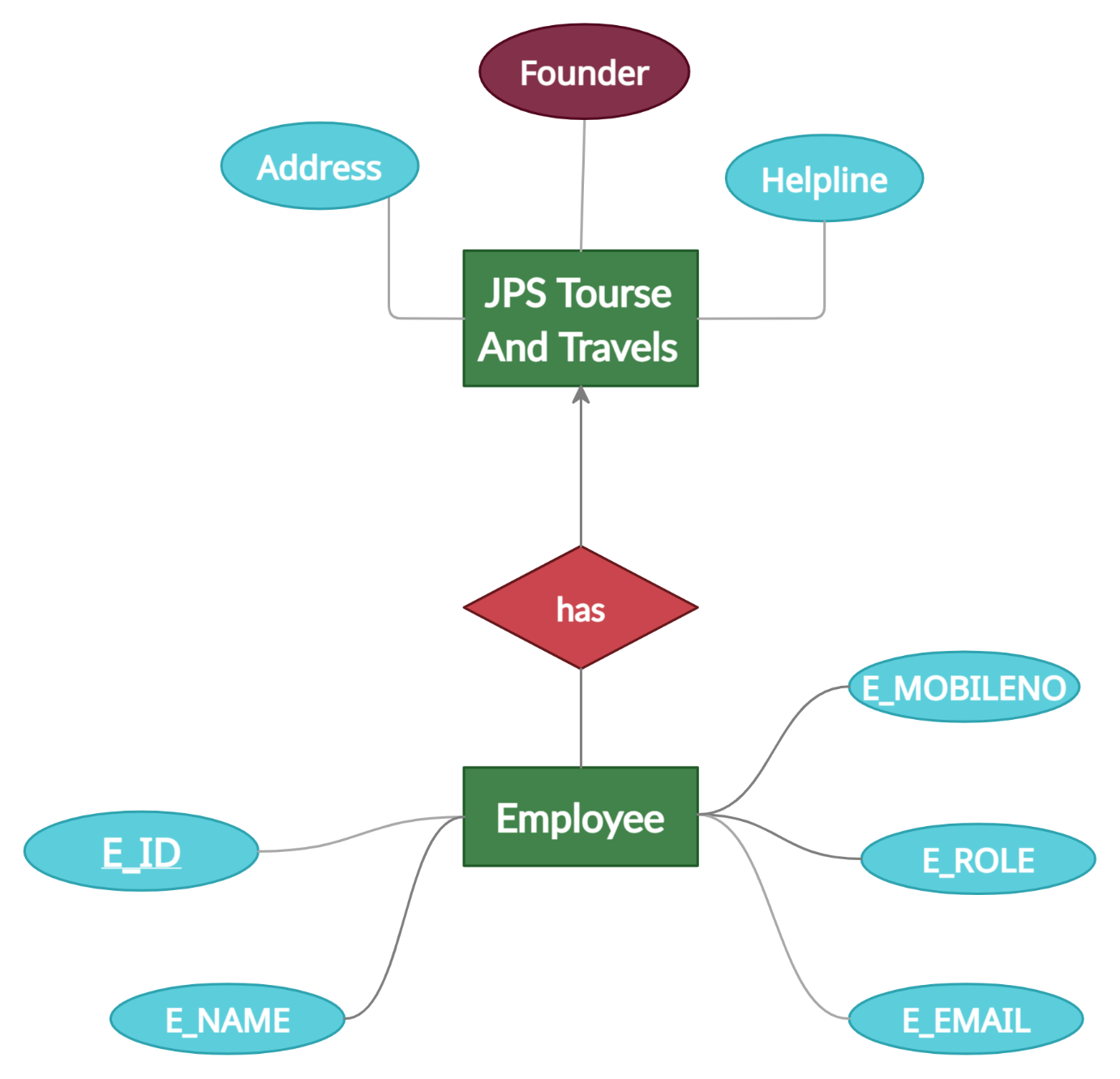
