

Relational Database Development

DAY 1 EXERCISES

25 MARKS

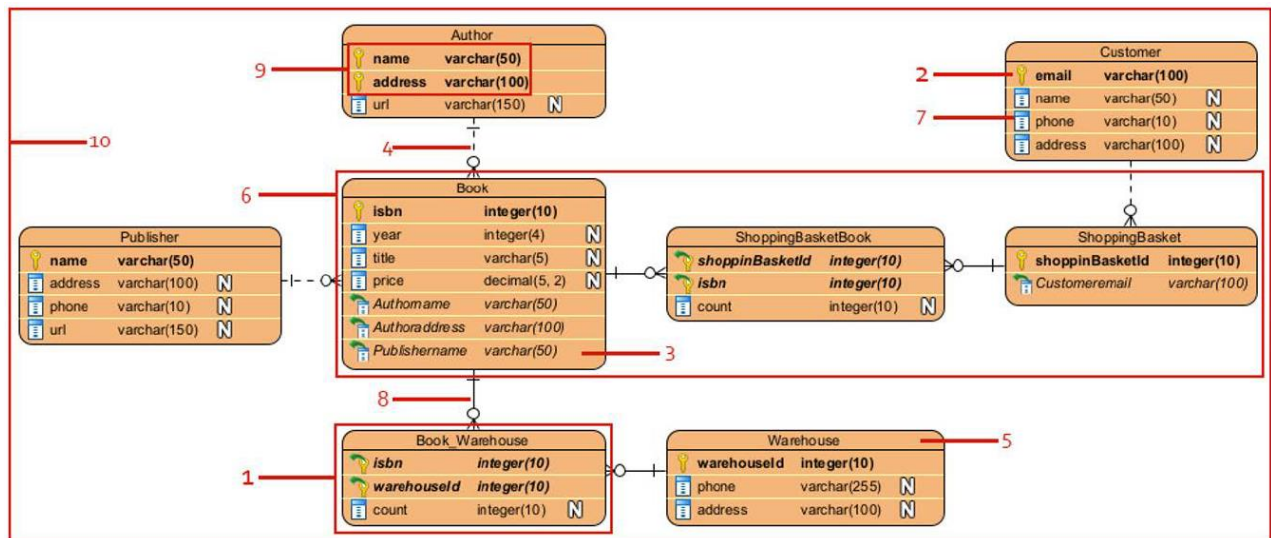
**Submission
Details:**

Please upload this document with your answers to the appropriate drop box.

**Late
Penalty:**

10% deducted each day this assignment is late so you can still submit late and get a decent mark within a reasonable time frame.

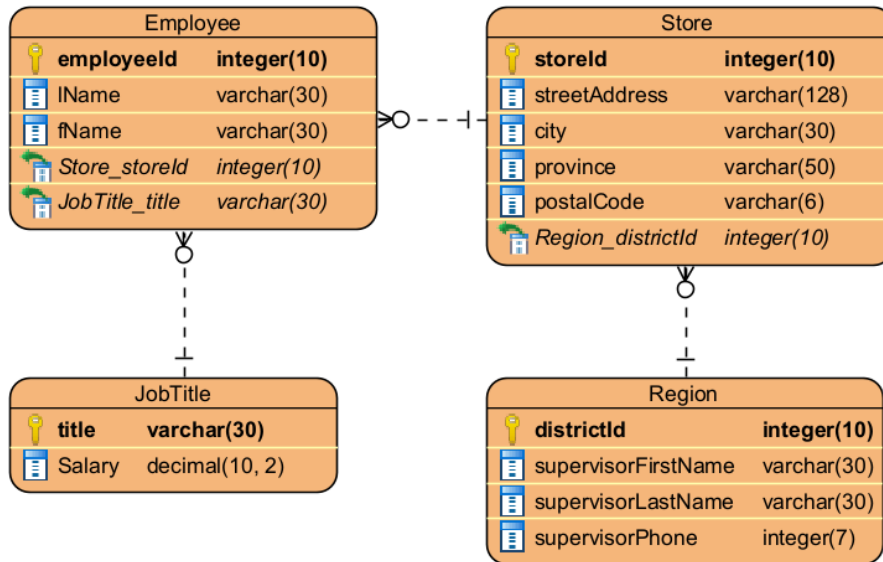
Exercise 1 (5 marks)



Match the numbers in the diagram above with the following terms:

- 7 Attribute.
- 1 Bridge Table.
- 5 Entity.
- 6 Many to Many relationship.
- 2 Primary Key.
- 8 Strong Relationship.
- 10 Schema.
- 9 Composite Key.
- 4 Weak Relationship.
- 3 Foreign Key.

