', 1),

('TXN00004', 'CUT00004', 'EM004', '2023-10-04', '2023-10-13', 1),

('TXN00005', 'CUT00005', 'EM005', '2023-10-05', '2023-10-14', 0),

('TXN00006', 'CUT00006', 'EM006', '2023-10-06', '2023-10-15', 1),

('TXN00007', 'CUT00007', 'EM007', '2023-10-07', '2023-10-16', 1),

('TXN00008', 'CUT00008', 'EM008', '2023-10-08', '2023-10-17', 0),

('TXN00009', 'CUT00009', 'EM009', '2023-10-09', '2023-10-18', 1),

('TXN00010', 'CUT00010', 'EM010', '2023-10-10', '2023-10-19', 1),

('TXN00011', 'CUT00011', 'EM001', '2023-10-11', '2023-10-20', 1),

('TXN00012', 'CUT00012', 'EM002', '2023-10-12', '2023-10-21', 1),

('TXN00013', 'CUT00013', 'EM003', '2023-10-13', '2023-10-22', 0),

('TXN00014', 'CUT00014', 'EM004', '2023-10-14', '2023-10-23', 1),

('TXN00015', 'CUT00015', 'EM005', '2023-10-15', '2023-10-24', 1),

('TXN00016', 'CUT00016', 'EM006', '2023-10-16', '2023-10-25', 1),

('TXN00017', 'CUT00017', 'EM007', '2023-10-17', '2023-10-26', 1),

('TXN00018', 'CUT00018', 'EM008', '2023-10-18', '2023-10-27', 0),

('TXN00019', 'CUT00019', 'EM009', '2023-10-19', '2023-10-28', 0),

('TXN00020', 'CUT00020', 'EM010', '2023-10-20', '2023-10-29', 0);

CREATE TABLE TransactionDetail(

TransactionCode VARCHAR (20) NOT NULL,

VCDCode VARCHAR (20) NOT NULL,

FOREIGN KEY (TransactionCode)

REFERENCES TransactionHeader(TransactionCode)

ON UPDATE CASCADE

ON DELETE CASCADE,

FOREIGN KEY (VCDCode)

REFERENCES VCD(VCDCode)

ON UPDATE CASCADE

ON DELETE CASCADE

);

INSERT INTO TransactionDetail(TransactionCode, VCDCode)

VALUES

('TXN00001', 'VCD001'),

('TXN00002', 'VCD002'),

('TXN00003', 'VCD003'),

('TXN00004', 'VCD004'),

('TXN00005', 'VCD005'),

('TXN00006', 'VCD006'),

('TXN00007', 'VCD007'),

('TXN00008', 'VCD008'),

('TXN00009', 'VCD009'),

('TXN00010', 'VCD010'),

('TXN00011', 'VCD011'),

('TXN00012', 'VCD012'),

('TXN00013', 'VCD013'),

('TXN00014', 'VCD014'),

('TXN00015', 'VCD015'),

('TXN00016', 'VCD016'),

('TXN00017', 'VCD017'),

('TXN00018', 'VCD018'),

('TXN00019', 'VCD019'),

('TXN00020', 'VCD020'),

('TXN00001', 'VCD021'),

('TXN00002', 'VCD022'),

('TXN00003', 'VCD023'),

('TXN00004', 'VCD024'),

('TXN00005', 'VCD025'),

('TXN00006', 'VCD001'),

('TXN00007', 'VCD002'),

('TXN00008', 'VCD003'),

('TXN00009', 'VCD004'),

('TXN00010', 'VCD005'),

('TXN00011', 'VCD006'),

('TXN00012', 'VCD007'),

('TXN00013', 'VCD008'),

('TXN00014', 'VCD009'),

('TXN00015', 'VCD010'),

('TXN00016', 'VCD011'),

('TXN00017', 'VCD012'),

('TXN00018', 'VCD013'),

('TXN00019', 'VCD014'),

('TXN00020', 'VCD015'),

('TXN00011', 'VCD016'),

('TXN00012', 'VCD017'),

('TXN00003', 'VCD018'),

('TXN00014', 'VCD019'),

('TXN00005', 'VCD020'),

('TXN00006', 'VCD021'),

('TXN00007', 'VCD022'),

('TXN00008', 'VCD023'),

('TXN00009', 'VCD024'),

('TXN00020', 'VCD025');

CREATE VIEW view\_tabel\_pelanggan\_yang\_sedang\_meminjam AS

SELECT th.TransactionCode as 'Transaction ID',

cus.CustomerCode as 'Customer Code',

cus.CustomerName as 'Customer Name',

emp.EmployeeCode as 'Employee Code',

emp.EmployeeName as 'Employee Name',

th.DueDate as 'Due Date'

FROM transactionheader AS th JOIN customer AS cus

ON th.CustomerCode = cus.CustomerCode

JOIN Employee AS emp

ON th.EmployeeCode = emp.EmployeeCode

WHERE th.RentStatus = 0;

CREATE VIEW view\_detail\_vcd\_yang\_tersedia AS

SELECT vcd.VCDCode as 'VCD Code',

vcd.VCDName as 'VCD Name',

vcd.VCDRating as 'Rating',

ratingvcd.AgeRequirements as 'Min. Age Required',

genrevcd.GenreName as 'Genre',

vcd.VCDStock as 'Avaiable Amount'

FROM vcd JOIN ratingvcd

ON vcd.VCDRating = ratingvcd.RatingID

JOIN genrevcd

ON vcd.GenreCode = genrevcd.GenreCode

WHERE vcd.VCDStock > 0

ORDER BY ratingvcd.AgeRequirements ASC;

CREATE VIEW view\_top\_5\_vcd\_paling\_sering\_dipinjam AS

SELECT td.VCDCode,

vcd.VCDName,

COUNT(\*) AS JumlahPeminjaman

FROM TransactionDetail as td

JOIN vcd ON td.VCDCode = vcd.VCDCode

GROUP BY VCDCode

ORDER BY JumlahPeminjaman DESC

LIMIT 5;

CREATE VIEW view\_total\_jumlah\_transaksi AS

SELECT COUNT(\*) AS TotalTransaksiRental

FROM TransactionHeader;

DELIMITER //

CREATE PROCEDURE GetCustomerTransactions(IN customerCodeParam VARCHAR(20))

BEGIN

SELECT

TH.TransactionCode,

TH.RentDate,

TH.DueDate,

TH.RentStatus,

V.VCDCode,

V.VCDName,

V.VCDRating,

GV.GenreName

FROM

TransactionHeader TH

JOIN TransactionDetail TD ON TH.TransactionCode = TD.TransactionCode

JOIN VCD V ON TD.VCDCode = V.VCDCode

JOIN GenreVCD GV ON V.GenreCode = GV.GenreCode

WHERE

TH.CustomerCode = customerCodeParam;

END //

DELIMITER ;

DELIMITER //

CREATE FUNCTION GetAvailableVCDStockByGenre(genreCodeParam VARCHAR(20)) RETURNS INT

BEGIN

DECLARE availableStock INT;

SELECT

SUM(V.VCDStock) INTO availableStock

FROM

VCD V

WHERE

V.GenreCode = genreCodeParam

AND V.VCDCode NOT IN (

SELECT TD.VCDCode

FROM TransactionDetail TD

JOIN TransactionHeader TH ON TD.TransactionCode = TH.TransactionCode

WHERE TH.RentStatus = 1

);

RETURN availableStock;

END //

DELIMITER ;