1-2:

EXERCISE 1:

1.
The school district should capture student personal information, enrollment information, guardian information, medical information, academic records, attendance, extracurriculars, and registration information

2.
XYZ should capture key information about the book, author, publisher, customers, reservations, staff members, book categories, library branches, and book condition tracking. This information should help with day to day operations and inventory management.

1-3:

EXERCISE 1:

1.

A. Hierarchical Model
B. Network Model
C. Object-Oriented Model
D. Relational Model
E. Flat File Model

1-4:

EXERCISE 1:

1.
LibBook has three different types of memberships; corporate, student, and individual. Corporate and individual require fees whereas student is free. Privileges depend on membership type. Changes require approval and payments must be made in advance.

2.
Doctor ID format starts with “DC.” They need to have a minimum 7 years of experience. Patient registration starts with “PT” and is required on the first visit. IDs need to be consistent and doctor experience requires verification. IDs should be unique.

2-1:

EXERCISE 1:

1.

BOOK [Book ID, Title, ISBN, Year, Price, Author ID, Publisher ID]
AUTHOR [Author ID, Name, Address, URL]
PUBLISHER [Publisher ID, Name, Address, Phone, URL]
WAREHOUSE [Warehouse ID, Book ID, Copies, Address, Phone]
CUSTOMER [Customer ID, Name, Address, Email, Phone]
SHOPPING CART [Shopping Cart ID, Customer ID]
ORDER [Order ID, Customer ID, Billing Address, Shipping Address, Shipping Option, Payment, Order Date]

2.
CUSTOMER [Customer ID, Name, Address, Phone, Email]
PRODUCT [Product ID, Product Name, Description, Quantity, Reorder, Supplier ID]
SUPPLIER [Supplier ID, Supplier Name, Address, Phone, Email, Alt Supplier]
ORDER [Order ID, Customer ID, Order Date, Status]
ORDER ITEM [Order Item ID, Order ID, Product ID, Qty Ordered, Qty Dispatched, Qty Backordered]
INVOICE [Invoice ID, Order ID, Invoice Date, Total Amount]

2-2:

EXERCISE 1:

1.
- Provides a high level overview that can be easily understood without getting technical
- Depicts business goals and requirements
- Can serve as a foundation for a detailed model
- Ensures consistency across different systems
- Acts as a communication tool between non-technical and technical stakeholders

2.
- An Entity-Relationship Diagram (ERD) and Unified Modeling Language (UML) diagrams are examples of conceptual diagrams.
- Database tables or schemas are examples of physical models

2-3:

EXERCISE 1:

1.
The following entities can be created for a School Management System
- Course
- Students
- Faculty

- Parent
- Departments
- Academic Session
- Exam

EXERCISE 2:

1.
COURSE:
* Course ID
* Course Name
* Department ID
* Credits
* Session ID
o Course Description

DEPARTMENT:
* Department ID
* Department Name
* School ID
* Head of Department
o Contact Information

STUDENT:
* Student ID
* First Name
* Last Name
* Address
* Contact Information
* Date of Birth
* Department ID
* Parent ID
* Attendance
o Email
o Middle Name

FACULTY:
* Faculty ID
* Faculties Name
* Department ID
* Faculty Login Time

* Faculty Logout Time
o Contact Information
o Email

EXAM:
* Exam ID
* Course ID
* Exam Date
* Exam Type

ACADEMIC SESSION:
* Session ID
* Session Start Date
* Session End Date
* Session Name

PARENT INFORMATION:
* Parent ID
* Student ID
* Parents Name
* Contact Information
o Email

2-4:

EXERCISE 1:

1.
A unique identifier for SONG ID can be used to look up the title of the song, the artist, or the album.

2.
Student ID, Full Name, or Email.

3.
[Student] Student ID
[Movie] Title and Date Released
[Locker] Number

EXERCISE 2:

1.

a. Unique Identifiers:
   - Course ID
   - Student ID
   - Department ID
   - Faculty ID
   - Exam ID
   - Session ID
   - Parent ID

a. Candidate Unique Identifiers:
   - Course Name
   - Student Name
   - Department Name
   - Parent Name
   - Session Name
   - Faculty Name

2-5:

EXERCISE 1:

1.

B. Each EMPLOYEE must be assigned to one and only one DEPARTMENT
Each DEPARTMENT must be responsible for one or more EMPLOYEES

2.

A PERSON can be born in one or more TOWNs.
A TOWN may be the birthplace of one and only one PERSON.

Multiple PERSONs can be living in one and only one TOWN.
A TOWN may be the hometown of multiple PERSONs.

Multiple PERSONs may be the visitor of multiple TOWNs.
Multiple TOWNs must be visited by multiple PERSONs.

A PERSON may be the mayor of only one TOWN.
A TOWN may be governed by only one PERSON

EXERCISE 2:

1.

FACULTY to COURSE: One (1) to Many (*), mandatory.

COURSE to FACULTY: Many (*) to One (1), mandatory.

STUDENT to COURSE: One (1) to Many (*), mandatory.

COURSE to STUDENT: Many (*) to One (1), mandatory.

PARENT INFORMATION to STUDENT: One (1) to Many (*), optional.

STUDENT to PARENT INFORMATION: Many (*) to One (1), mandatory.

FACULTY to DEPARTMENT: One (1) to Many (*), mandatory.

