

Table of Contents

[Question 1 3](#_Toc169601714)

[1.1.a 3](#_Toc169601715)

[1.1.b 4](#_Toc169601716)

[1.2.a 6](#_Toc169601717)

[1.2.b 6](#_Toc169601718)

[1.2.c 6](#_Toc169601719)

[1.3.a 6](#_Toc169601720)

[1.3.b 6](#_Toc169601721)

[1.3.c 6](#_Toc169601722)

[1.3.d 7](#_Toc169601723)

[1.3.e 7](#_Toc169601724)

[1.3.f 7](#_Toc169601725)

[1.3.e 8](#_Toc169601726)

[Question 2 8](#_Toc169601727)

[2.1 8](#_Toc169601728)

[2.2 8](#_Toc169601729)

[2.3 8](#_Toc169601730)

[2.4 9](#_Toc169601731)

[2.5 12](#_Toc169601732)

[2.6 13](#_Toc169601733)

[2.7 15](#_Toc169601734)

[2.8 16](#_Toc169601735)

[2.9 16](#_Toc169601736)

[2.10 17](#_Toc169601737)

[References 18](#_Toc169601738)

# Question 1

## 1.1.a

CREATE TABLE TITLE (

TITLE\_ID VARCHAR(10) NOT NULL,

TITLE VARCHAR(32) NOT NULL,

DESCRIPTION VARCHAR(256) NOT NULL,

RATING NUMBER(3,1) NOT NULL,

CATEGORY VARCHAR(15) NOT NULL,

RELEASEDATE DATE NOT NULL,

PRICE NUMBER(8,2) NOT NULL,

PRIMARY KEY (TITLE\_ID)

);

CREATE TABLE MEMBER (

MEMBER\_ID VARCHAR(10) NOT NULL,

LASTNAME VARCHAR(32) NOT NULL,

FIRSTNAME VARCHAR(32) NOT NULL,

ADDRESS VARCHAR(128) NOT NULL,

CITY VARCHAR(32) NOT NULL,

PHONE NUMBER NOT NULL,

JOINDATE DATE NOT NULL,

PRIMARY KEY (MEMBER\_ID)

);

CREATE TABLE TITLE\_COPY (

COPY\_ID VARCHAR(10) NOT NULL,

TITLE\_ID VARCHAR(10) NOT NULL,

STATUS VARCHAR(13) NOT NULL,

PRIMARY KEY (COPY\_ID),

FOREIGN KEY (TITLE\_ID) REFERENCES TITLE(TITLE\_ID)

);

CREATE TABLE RENTAL (

BOOKDATE DATE NOT NULL,

ACCTRENTALDATE DATE NOT NULL,

EXPRENTALDATE DATE NOT NULL,

COPY\_ID VARCHAR(10) NOT NULL,

MEMBER\_ID VARCHAR(10) NOT NULL,

PRIMARY KEY (BOOKDATE),

FOREIGN KEY (COPY\_ID) REFERENCES TITLE\_COPY (COPY\_ID),

FOREIGN KEY (MEMBER\_ID) REFERENCES MEMBER (MEMBER\_ID)

);

CREATE TABLE RESERVATION (

RESDATE DATE NOT NULL,

TITLE\_ID VARCHAR(10) NOT NULL,

MEMBER\_ID VARCHAR(10) NOT NULL,

PRIMARY KEY (RESDATE),

FOREIGN KEY (TITLE\_ID) REFERENCES TITLE (TITLE\_ID),

FOREIGN KEY (MEMBER\_ID) REFERENCES MEMBER (MEMBER\_ID)

);

## 1.1.b

INSERT INTO TITLE VALUES ('T001', 'The Shawshank Redemption', 'Two imprisoned men bond over a number of years, finding solace and eventual redemption through acts of common decency.', 9.3, 'Drama', '1994-10-14', 12.99);

INSERT INTO TITLE VALUES ('T002', 'Inception', 'A thief who enters the dreams of others to steal secrets from their subconscious. He is offered a chance to have his criminal history erased as payment for the implantation of another persons idea into a targets subconscious.', 8.8, 'Sci-Fi', DATE '2010-07-16', 14.99);

INSERT INTO TITLE VALUES ('T003', 'The Godfather', 'The aging patriarch of an organized crime dynasty transfers control of his clandestine empire to his reluctant son.', 9.2, 'Crime', DATE '1972-03-24', 11.99);

