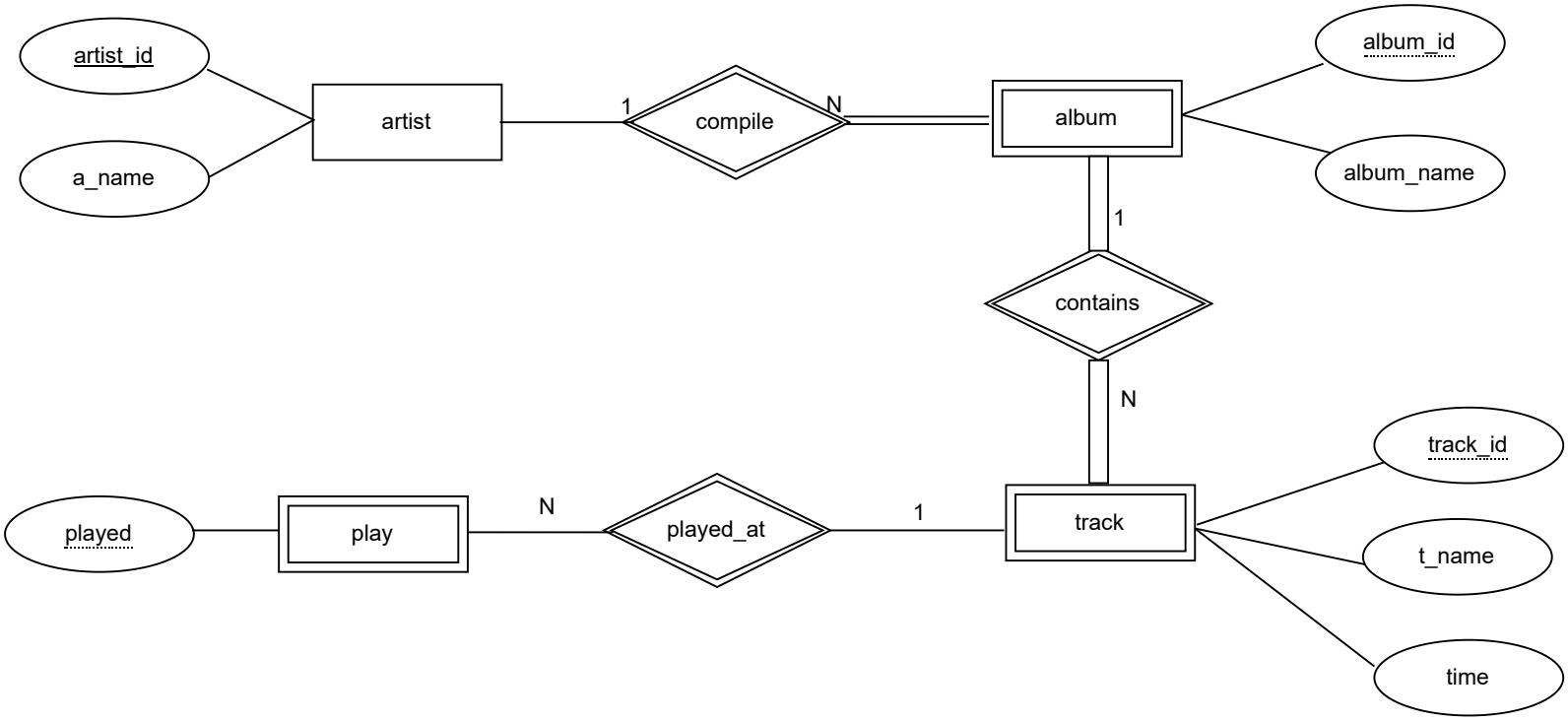
NAME - JATAN SAHU
STUDENT ID - 202218061