Exercise 2 (3 marks)



In the boxes below, describe each relationship in the tables above. Both directions need to be included in your answer, like it was in today's Lesson.

Each JobTitle has zero to many Employees

Each Employee has one JobTitle

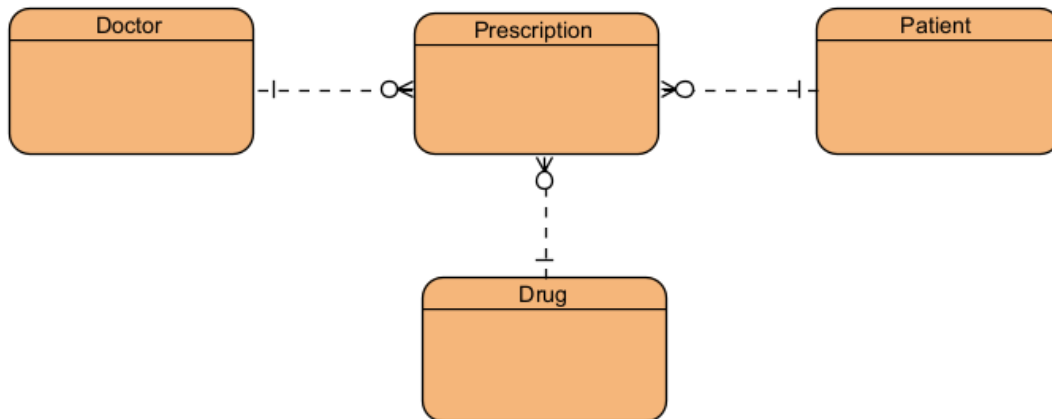
Each Employee has one Store

Each Store has zero to many Employees

Each Store has one Region

Each Region has zero to many Stores

Exercise 3 (3 marks)



The tables above don't have any attributes listed. In the boxes below, list suitable primary keys in each table.

Doctor:

pkDoctorID

Prescription:

pkPrescriptionID

Patient:

pkPatientID

Drug:

pkDrugID

Exercise 4 (4 marks)

Use Visual Paradigm to create an ERD using the following specifications.

Tables

Book, author, member, borrowed.

Table Attributes

Book: title, isbn, publication year, author Id (FK).

Author: first name, last name, nationality.

Member: first name, last name, address, email, phone.

Borrowed: member Id (FK), book Id (FK), borrow date, return date.

Primary Keys:

Book: book Id

Author: author Id

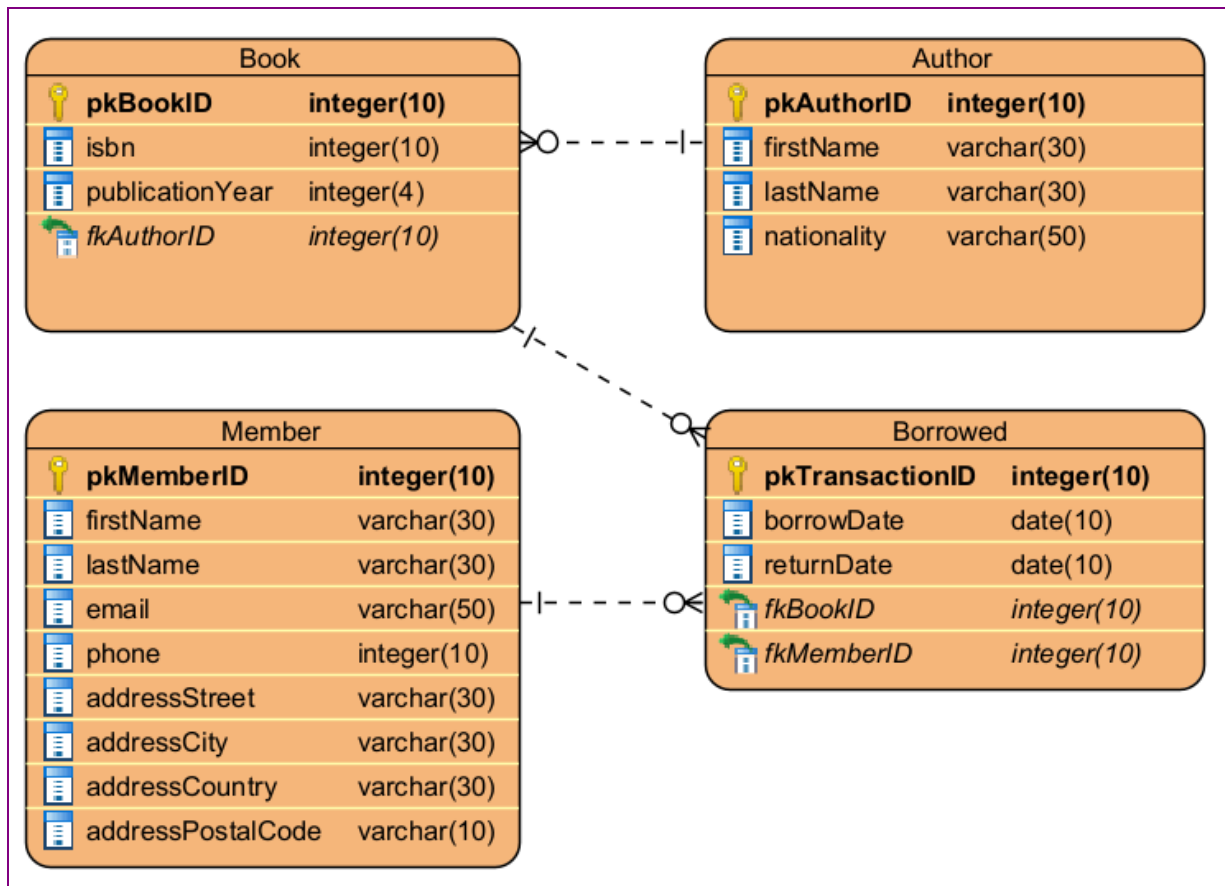
Member: member Id

Borrowed: transaction Id

Relationships

1. An Author can write multiple Books, but each Book has one Author.
2. A Member can borrow multiple Books, and each Book can be borrowed by different Members over time.
3. Each Borrowed transaction refers to a single Book and a single Member.

Place a screenshot of your finished ERD in the box below.



Exercise 5 (5 marks)

You've been asked to design a database system for a local health clinic. The clinic wants to keep track of their doctors, patients, and the appointments scheduled between doctors and patients. Use Visual Paradigm to create an ERD using the following specifications.

Tables

Doctor, patient, appointment.

Table Attributes

Represent all tables with their respective attributes and primary keys.

Relationships

Determine the type of relationships (e.g., one-to-many) between the tables and represent them using appropriate notation.