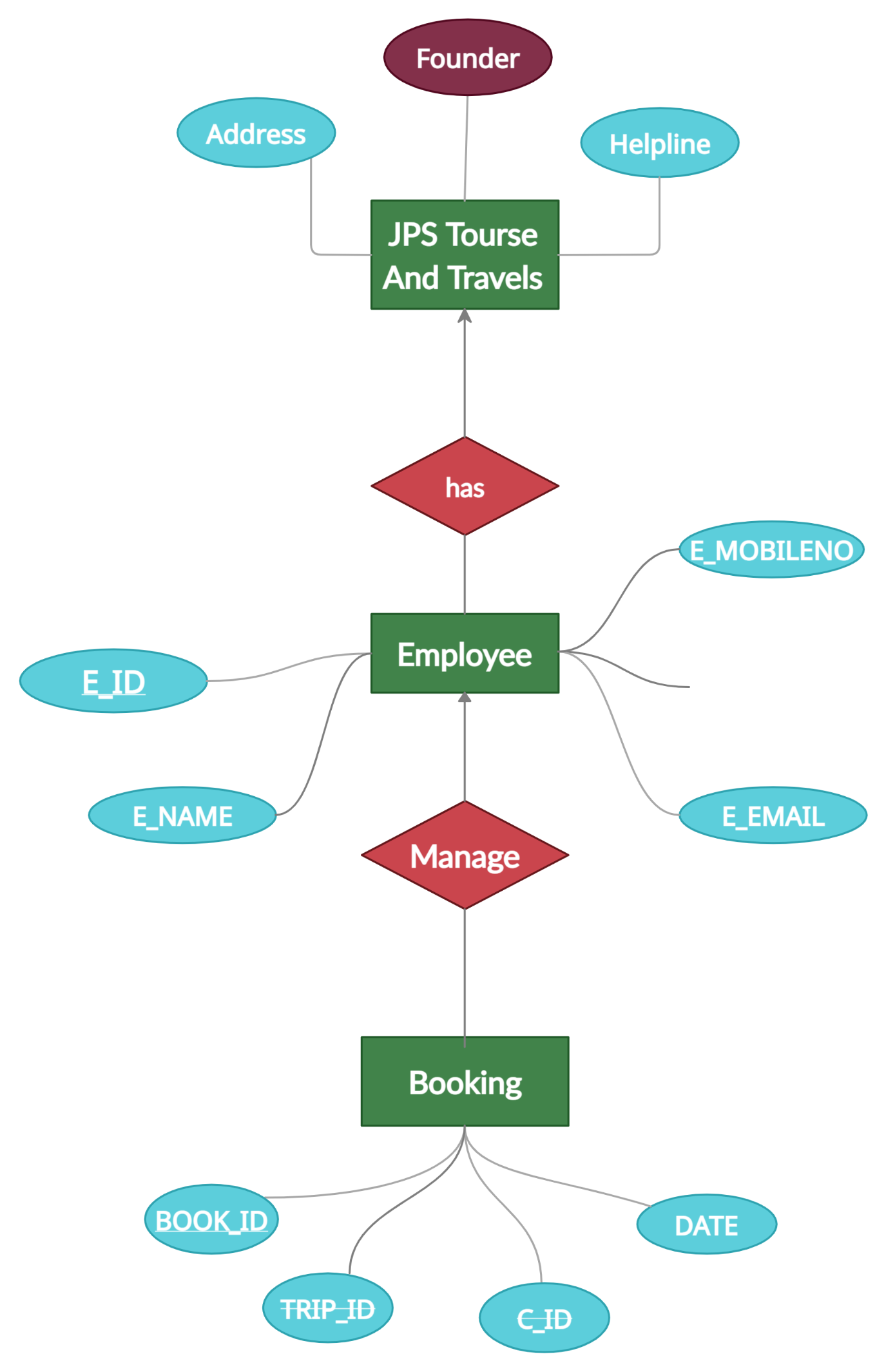
**Underline** :- **Primary key**