IT667 - Database Management Systems
Lab Assignment 7 - Database Design

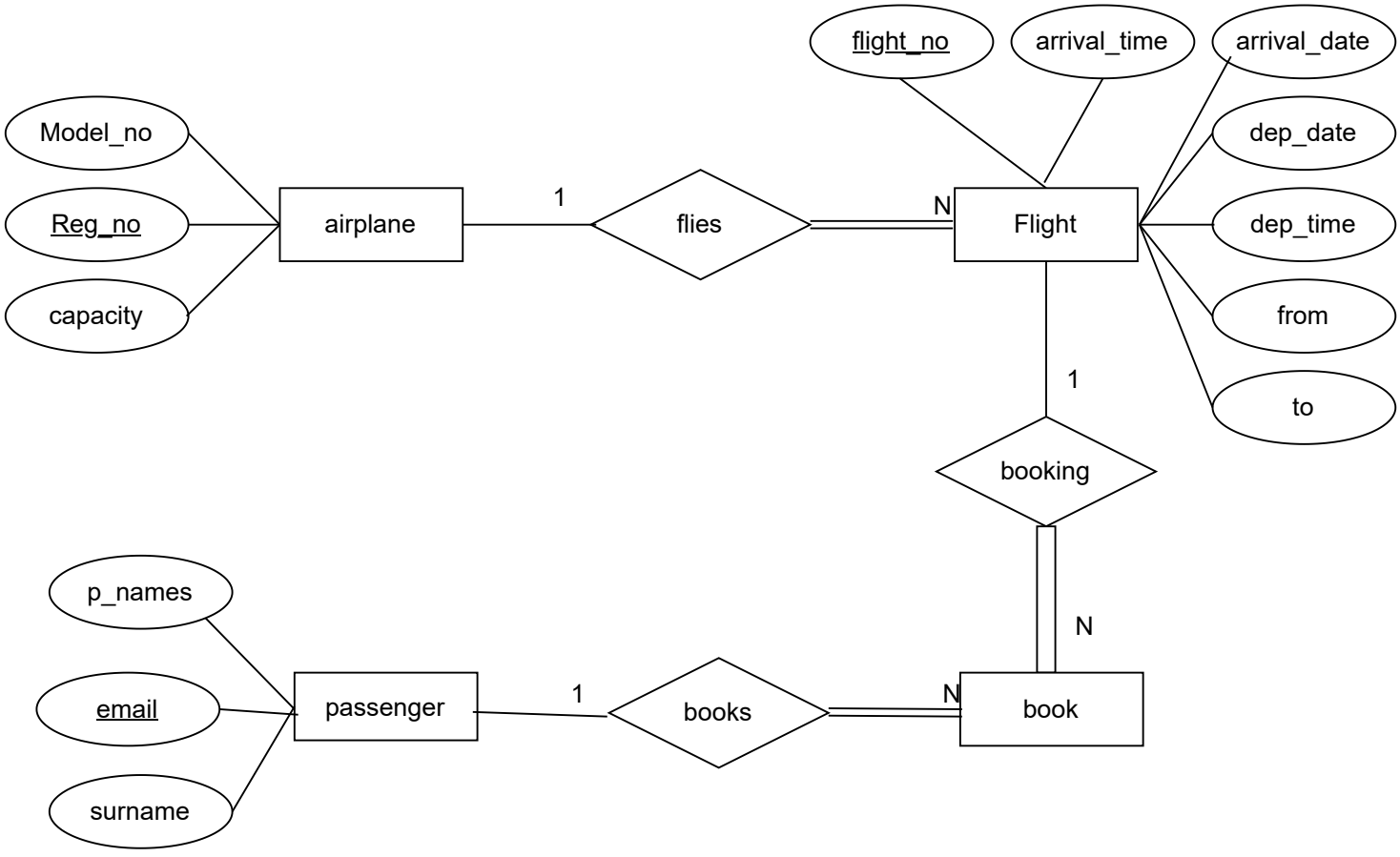
1. Construct an ER Diagram for the following system description:
Note - Illustrate the cardinality as well

a. A music database stores relationships between artists, albums, and tracks. The following
requirements must be satisfied:
- The collection consists of several albums.
- An album is composed by exactly one artist.
- An artist makes one or multiple albums.
- An album can contain more than one tracks
- Artists, albums, and tracks each have a name.
- Each track is on one album only.
- Each track has a time length, measured in seconds.
- When a track is played, the date and time the playback began should be recorded.