INSERT INTO TITLE VALUES ('T004', 'The Dark Knight', 'When the menace known as The Joker emerges from his mysterious past, he wreaks havoc and chaos on the people of Gotham. The Dark Knight must accept one of the greatest psychological and physical tests of his ability to fight injustice.', 9.0, 'Action', DATE '2008-07-18', 13.99);

INSERT INTO TITLE VALUES ('T005', 'Forrest Gump', 'The presidencies of Kennedy and Johnson, the events of Vietnam, Watergate, and other historical events unfold from the perspective of an Alabama man with an IQ of 75, whose only desire is to be reunited with his childhood sweetheart.', 8.8, 'Drama', DATE '1994-07-06', 10.99);

INSERT INTO MEMBER VALUES ('M001', 'Smith', 'John', '123 Main Street', 'New York', '0115551234', DATE '2023-05-10');

INSERT INTO MEMBER VALUES ('M002', 'Johnson', 'Emily', '456 Elm Avenue', 'Los Angeles', '0215555678',DATE '2022-11-30');

INSERT INTO MEMBER VALUES ('M003', 'Garcia', 'Carlos', '789 Oak Street', 'Chicago', '0315559012',DATE '2024-01-15');

INSERT INTO MEMBER VALUES ('M004', 'Brown', 'Jessica', '101 Pine Lane', 'Houston', '0415553456',DATE '2023-09-22');

INSERT INTO MEMBER VALUES ('M005', 'Martinez', 'Maria', '222 Cedar Street', 'Miami', '0515557890',DATE '2022-08-05');

INSERT INTO TITLE\_COPY VALUES ('C001', 'T001', 'Available');

INSERT INTO TITLE\_COPY VALUES ('C002', 'T002', 'Rented');

INSERT INTO TITLE\_COPY VALUES ('C003', 'T003', 'Available');

INSERT INTO TITLE\_COPY VALUES ('C004', 'T002', 'Available');

INSERT INTO TITLE\_COPY VALUES ('C005', 'T004', 'Available');

INSERT INTO RENTAL VALUES (DATE '2023-05-20', DATE '2023-05-20', DATE '2023-06-20', 'C001', 'M001');

INSERT INTO RENTAL VALUES (DATE '2023-06-10', DATE '2023-06-10', DATE '2023-07-10', 'C002', 'M002');

INSERT INTO RENTAL VALUES (DATE '2023-07-05', DATE '2023-07-05', DATE '2023-08-05', 'C003', 'M003');

INSERT INTO RENTAL VALUES (DATE '2023-08-15', DATE '2023-08-15', DATE '2023-09-15', 'C004', 'M004');

INSERT INTO RENTAL VALUES (DATE '2023-09-10', DATE '2023-09-10', DATE '2023-10-10', 'C005', 'M005');

INSERT INTO RESERVATION VALUES (DATE '2023-05-25', 'T001', 'M001');

INSERT INTO RESERVATION VALUES (DATE '2023-06-15', 'T002', 'M002');

INSERT INTO RESERVATION VALUES (DATE '2023-07-10', 'T003', 'M003');

INSERT INTO RESERVATION VALUES (DATE '2023-08-20', 'T004', 'M004');

INSERT INTO RESERVATION VALUES (DATE '2023-09-05', 'T005', 'M005');

## 1.2.a

UPDATE TITLE

SET TITLE.PRICE = 200

WHERE TITLE.TITLE\_ID = 'T001';

## 1.2.b

SELECT TITLE, PRICE FROM TITLE;

## 1.2.c

SELECT FIRSTNAME, TITLE, BOOKDATE, (rental.exprentaldate - rental.bookdate) AS DURATION

FROM TITLE, MEMBER, RENTAL, TITLE\_COPY

WHERE title\_copy.status = 'Rented';

## 1.3.a

CREATE VIEW MemberInfo

AS

SELECT \* FROM MEMBER;

## 1.3.b

SELECT \* FROM memberinfo ORDER BY member\_id DESC;

## 1.3.c

CREATE VIEW MOVIE\_VIEW AS SELECT TITLE, STATUS, EXPRENTALDATE FROM TITLE, TITLE\_COPY, RENTAL WHERE title\_copy.copy\_id = rental.copy\_id;

## 1.3.d

SELECT \* FROM MOVIE\_VIEW ORDER BY title;

## 1.3.e

CREATE INDEX title\_description ON TITLE(Description);

## 1.3.f

CREATE OR REPLACE FUNCTION MovieTitleDiscount (title\_Name IN VARCHAR)