DEPARTMENT to FACULTY: Many (*) to One (1), mandatory.

DEPARTMENT to COURSE: One (1) to Many (*), mandatory.

COURSE to DEPARTMENT: Many (*) to One (1), mandatory.

STUDENT to EXAM: One (1) to Many (*), mandatory.

EXAM to STUDENT: Many (*) to One (1), mandatory.

COURSE to EXAM: One (1) to Many (*), mandatory.

EXAM to COURSE: Many (*) to One (1), mandatory.

ACADEMIC SESSION to COURSE: One (1) to Many (*), mandatory.

COURSE to ACADEMIC SESSION: Many (*) to One (1), mandatory.

2-6:

1.

Department
- * Department Name
- * Supervisor ID
- *Department Number

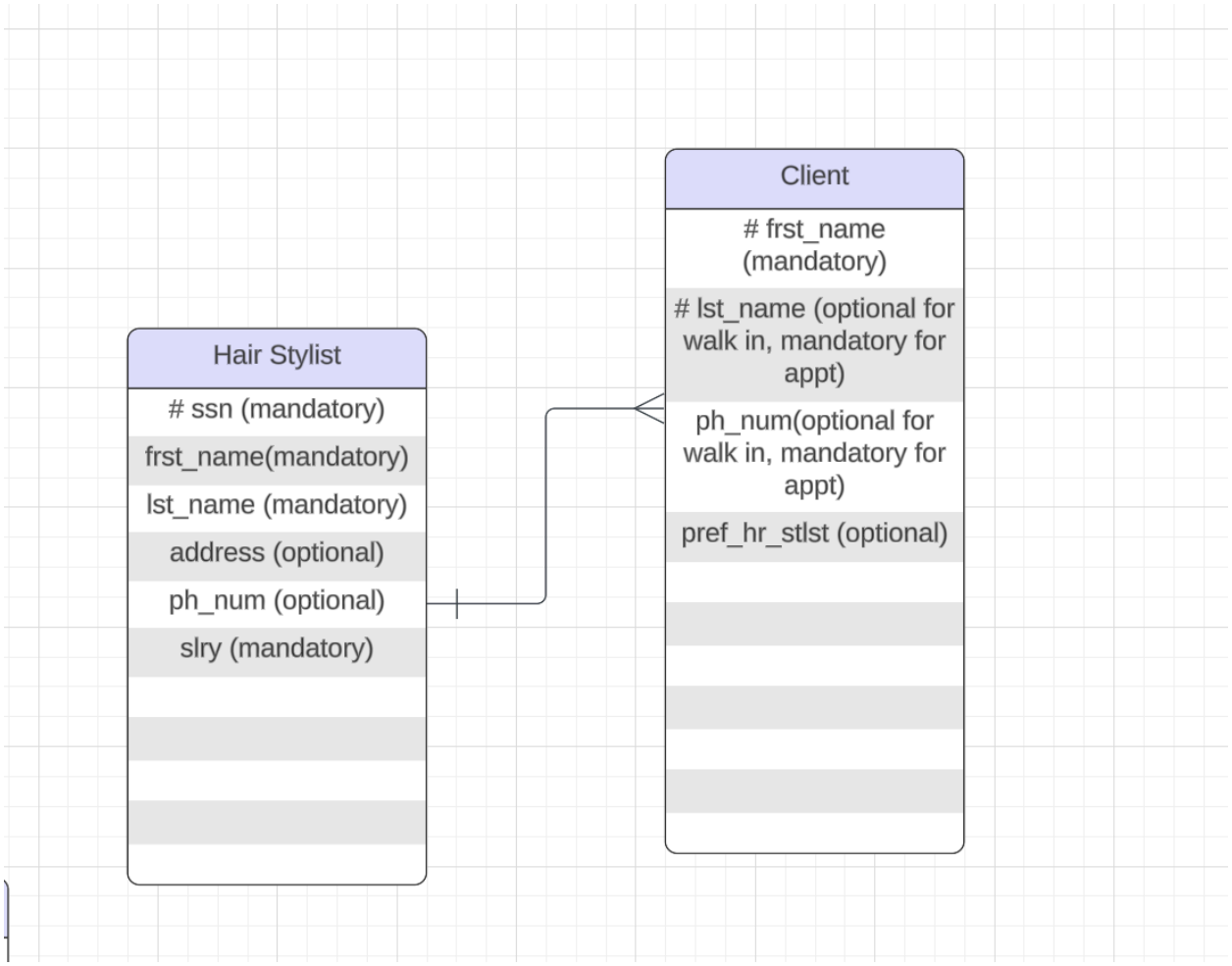
Employee
- * Employee ID

- * Employee Name
- o Vacation status

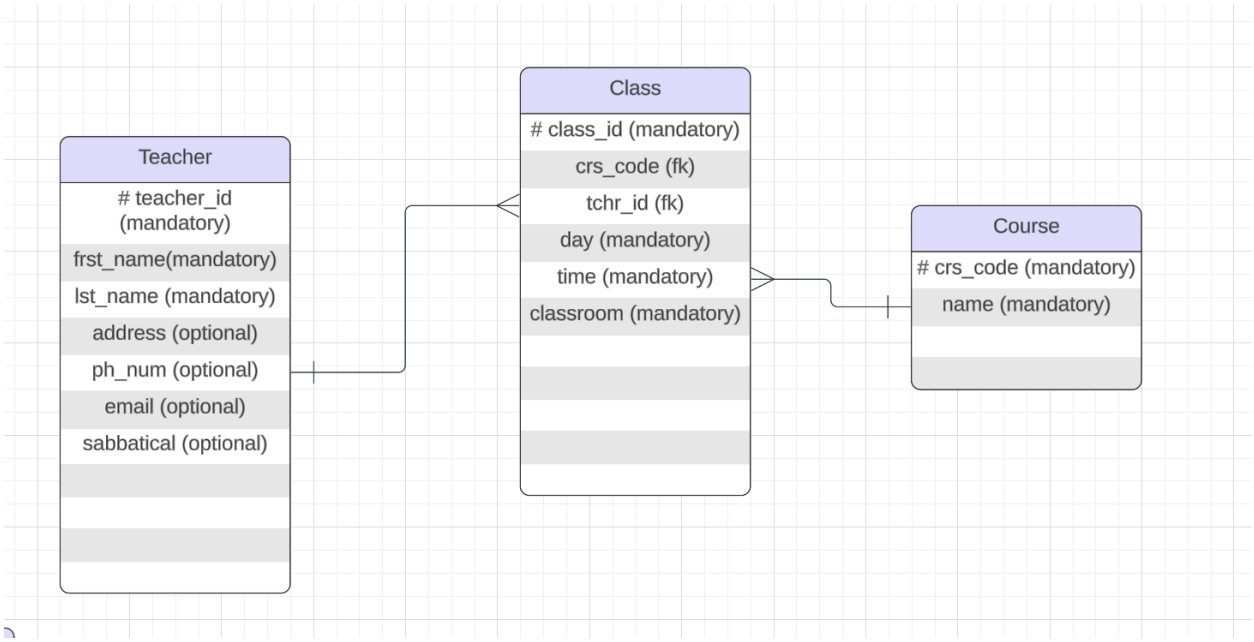
Project

- * Project ID
- * Project Name

2.



3.



3-1:

Exercise 1:

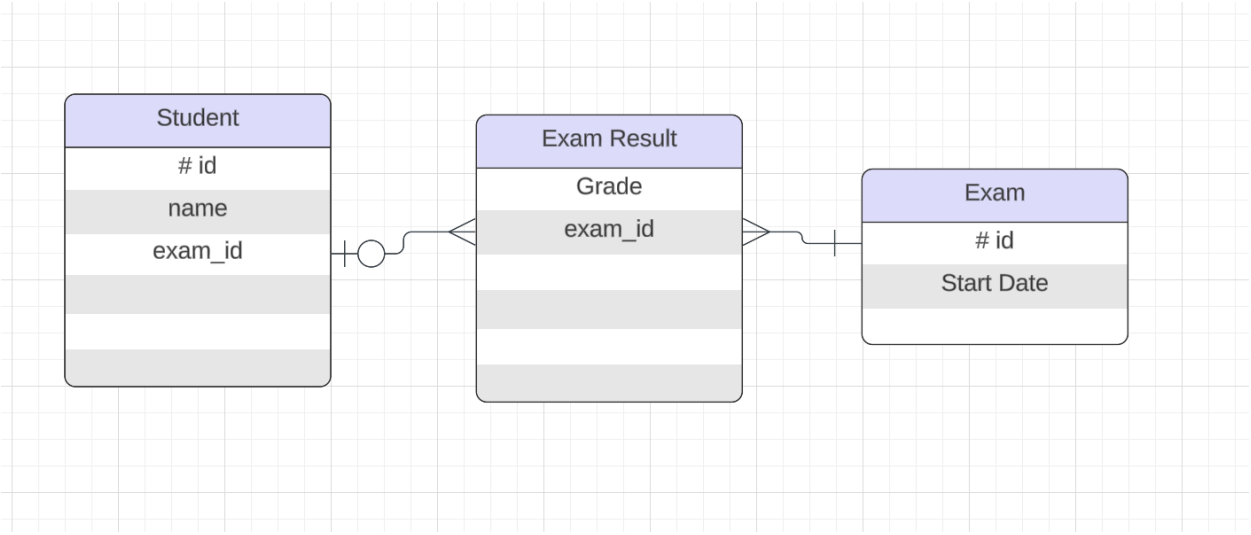
1.
We can create another table that holds Student ID and Course ID foreign keys. This will ensure that each student-course pairing is unique.

2.
We can create another table that holds Course ID and Faculty ID foreign keys. This will ensure that each faculty-course pairing is unique.

3.
We will create two additional tables for Student_Course and Student_Exam. The Student_Course table would have Student ID and Course ID as foreign keys. The Student_Exam table would have Student ID and Exam ID as foreign keys. We will then be able to ensure that a single student can enroll in multiple courses and take multiple exams. Likewise, multiple students can enroll in the same course and exam

EXERCISE 2:

1.



EXERCISE 3

1.
Faculty (Supertype)

- ID
- First Name
- Last Name
- Email
- Login Date
- Login Time
- Details

FullTimeFaculty (Subtype)

- Salary
- Insurance Plan

PartTimeFaculty (Subtype)

- Hourly Rate

EXERCISE 4:

1.
Exclusive Relationship can be modeled by having **Course** as the main entity and **Seated** and **Online** as the subtype entities. Each course can only be in either of the locations.