**~~Crossline~~ :- Foreign key**

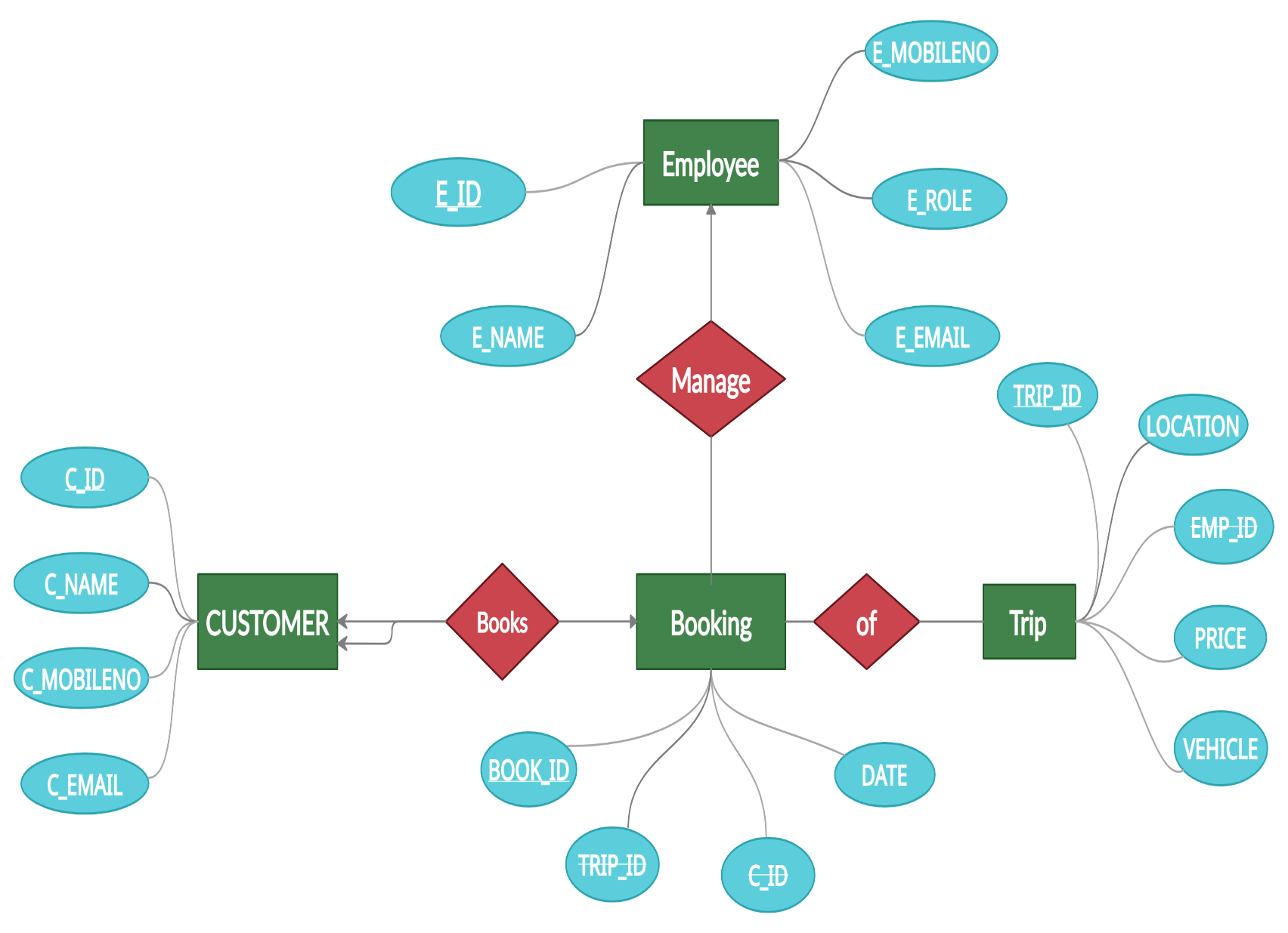


* ***JPS Tours And Travels (Founder,Address,Helpline)***
* As the Entity Jps tours and travels have only relation with Employee which is **one to many** relation so there is no change in its relational attributes
* The has relation is uniquely define by the primary key of the employee as there is no such primary key for entity JPS Tours And Travels