RETURN VARCHAR

IS

MESSAGE VARCHAR(100);

DISCOUNTEDPRICE NUMBER(10,2);

BEGIN

SELECT CASE

WHEN TITLE\_COPY.STATUS = 'Rented' THEN '40% Discount applied'

ELSE 'Title is available'

END

INTO MESSAGE

FROM TITLE

JOIN TITLE\_COPY ON TITLE.TITLE\_ID = TITLE\_COPY.TITLE\_ID

WHERE TITLE.TITLE = title\_Name;

SELECT CASE

WHEN TITLE\_COPY.STATUS = 'Rented' THEN TITLE.PRICE \* 0.6

ELSE TITLE.PRICE

END

INTO DISCOUNTEDPRICE

FROM TITLE

JOIN TITLE\_COPY ON TITLE.TITLE\_ID = TITLE\_COPY.TITLE\_ID

WHERE TITLE.TITLE = title\_Name;

RETURN MESSAGE || ' - Title: ' || title\_Name || ', Price: ' || TO\_CHAR(DISCOUNTEDPRICE, '999.99');

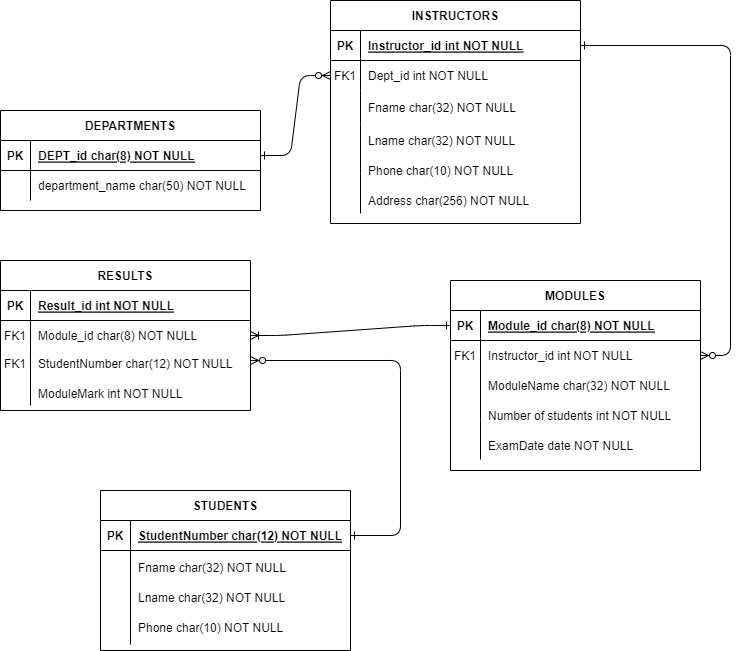
END;

## 1.3.e

SELECT MovieTitleDiscount('The Godfather') AS Discount\_Info FROM dual;

# Question 2

## 2.1



## 2.2

use EduEnrolment

## 2.3

db.Department.insertOne({"DepID": 1, "DepName": "IT Department" });

{

acknowledged: true,

insertedId: ObjectId('6666bba8fa307ceca5cdcdfa')

}

db.Students.insertOne({"StudentNumber": "STU001", "FName": "James", "LName": "Wright", "Phone": "0812682323" });

{

acknowledged: true,

insertedId: ObjectId('6666bc76fa307ceca5cdcdfb')

}

db.Instructors.insertOne({"InstructorID": 1, "Department\_DepID": 1, "FName": "Wendy", "LName": "Tonga", "Phone": "0825632486", "Address": "17 Bicker Street, Midrand, Gauteng" });

{

acknowledged: true,

insertedId: ObjectId('6666bf6bfa307ceca5cdce00')

}

db.Modules.insertOne({"ModuleID": "CPP1513", "Instructors\_InstructorID": 1, "ModuleName": "C++ Programming and development", "NumberOfStudents": 15, "ExamDate": "2024-07-01"});

{

acknowledged: true,

insertedId: ObjectId('6666c905fa307ceca5cdce05')

}

db.Results.insertOne({"ResultID": 1, "Modules\_ModuleID": "CPP1513", "Students\_StudentNumber ": "STU001", "ModuleMark": 85});

{

acknowledged: true,

insertedId: ObjectId('6666cc3dfa307ceca5cdce0a')

}

## 2.4

db.Department.insertMany([{"DepID": 2, "DepName": "Law Department"},

{"DepID": 3, "DepName": "Accounting Department"},

{"DepID": 4, "DepName": "Humanities and Art Department"},

{"DepID": 5, "DepName": "Science Department"}]);