Course (supertype)

- Course ID
- Course Name

Seated (subtype)

- Building Name
- Room Number
- Date/Time Offered

Online (subtype)

- Logon ID
- Password

EXERCISE 5:

1.
- A hotel will have many floors
- Each floor can have many suites
- Each suite can have many rooms
- The unique identifiers can be: Hotel ID, Floor ID, Suite ID, and Room ID

Hotel

- **Attributes:**
  - Hotel ID (Unique Identifier)
  - Hotel Name
  - Address
  - Phone Number
  - Email

Floor

- **Attributes:**
  - Floor ID (Unique Identifier)
  - Floor Number
  - Hotel ID (Foreign Key, references Hotel)

Suite

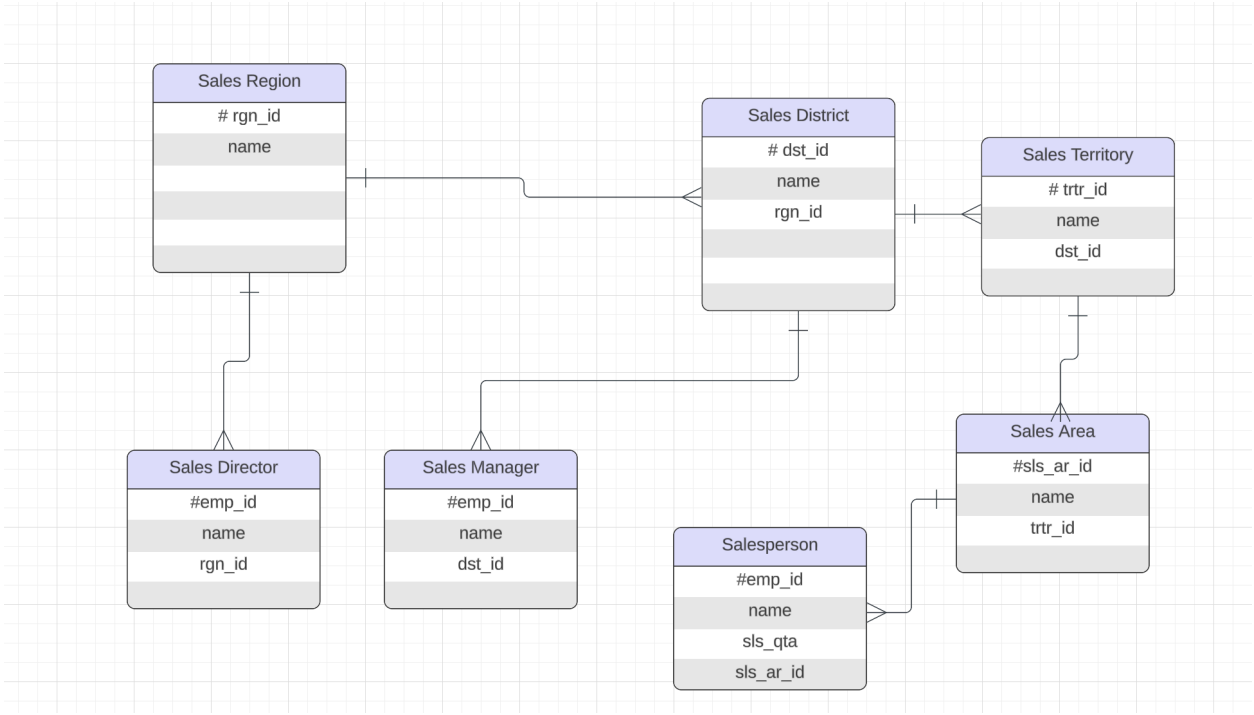
- **Attributes:**
  - Suite ID (Unique Identifier)
  - Suite Number
  - Floor ID (Foreign Key, references Floor)