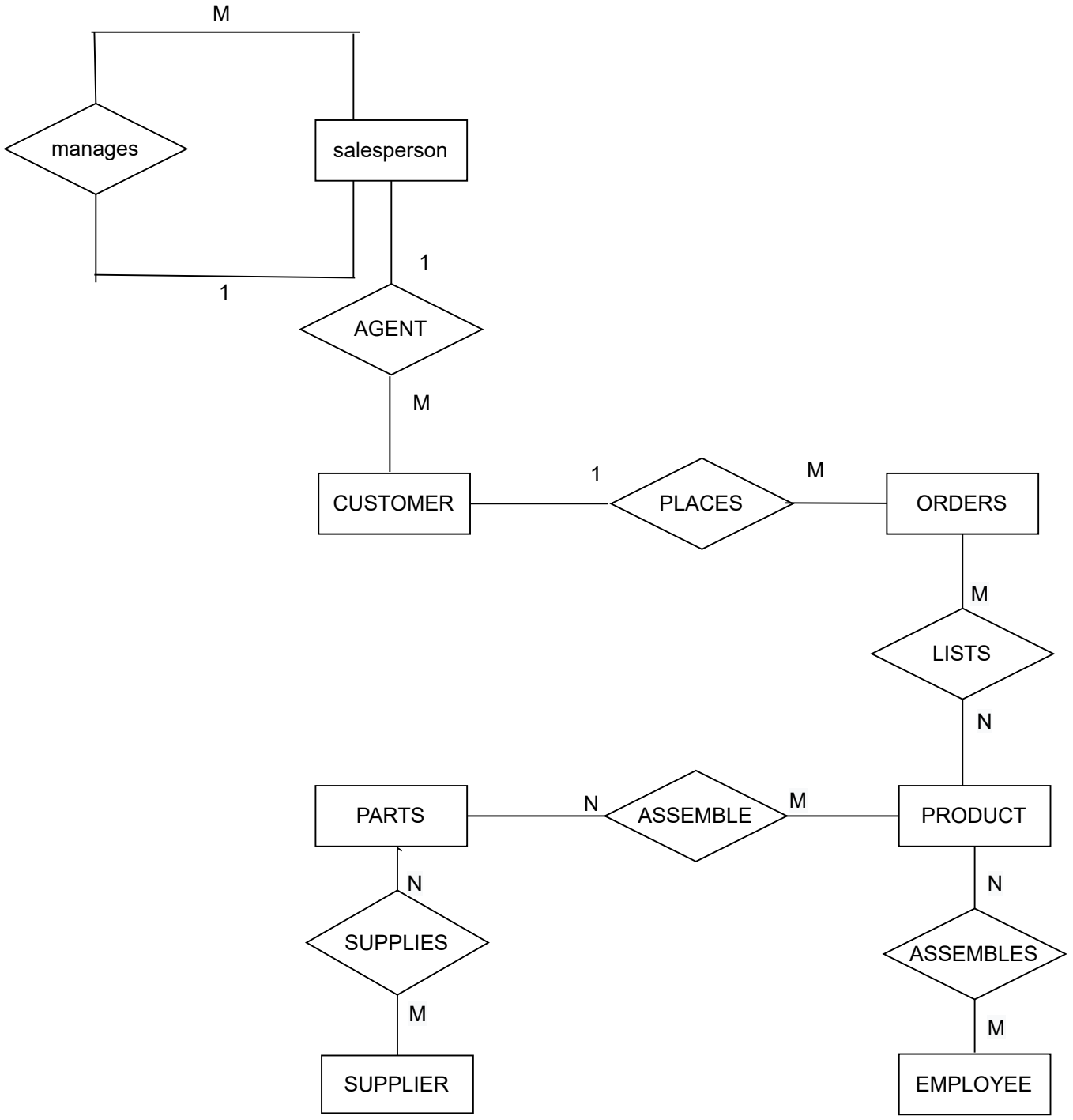


b. A flight database stores details about an airline’s data including flight and bookings. The
following requirements must be satisfied:

- The airline can have more than airplanes.
- An airplane has a model_number, an unique registration_number, and the capacity to
take passengers.
- An airplane flight has a unique flight_number, a departure_airport, a destination_airport,
a departure_date and departure_time, and an arrival_date and arrival_time.
- Each flight is carried out by a single airplane.
- A passenger has given names, a contact_number, and a unique email_address.
- A passenger can book a seat on a flight.



c. A company has the following scenario:
● There are several salespersons.
● Some of the salespersons manage other salespersons.
● A salesperson cannot have more than one manager.
● A salesperson can be an agent for multiple customers.
● A customer is managed by exactly one salesperson.
● A customer can place multiple orders.
● An order can be placed by exactly one customer.
● Each order contains one or more products.
● A product may be listed in many orders.
● A product is assembled from different parts and parts can be common for many products.
● One or more employees assemble a product from parts.
● A supplier can supply different parts in certain quantities.
● A part can be supplied by different suppliers.