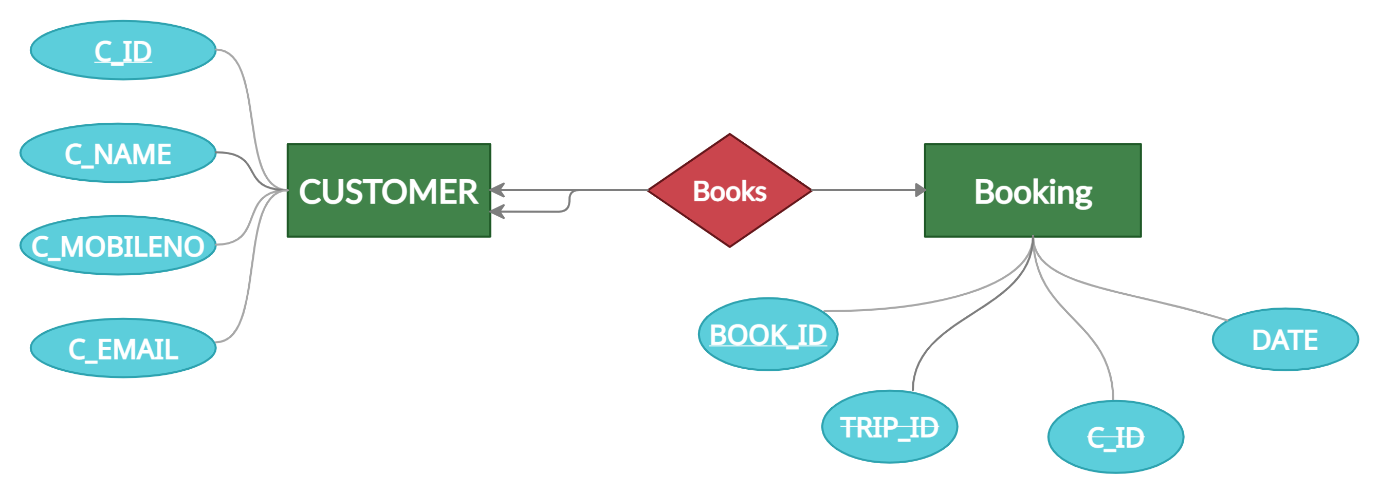


* ***Employee (E\_ID,E\_NAME,E\_MOBILENO,E\_ROLE,E\_EMAIL)***
* AS the “Employee” has the **many to one** relation with “Jps tours and travels” and **no total participation** so due to it there is no change in relational attributes of “Employee”.
* Also “Employee” has the **one** **to many** relation with “Booking” so due to it there is no change in relational attributes of “Employee”.
* The relation between manages uniquely identify by the Primary key of Both the relation like