Room

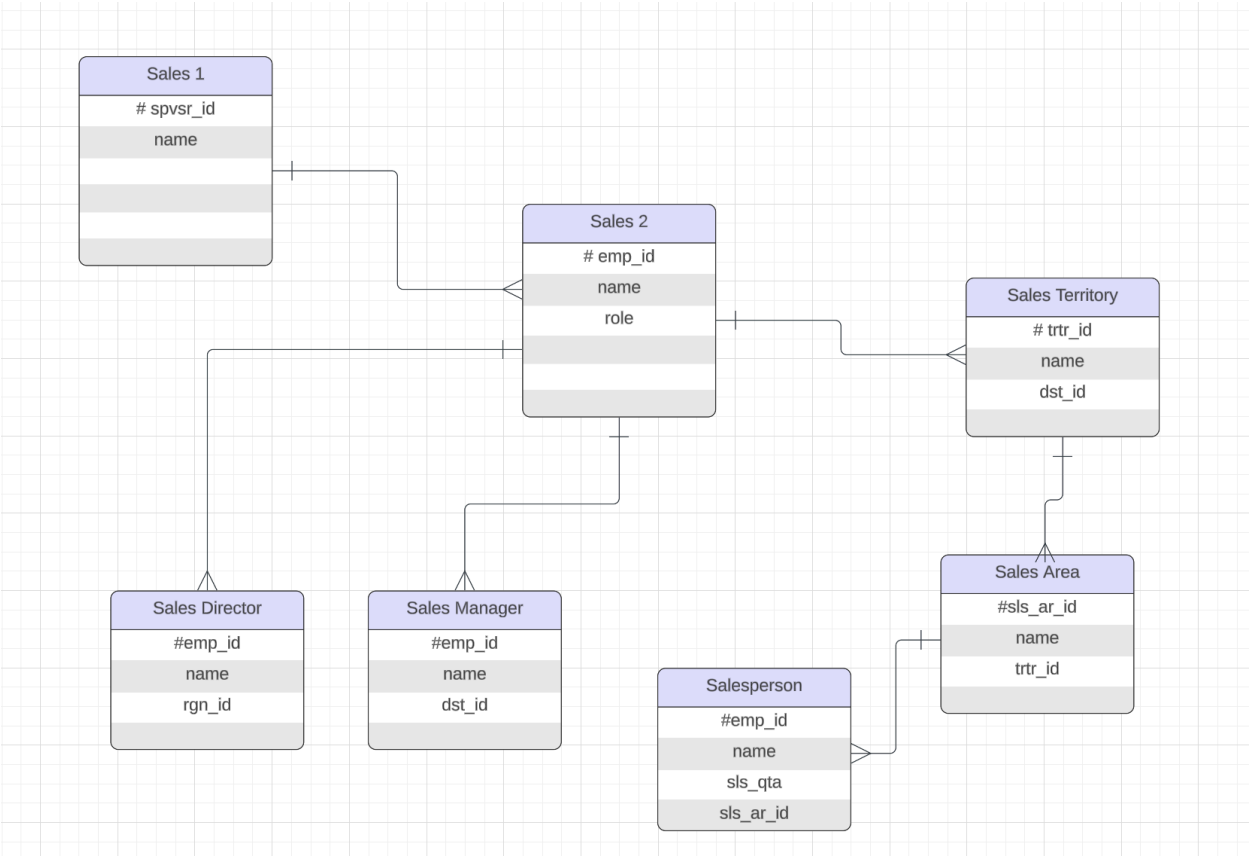
- **Attributes:**
  - Room ID (Unique Identifier)
  - Room Number
  - Suite ID (Foreign Key, references Suite)
  - Room Type (e.g., Single, Double, Suite)

EXERCISE 6:

1.
Hierarchical structure

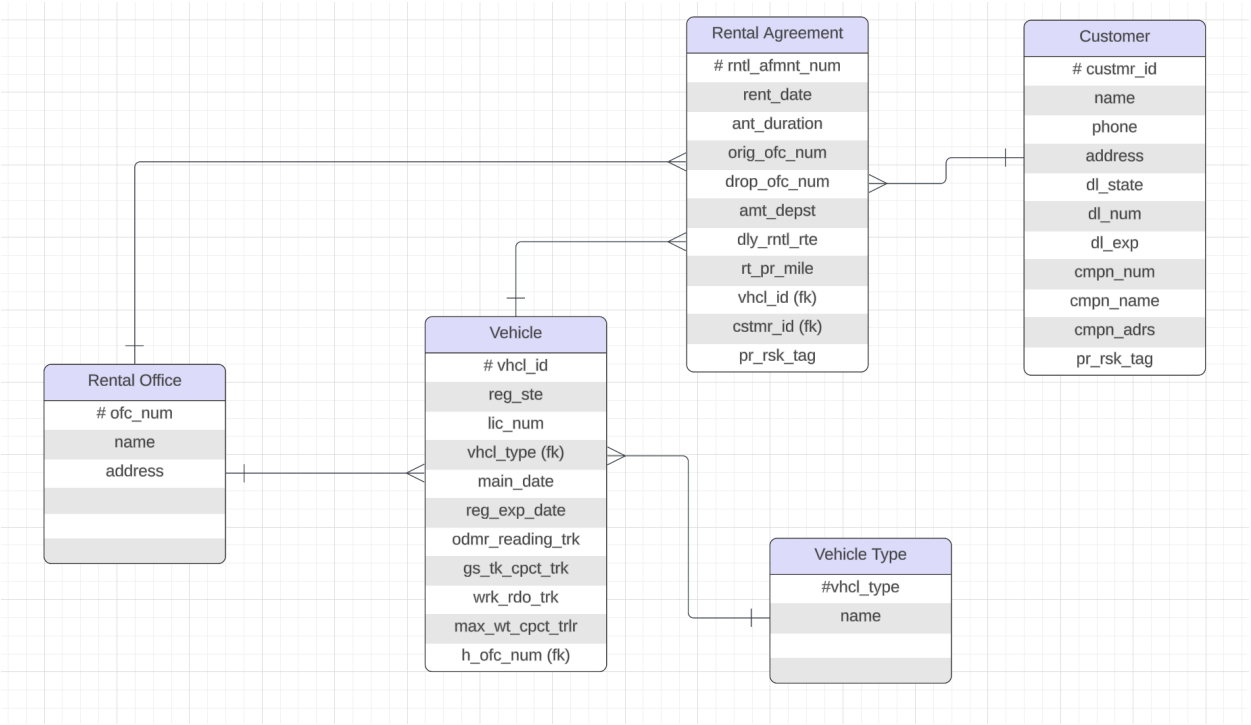


Recursive structure


Sales 1
spvsr_id
name
Sales 2
emp_id
name
role
Sales Territory
trtr_id
name
dst_id
Sales Director
#emp_id
name
rgn_id
Sales Manager
#emp_id
name
dst_id
Sales Area
#sls_ar_id
name
trtr_id
Salesperson
#emp_id
name
sls_qta
sls_ar_id