{

acknowledged: true,

insertedIds: {

'0': ObjectId('6666bb39fa307ceca5cdcdf6'),

'1': ObjectId('6666bb39fa307ceca5cdcdf7'),

'2': ObjectId('6666bb39fa307ceca5cdcdf8'),

'3': ObjectId('6666bb39fa307ceca5cdcdf9')

}

}

db.Students.insertMany([{"StudentNumber": "STU002", "FName": "Lindsay", "LName": "Coleman", "Phone": "0712682223" }, {"StudentNumber": "STU003", "FName": "Dick", "LName": "Romans", "Phone": "0612612343" }, {"StudentNumber": "STU004", "FName": "Jacob", "LName": "White", "Phone": "0615234123" }, {"StudentNumber": "STU005", "FName": "Isak", "LName": "Pillay", "Phone": "0718543269" }]);

{

acknowledged: true,

insertedIds: {

'0': ObjectId('6666bd8dfa307ceca5cdcdfc'),

'1': ObjectId('6666bd8dfa307ceca5cdcdfd'),

'2': ObjectId('6666bd8dfa307ceca5cdcdfe'),

'3': ObjectId('6666bd8dfa307ceca5cdcdff')

}

}

db.Instructors.insertMany([{"InstructorID": 2, "Department\_DepID": 2, "FName": "Jessica", "LName": "Hill", "Phone": "0716542389", "Address": "1 Kobalt Street, Midrand, Gauteng" }, {"InstructorID": 3, "Department\_DepID": 3, "FName": "Phillip", "LName": "Setonga", "Phone": "0612354789", "Address": "8 Nice Lane, Sandton, Gauteng" }, {"InstructorID": 4, "Department\_DepID": 4, "FName": "Lana", "LName": "Jacobsons", "Phone": "0630253461", "Address": "65 Loner Road, Edenvale, Gauteng" }, {"InstructorID": 5, "Department\_DepID": 5, "FName": "Elias", "LName": "Ngwena", "Phone": "0812357896", "Address": "100 CloutMain Avenue, Sandton, Gauteng" }]);

{

acknowledged: true,

insertedIds: {

'0': ObjectId('6666c118fa307ceca5cdce01'),

'1': ObjectId('6666c118fa307ceca5cdce02'),

'2': ObjectId('6666c118fa307ceca5cdce03'),

'3': ObjectId('6666c118fa307ceca5cdce04')

}

}

db.Modules.insertMany([{"ModuleID": "ILW322", "Instructors\_InstructorID": 2, "ModuleName": "Introduction to Law", "NumberOfStudents": 20, "ExamDate": "2024-07-11"},

{"ModuleID": "FAC428", "Instructors\_InstructorID": 3, "ModuleName": "Financial Accounting", "NumberOfStudents": 28, "ExamDate": "2024-07-05"},

{"ModuleID": "HOA758", "Instructors\_InstructorID": 4, "ModuleName": "History of Art", "NumberOfStudents": 17, "ExamDate": "2024-07-08"},

{"ModuleID": "ITB528", "Instructors\_InstructorID": 5, "ModuleName": "Introduction in Biology", "NumberOfStudents": 30, "ExamDate": "2024-07-05"},]);

{

acknowledged: true,

insertedIds: {

'0': ObjectId('6666cb02fa307ceca5cdce06'),

'1': ObjectId('6666cb02fa307ceca5cdce07'),

'2': ObjectId('6666cb02fa307ceca5cdce08'),

'3': ObjectId('6666cb02fa307ceca5cdce09')

}

}

db.Results.insertMany([{"ResultID": 2, "Modules\_ModuleID": "ILW322", "Students\_StudentNumber": "STU002", "ModuleMark": 92},

{"ResultID": 3, "Modules\_ModuleID": "FAC428", "Students\_StudentNumber": "STU003", "ModuleMark": 71},

{"ResultID": 4, "Modules\_ModuleID": "HOA758", "Students\_StudentNumber": "STU004", "ModuleMark": 78},

{"ResultID": 5, "Modules\_ModuleID": "ITB528", "Students\_StudentNumber": "STU005", "ModuleMark": 89}]);

{

acknowledged: true,

insertedIds: {

'0': ObjectId('6666cd29fa307ceca5cdce0b'),

'1': ObjectId('6666cd29fa307ceca5cdce0c'),

'2': ObjectId('6666cd29fa307ceca5cdce0d'),

'3': ObjectId('6666cd29fa307ceca5cdce0e')