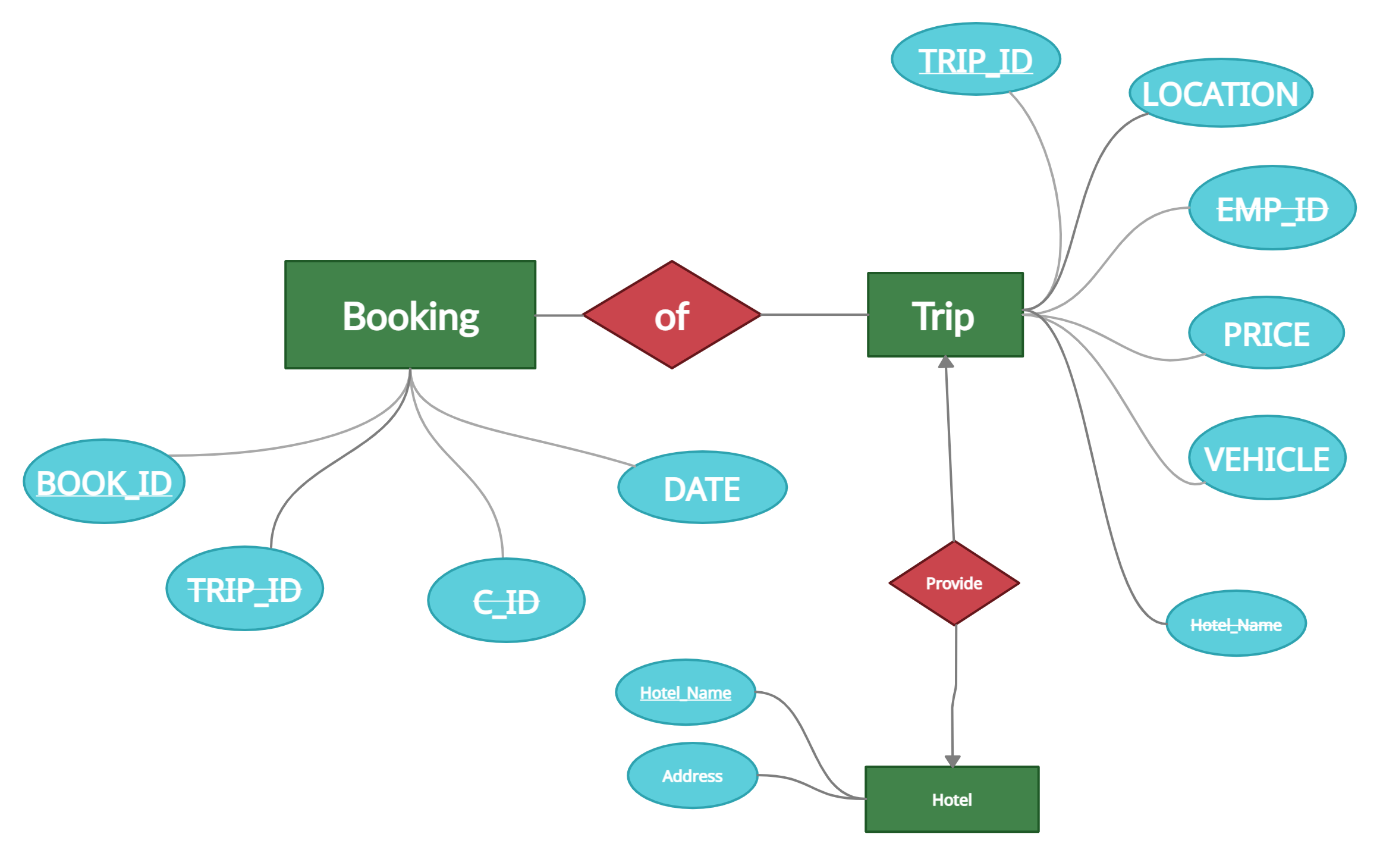
**Manages(E\_ID,BOOKING\_ID)**



* ***Booking (BOOK\_ID,TRIP\_ID,~~C\_ID~~,DATE)***
* AS the “Booking” has the **many to one** relation with “Employee” and **no total participation** so due to it there is no change in relational attributes of “Booking”.
* AS the “Booking” has the **many to one** relation with “Customer” and **no total participation by Booking** so due to it there is no change in relational attributes of “Booking”.
* AS the “Booking” has the **many to many** relation with “Trip” so due to it there is no change in relational attributes of “Booking”.