EXERCISE 7:

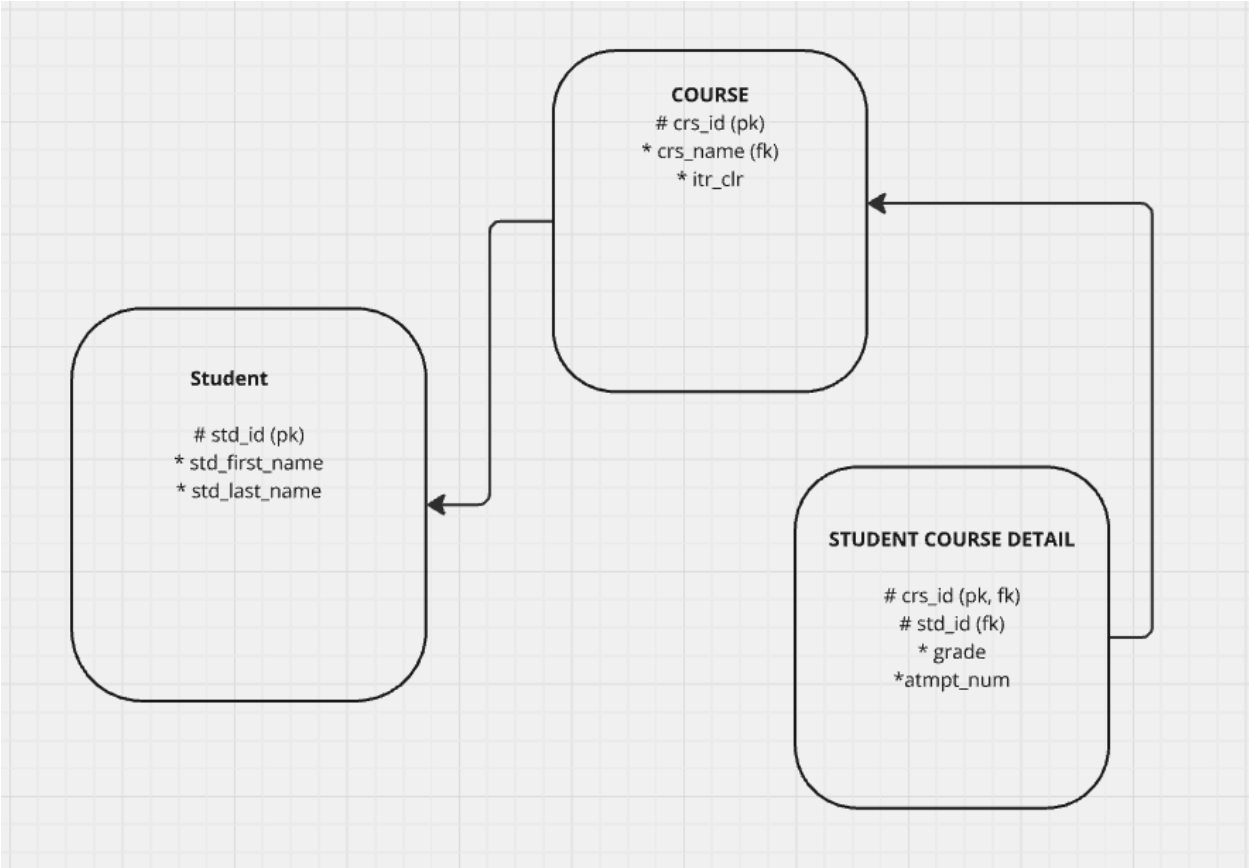
1.



3.2

EXERCISE 1:

1.
Adding a new attribute called **attempt_num** under Student Course Detail would help determine the number of attempts a student has taken and the grade received.



2.

a. Start Time is part of UID of ASSIGNMENT because multiple exams could fall under the same date.

b.

i. You need to ensure that start time does not overlap with another exam for the same class. This represent conditional non-transferability because it makes sure a class is not double booked

ii. The exam date should be within the academic term. This represent conditional non-transferability because exams should occur within a valid period

iii. End time must be later than start time. This represent conditional non-transferability because it is logically consistent.

3.3

Exercise 1:

1.

Item ID	Color	Unit Price
IT001	Red	$16.56

IT001	Blue	$16.56
IT002	Yellow	$17.48
IT003	Green	$18.76
IT004	Blue	$20.00
IT004	Yellow	$20.00

2.

Supplier Table

Supplier ID	**Store ID**
SP001	S1
SP001	S3
SP002	S1
SP003	S2
SP004	S3

Store Table

Store ID	**Location**
S1	New York
S3	Vermont
S1	New Hampshire
S2	Rhode Island
S3	Illinois

3.

