Q1.

a.

CREATE TABLE Library (

Section VARCHAR(50),

BookID INT,

Book\_name VARCHAR(100),

Authors VARCHAR(100),

borrower\_ID INT,

borrower\_name VARCHAR(100),

Issue\_Date DATE,

Return\_Date DATE,

PRIMARY KEY (BookID, borrower\_ID, Issue\_Date)

);

b.

CREATE FUNCTION GetBookDetailsInRange(@BookID INT, @StartDate DATE, @EndDate DATE)

RETURNS TABLE

AS

RETURN (

SELECT Book\_name, COUNT(\*) AS IssuedCount

FROM Library

WHERE BookID = @BookID AND Issue\_Date BETWEEN @StartDate AND @EndDate

GROUP BY Book\_name

);

c.

CREATE FUNCTION GetBorrowerBookCount(@BookID INT, @Borrower\_ID INT, @Month INT, @Year INT)

RETURNS INT

AS

BEGIN

DECLARE @IssueMonth INT;

SET @IssueMonth = MONTH(GETDATE());

RETURN (

SELECT COUNT(\*)

FROM Library

WHERE BookID = @BookID AND borrower\_ID = @Borrower\_ID AND MONTH(Issue\_Date) = @Month AND YEAR(Issue\_Date) = @Year

);

END;

Q2.

a.

CREATE TABLE Students (

S\_ID INT PRIMARY KEY,

S\_Name VARCHAR(100),

S\_AN VARCHAR(12) UNIQUE,

S\_CA VARCHAR(100),

S\_PA VARCHAR(100),

S\_DOB DATE,

Sex VARCHAR(10),

Batch VARCHAR(20),

Department VARCHAR(50),

Program VARCHAR(20)

);

b.

CREATE TABLE Departments (

D\_code INT PRIMARY KEY,

D\_Name VARCHAR(100) UNIQUE,

D\_OA VARCHAR(100),

HOD\_name VARCHAR(100)

);

c.

CREATE TABLE Courses (

C\_num INT PRIMARY KEY,

C\_name VARCHAR(100),

D\_name VARCHAR(100),

C\_credit INT,

C\_program VARCHAR(20),

C\_batch VARCHAR(20)

);

d.

CREATE TABLE Classes (

class\_ID INT PRIMARY KEY,

I\_name VARCHAR(100),

I\_ID INT,

C\_ID INT,

Program VARCHAR(20),

Batch VARCHAR(20)

);

e.

CREATE TABLE GradeSheet (

StudentName VARCHAR(100),

StudentID INT,

Course\_ID INT,

Course\_name VARCHAR(100),

Program VARCHAR(20),

Batch VARCHAR(20),

Grade VARCHAR(2),

PRIMARY KEY (StudentID, Course\_ID)

);

2.(a)

CREATE TABLE Students (

S\_ID INT PRIMARY KEY,

S\_Name VARCHAR(100),

S\_AN VARCHAR(12) UNIQUE,

S\_CA VARCHAR(100),

S\_PA VARCHAR(100),

S\_DOB DATE,

Sex VARCHAR(10),

Batch VARCHAR(20),

Department VARCHAR(50),

Program VARCHAR(20)

);

CREATE TABLE Departments (

D\_code INT PRIMARY KEY,

D\_Name VARCHAR(100) UNIQUE,

D\_OA VARCHAR(100),

HOD\_name VARCHAR(100)

);

CREATE TABLE Courses (

C\_num INT PRIMARY KEY,

C\_name VARCHAR(100),

D\_name VARCHAR(100),

C\_credit INT,

C\_program VARCHAR(20),

C\_batch VARCHAR(20)

);

CREATE TABLE Classes (

class\_ID INT PRIMARY KEY,

I\_name VARCHAR(100),

I\_ID INT,

C\_ID INT,

Program VARCHAR(20),

Batch VARCHAR(20)

);

CREATE TABLE GradeSheet (

StudentName VARCHAR(100),

StudentID INT,

Course\_ID INT,

Course\_name VARCHAR(100),

Program VARCHAR(20),

Batch VARCHAR(20),

Grade VARCHAR(2),

PRIMARY KEY (StudentID, Course\_ID)

);

2(b) List of Functional Dependencies:

In the Students table:

S\_ID → {S\_Name, S\_AN, S\_CA, S\_PA, S\_DOB, Sex, Batch, Department, Program}

S\_AN → {S\_ID, S\_Name, S\_CA, S\_PA, S\_DOB, Sex, Batch, Department, Program}

In the Departments table:

D\_code → {D\_Name, D\_OA, HOD\_name}

D\_Name → {D\_code, D\_OA, HOD\_name}

In the Courses table:

C\_num → {C\_name, D\_name, C\_credit, C\_program, C\_batch}

In the Classes table:

class\_ID → {I\_name, I\_ID, C\_ID, Program, Batch}

In the GradeSheet table:

{StudentID, Course\_ID} → {StudentName, Course\_name, Program, Batch, Grade}

2(c) Relational Schema in 4NF:

For 4NF, we ensure no non-trivial multi-valued dependencies (MVDs) and no join dependencies (JDs):

Students table:

Primary Key: S\_ID

Prime Attributes: S\_ID

Non-Prime Attributes: S\_Name, S\_AN, S\_CA, S\_PA, S\_DOB, Sex, Batch, Department, Program

No non-trivial MVDs or JDs present, already in 4NF.

Departments table:

Primary Key: D\_code

Prime Attributes: D\_code

Non-Prime Attributes: D\_Name, D\_OA, HOD\_name

No non-trivial MVDs or JDs present, already in 4NF.

Courses table:

Primary Key: C\_num

Prime Attributes: C\_num

Non-Prime Attributes: C\_name, D\_name, C\_credit, C\_program, C\_batch

No non-trivial MVDs or JDs present, already in 4NF.

Classes table:

Primary Key: class\_ID

Prime Attributes: class\_ID

Non-Prime Attributes: I\_name, I\_ID, C\_ID, Program, Batch

No non-trivial MVDs or JDs present, already in 4NF.

GradeSheet table:

Primary Key: (StudentID, Course\_ID)

Prime Attributes: StudentID, Course\_ID

Non-Prime Attributes: StudentName, Course\_name, Program, Batch, Grade

No non-trivial MVDs or JDs present, already in 4NF.