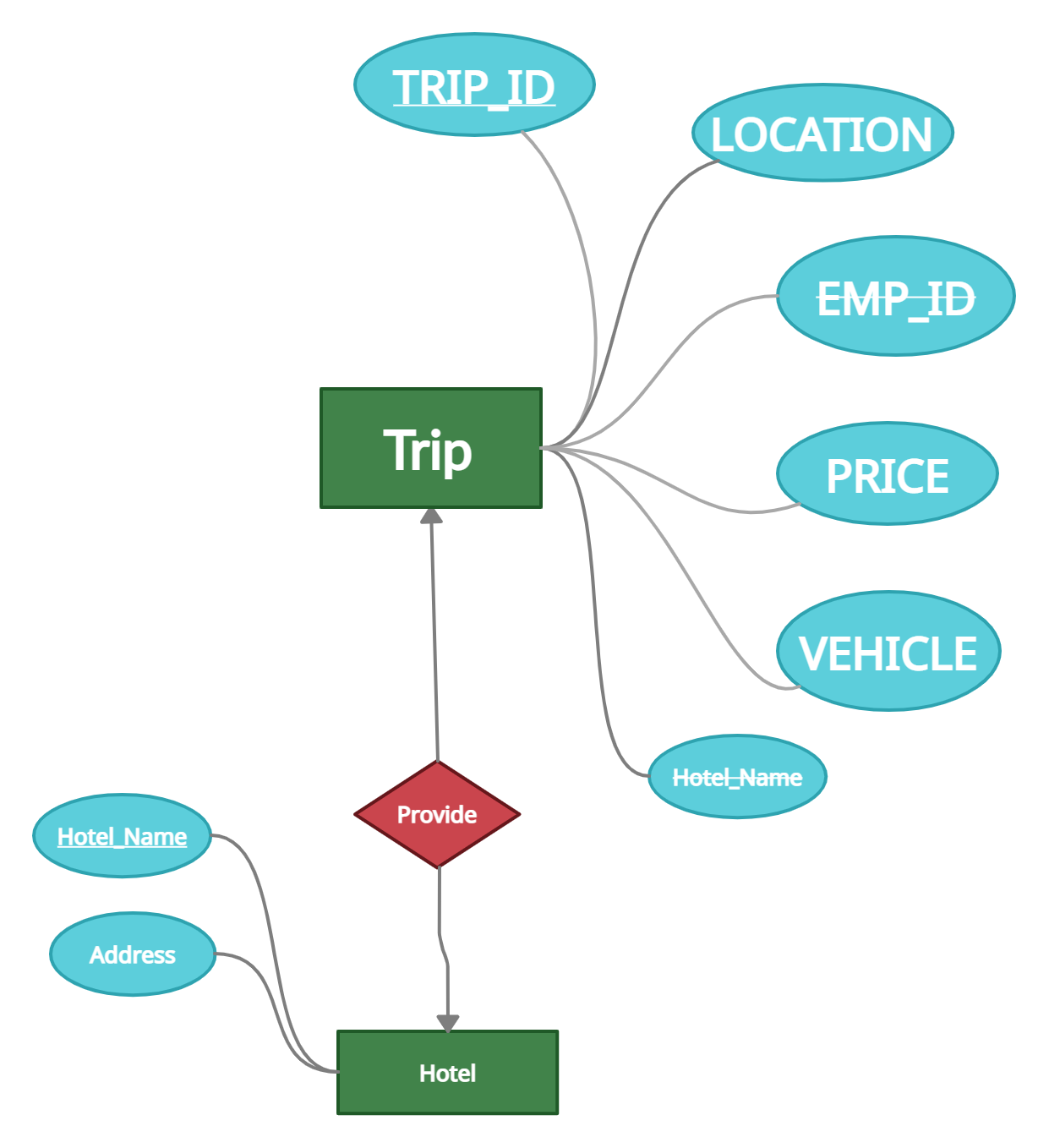


* ***Customer (C\_ID,C\_NAME,C\_MOBILENO,C\_EMAIL,~~BOOK\_ID~~)***
* As Entity “Customer” have only relation with the “Booking” which is **Many to one** and here **customer have total participation with relation Books** so customer and books will merge so booking primary key will be given as foreign key to the customer



* ***Trip (Trip\_Id,Location,~~Emp\_id,~~Price,Vehicle~~,Hotel\_name~~)***
* AS the “Trip” has the **many to many** relation with “Booking” so due to it there is no change in relational attributes of “Trip”.
* AS the “Trip” has the **one to one** relation with “Hotel” and **no total participation** so due to it there is no change in relational attributes of “Trip”.
* OF relation will uniquely identify by the primary key of Booking and Trip

**OF (BOOK\_ID,TRIP\_ID)**



* ***Hotel (Hotel\_name,Address)***
* AS the “Hotel” has the **one to one** relation with “Trip” and **no total participation** so due to it there is no change in relational attributes of “Hotel”
* The relation **“Provide”** will uniquely identify by the primary key of Hotel and Trip

***Provide (TRIP\_ID,HOTEL\_NAME)***

* **Relational entity and attributes**
* ***JPS Tours And Travels (Founder,Address,Helpline)***
* ***Employee (E\_ID,E\_NAME,E\_MOBILENO,E\_ROLE,E\_EMAIL)***
* ***Booking (BOOK\_ID,TRIP\_ID,~~C\_ID~~,DATE)***
* ***Customer (C\_ID,C\_NAME,C\_MOBILENO,C\_EMAIL,~~BOOK\_ID~~)***
* ***Trip (Trip\_Id,Location,~~Emp\_id,~~Price,Vehicle~~,Hotel\_name~~)***
* ***Hotel (Hotel\_name,Address)***
* **Relation**
* ***Provide (TRIP\_ID,HOTEL\_NAME)***
* **OF (BOOK\_ID,TRIP\_ID)**
* **Manages(E\_ID,BOOKING\_ID)**