Book Table

Book ID	**Category ID**	**Price**
1	1	$27.99
2	2	$17.99

3	1	$20.99
4	3	$40.99
5	2	$19.99

Category Table

Category ID	**Category Description**
1	Cooking
2	Travel
1	Cooking
3	Computers
2	Travel

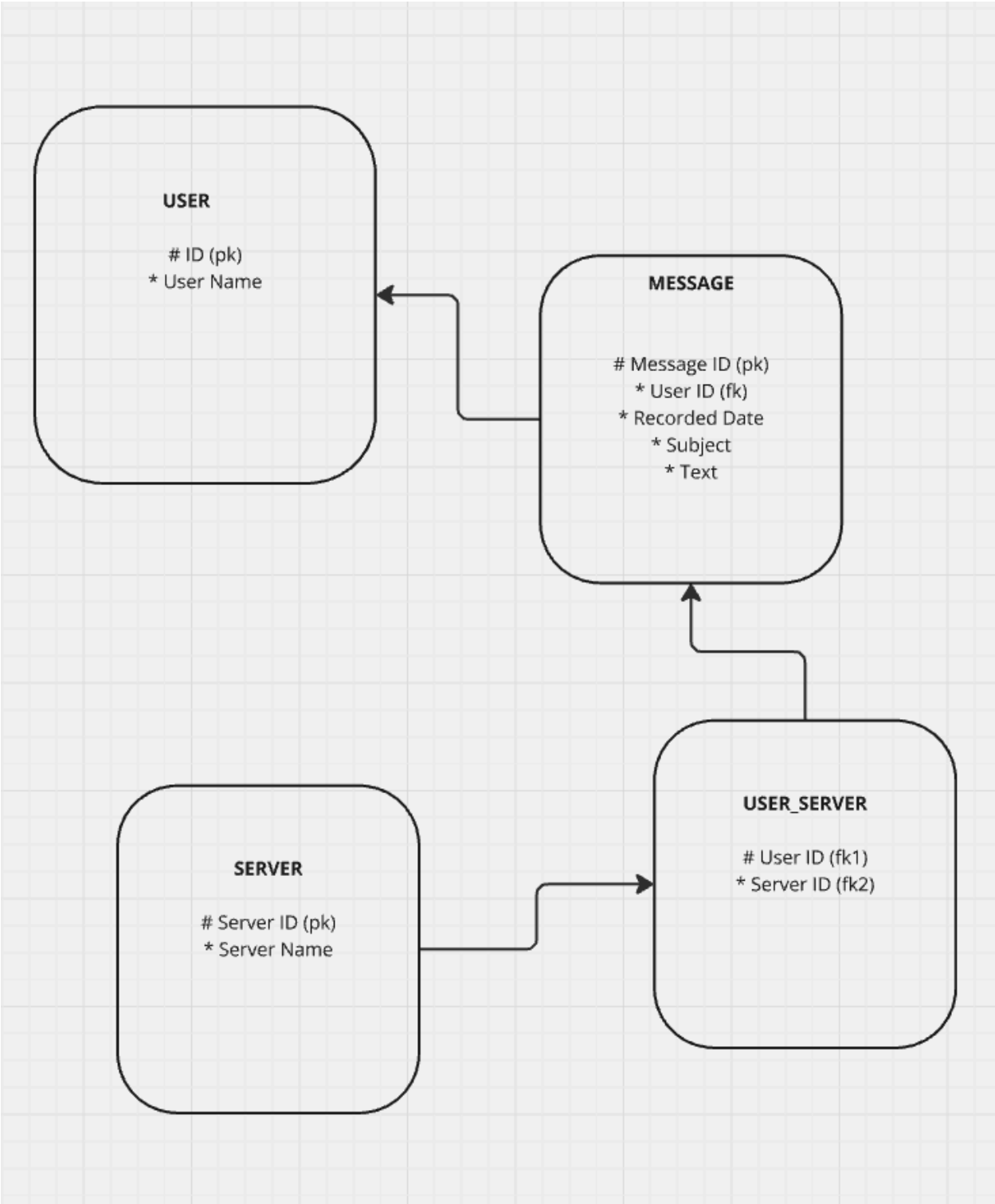
Exercise 2:

1.