2. For the table given below, using the concept of Normalization, Normalize the table to 1NF and
then 2NF. With the help of the table, state the conditions for 1NF and 2NF as well. Further, if possible, normalize it to 3NF, stating the conditions as well.

ORDER_ID	ORDER_DATE	C_ID	C_NAME	C_STATE	P_ID	P_NAME	P_PRICE	P_QUANTITY
101	1/10/22	1	ALICE	LA	3,7,6	PEN, PAPER, PENCIL	50, 100, 20	1,2,1
102	9/10/22	2	BOB	CA	5,6	SCALE, PENCIL	10, 20	4,5

TABLE TO 1 NF

- Every column in the table must be unique
- Separate tables must be created for each set of related data
- Each table must be identified with a unique column or concatenated columns
- called the primary key
- No rows may be duplicated
- no columns may be duplicated
- no row/column intersections contain a null value
- no row/column intersections contain multivalued fields

ORDER_ID	ORDER_DATE	C_ID	C_NAME	C_STATE	P_ID	P_NAME	P_PRICE	P_QUANT
101	1/10/2022	1	ALICE	LA	3	PEN	50	1
101	1/10/2022	1	ALICE	LA	7	PAPER	100	2
101	1/10/2022	1	ALICE	LA	6	PENCIL	20	1
102	9/10/2022	2	BOB	CA	5	SCALE	10	4
102	9/10/2022	2	BOB	CA	6	PENCIL	20	5

TABLE 1NF TO 2NF

Conditions:-

- Relation should be in 1NF and
- All the attributes of the non-primary keys are fully functional dependent on candidate key i.e. **no partial dependency**

Order_ID -> Order_Date ⟶ does not satisfy 2NF conditions
C_ID -> C_Name ⟶ satisfies
P_ID -> P_Name ⟶ does not satisfy
P_ID -> P_Price ⟶ does not satisfy
Order_ID -> C_ID ⟶ does not satisfy
C_ID -> C_State ⟶ satisfies
P_ID and Order_ID -> P_Quantity ⟶ satisfies

here **CANDIDATE KEY :**
order_id-p_id
(composite key)

PRIME ATTRIBUTES:
order_id,p_id
NON-PRIME ATTRIBUTES:
order date,c_id,c name,c state,p name,p price,p quantity

R1(Order_id , P_ID, C_ID, C_name, C_state , P_quantity**)**

Order_id	**P_ID**	C_ID	C_name	C_state	P_quantity
101	3	1	Alice	LA	1
101	7	1	Alice	LA	2
101	6	1	Alice	LA	1
102	5	2	Bob	CA	4
102	6	2	Bob	CA	5

R2(Order_ID , Order_date , C_ID**)**

Order_ID	Order_date	C_ID
101	1/10/22	1
102	9/10/22	2

R3(P_ID , P_name , P_price**)**

P_ID	P_Name	P_price
3	pen	50
7	paper	100
6	pencil	20
5	scale	10

2NF to 3NF

CONDITIONS:

- Relation should be in 2NF
- There should be no transitive dependency.

non-prime attribute is determined by non-prime attribute only.

here **CANDIDATE KEY :**
order_id-p_id
(composite key)

PRIME ATTRIBUTES:
order_id,p_id
NON-PRIME ATTRIBUTES:
order date,c_id,c name,c state,p name,p price,p quantity

Order_ID -> Order_Date → does not satisfy 3NF conditions
C_ID -> C_Name → does not satisfy
P_ID -> P_Name → does not satisfy
P_ID -> P_Price → does not satisfy
Order_ID -> C_ID → does not satisfy
C_ID -> C_State → does not satisfy
P_ID and Order_ID -> P_Quantity → satisfies

R1(Order_id , P_ID, P_quantity**)**

P_ID	**Order_ID**	P_quantity
3	101	1
7	101	2
6	101	1
5	102	4
6	102	5

R2(Order_ID , Order_date , C_ID**)**

Order_ID	Order_date	C_ID
101	1/10/22	1
102	9/10/22	2

R3(P_ID , P_name , P_price**)**

P_ID	P_Name	P_price
3	pen	50
7	paper	100
6	pencil	20
5	scale	10

R4(C_ID, C_name , C_state**)**

C_ID	C_name	C_state
1	Alice	LA
2	Bob	CA