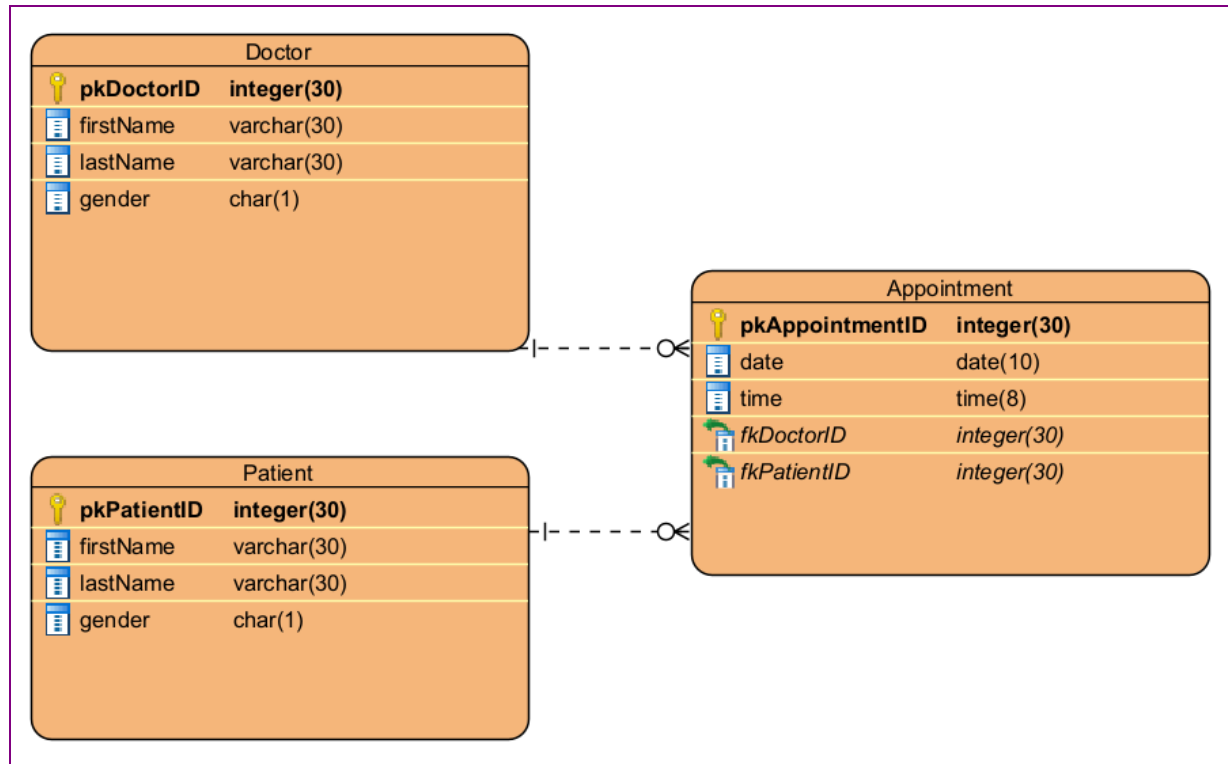
Doctors → Appointments Relationships

- One Doctor can have no appointments and many appointments.
- An appointment is associated with one and only one doctor.
- Considered as a one-to-many relationship from Doctors to Appointments.

Patients → Appointments Relationships

- One Patient can have no appointments and many appointments.
- An appointment is for one and only one patient.
- Considered as a one-to-many relationship from Patients to Appointments.

Place a screenshot of your finished ERD in the box below.



Exercise 6 (5 marks)

A mid-sized university is revamping its student registration system. Your job is to design the Entity-Relationship Diagram (ERD) for the new database.

Tool

Use Visual Paradigm to create the ERD.

Tables

student, professor, course and enrollment.

Table Attributes

Add the following attributes in addition to the primary keys and foreign keys. All attributes are not null.

student:

- first name (string)
- last name (string)
- date of birth (date)
- email (string)

professor:

- first name (string)
- last name (string)
- department (string)
- email (string)

course:

- name (string)
- credits (integer) start date (date)
- end date (date)

enrollment:

- enrollment date (date)
- grade (string)
- paid (char)

Relationships (crow's-foot notation)

Model the following with proper crow's-foot notation. Your design will eliminate the natural many-to-many between Student and Course by using Enrollment as the bridge.

student → enrollment relationship

- Each student can have none or many enrollments, but each enrollment belongs to exactly 1 student.
- The PK in the Students table will have a foreign key relationship to the Enrollment table.

course → enrollment relationship

- Each course can have none or many enrollments, but each enrollment belongs to exactly 1 course.
- The PK in the Courses table will have a foreign key relationship to the Enrollment table.

professor → course relationship

- One professor can teach none or multiple courses, but each course is taught by only one professor.
- The PK in the Professor table will have a foreign key relationship to the Course table.

Paste a screenshot of your ERD showing: entities, attributes, data types, keys, nullability, and relationship cardinalities in the box below.