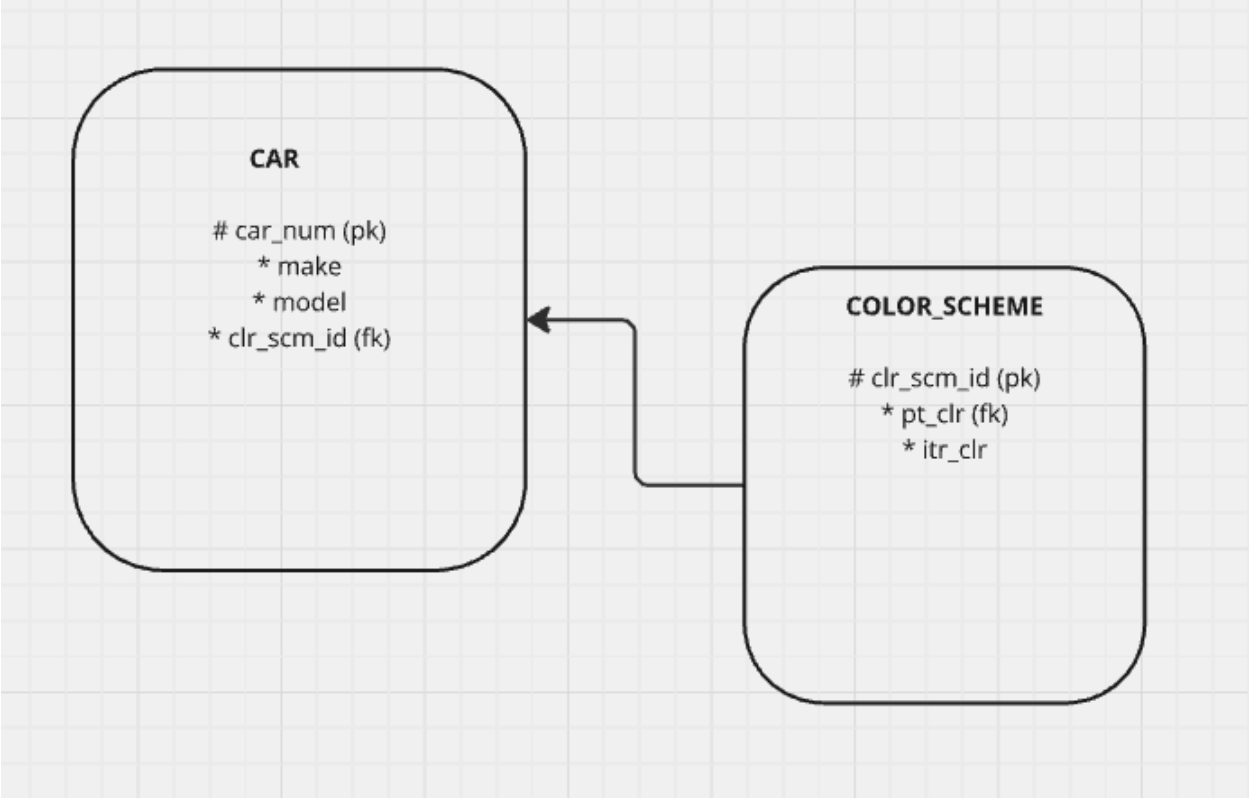
- COURSE should belong to ACADEMIC SESSION
- There should be another table that has the number of working days, number of days off and the eligibility for exam
- Parent 2 First Name and Last Name should not be optional attributes
- Login Date/Time details should be a different table
- Exam Type should be a different table

Exercise 3:

1.



2.
A separate color scheme table is needed because it would make sense for paint color and interior color to be dependent on color scheme.



EXERCISE 4:

1.
The following business rules can be implemented:

- A book should have a unique title, isbn, year, and price
- A book should be associate with one or more authors
- An author should have unique name and contact information
- A publisher should also have unique name and contact information
- Warehouses should have unique IDs and can have different books
- A book can be in multiple warehouses
- Customers should have unique contact details
- Customers can have multiple shopping carts which can contain multiple books. Quantity should be recorded
- Billing and shipping information should be required to complete purchase
- Email notifications should be sent after purchase

2.

Business Rule	Structural Business Rule	Procedural Business Rule	Programmatic Business Rule
All teachers in our school must possess a valid teaching certificate	X		
Each Department must offer a Course	X		
Approval of travel requests to an event must be signed by the project manager of the event		X	
A customer may make numerous payments on account	X		
A machine operator may not work more than 10 hours in a day		X	
The Rental amount in RENTAL is calculated from the Rental rate multiplied by the number of days			X
A Customer can have zero, one or many ORDERS	X		
The Total cost of the RENTAL is calculated from the sum of Insurance amount, Rental amount, and Late charge			X
A customer's debt must not exceed the customer's credit limit.			X

3.4

EXERCISE 1:

1. Match the ERD elements to their corresponding data elements.

Analysis	Design
Attribute	Column
Entity	Table
ER Model	Physical design
Instance	Row
Primary UID	Primary Key
Relationship	Foreign Key
Secondary UID	Unique Key

2.

a. Primary Key
b. Foreign Key
c. Unique Key
d. Primary Key
e. Optional attribute

3.

a. ATS
b. PLS
c. CTMS

4.

SONG	EVENT	CUSTOMER	
X	X		Title
X	X		Description
X	X		Venue
		X	First Name

X	X	X	Phone Number
X	X		Release Date
		X	Last Name
X	X		Type
	X	X	Email Address

EXERCISE 2:

PARENT INFORMATION

Key Type	Optionality	Parent Information
pk		id
	*	p1_first_name
	*	p1_last_name
	*	p2_first_name
	*	p2_last_name

EXAM TYPE

Key Type	Optionality	Exam Type
pk	*	id
	*	exam_name
	*	exam_dsc

DEPARTMENT

Key Type	Optionality	Department
pk		id
	*	name
	*	head
fk1	*	crs_id
	*	crs_name

STUDENT

Key Type	Optionality	Student
pk	*	id
	*	first_name
	*	last_name
	*	rgtn_yr
	*	email
fk1	*	prt_id
	o	p1_first_name
	o	p1_last_name

FACULTY

Key Type	Optionality	Parent Information
pk		id
	*	first_name
	*	last_name
	*	email
	*	salary
	o	ins_pln
	*	hrly_rt
fk1	*	dpt_id
	*	dpt_name
	o	dpt_head

COURSE

Key Type	Optionality	Course
pk	*	id
	*	name
fk1	*	dept_id
	*	dept_name
fk2	*	acdmc_ssn_id

ACADEMIC SESSION

Key Type	Optionality	Academic Session
pk	*	id
	*	name