}

}

## 2.5

db.Students.find().sort({"FName": 1})

[

{

\_id: ObjectId('6666bd8dfa307ceca5cdcdfd'),

StudentNumber: 'STU003',

FName: 'Dick',

LName: 'Romans',

Phone: '0612612343'

},

{

\_id: ObjectId('6666bd8dfa307ceca5cdcdff'),

StudentNumber: 'STU005',

FName: 'Isak',

LName: 'Pillay',

Phone: '0718543269'

},

{

\_id: ObjectId('6666bd8dfa307ceca5cdcdfe'),

StudentNumber: 'STU004',

FName: 'Jacob',

LName: 'White',

Phone: '0615234123'

},

{

\_id: ObjectId('6666bc76fa307ceca5cdcdfb'),

StudentNumber: 'STU001',

FName: 'James',

LName: 'Wright',

Phone: '0812682323'

},

{

\_id: ObjectId('6666bd8dfa307ceca5cdcdfc'),

StudentNumber: 'STU002',

FName: 'Lindsay',

LName: 'Coleman',

Phone: '0712682223'

}

]

## 2.6

db.Instructors.find().sort({"InstructorID": 1}, {"FName": -1})

[

{

\_id: ObjectId('6666bf6bfa307ceca5cdce00'),

InstructorID: 1,

Department\_DepID: 1,

FName: 'Wendy',

LName: 'Tonga',

Phone: '0825632486',

Address: '17 Bicker Street, Midrand, Gauteng'

},

{

\_id: ObjectId('6666c118fa307ceca5cdce01'),

InstructorID: 2,

Department\_DepID: 2,

FName: 'Jessica',

LName: 'Hill',

Phone: '0716542389',

Address: '1 Kobalt Street, Midrand, Gauteng'

},

{

\_id: ObjectId('6666c118fa307ceca5cdce02'),

InstructorID: 3,

Department\_DepID: 3,

FName: 'Phillip',

LName: 'Setonga',

Phone: '0612354789',

Address: '8 Nice Lane, Sandton, Gauteng'

},

{

\_id: ObjectId('6666c118fa307ceca5cdce03'),

InstructorID: 4,

Department\_DepID: 4,

FName: 'Lana',

LName: 'Jacobsons',

Phone: '0630253461',

Address: '65 Loner Road, Edenvale, Gauteng'

},

{

\_id: ObjectId('6666c118fa307ceca5cdce04'),

InstructorID: 5,

Department\_DepID: 5,

FName: 'Elias',

LName: 'Ngwena',

Phone: '0812357896',

Address: '100 CloutMain Avenue, Sandton, Gauteng'

}

]

## 2.7

db.Students.find( { FName: { $regex: /^A/i } } )

Tested but had no names starting with A so I tested the following for J:

EduEnrolment> db.Students.find( { FName: { $regex: /^J/i } } )

[

{

\_id: ObjectId('6666bc76fa307ceca5cdcdfb'),

StudentNumber: 'STU001',

FName: 'James',

LName: 'Wright',

Phone: '0812682323'

},

{

\_id: ObjectId('6666bd8dfa307ceca5cdcdfe'),

StudentNumber: 'STU004',

FName: 'Jacob',

LName: 'White',

Phone: '0615234123'

}

]

## 2.8

db.Modules.updateMany({"Instructors\_InstructorID": 1},{$set:{"ModuleName":"Introduction to C++"}})

{

acknowledged: true,

insertedId: null,

matchedCount: 1,

modifiedCount: 1,

upsertedCount: 0

}

db.Modules.find({"Instructors\_InstructorID":1})

[

{

\_id: ObjectId('6666c905fa307ceca5cdce05'),

ModuleID: 'CPP1513',

Instructors\_InstructorID: 1,

ModuleName: 'Introduction to C++',

NumberOfStudents: 15,

ExamDate: '2024-07-01'

}

]

## 2.9

db.Instructors.deleteOne({"InstructorID": 5})

{ acknowledged: true, deletedCount: 1 }

## 2.10

The collections of EduEnrolment has been added to the zip file.

# References

Coronel, C. & Morris, S., 2022. *Database Systems - Design, implementation, and management.* 14th ed. Boston: Cengage Learning, Inc..

MongoDB, 2024. *https://www.mongodb.com/.* [Online]   
Available at: https://www.mongodb.com/docs/manual/crud/  
[Accessed 18 June 2024].