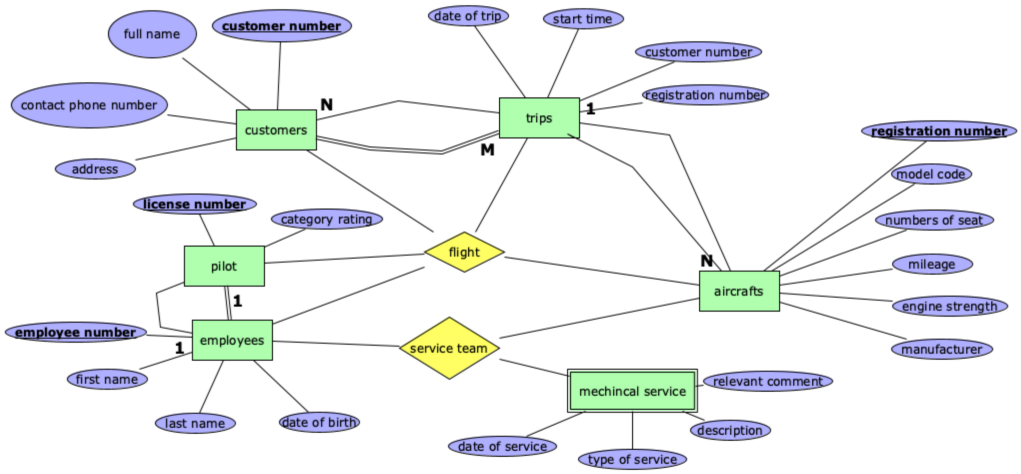
1,

Since customer number is unique , it is a key attributes. Assume that employee number identify each employee and registration number identify each aircraft so they are key attributes.



Requirements that can't be described in ERR diagram is as follows:
1,Each service team is composed of three service engineers and one administrative staff member.

2,To maintain optimal performance and compliance with safety standards, each aircraft undergoes a thorough mechanical service every six month.

3,All pilots must hold either a Commercial Pilot License (CPL) or an Air Transport Pilot License (ATPL).

4, A pilot is only allowed to fly aircraft that fall within their specific category ratings, ensuring they are qualified and authorized to operate those types of aircraft.

5,Each flight requires a pilot and a co-pilot.

2.1

{Candidate Name, Candidate DoB, Job ID} is the candidate key.
Thus *{Candidate Name, Candidate DoB}* is part of the candidate key which means FD1 satisfy the 3NF.The same is FD2 and FD6.However, there is no any part of candidate key in FD3 or FD4 or FD5.

Minimal cover:
{

{Candidate Name, Candidate DoB}→{Candidate Email}
{Candidate Name, Candidate DoB}→{Candidate Education}
{Job ID}→{Position}
{Job ID}→{Type}
{Job ID}→{Closing Date}
{Job ID}→{Employer ID}
{Position, Type, Location, Employer ID, Employer Address}→{Salary}
{Employer ID}→{Employer Name}
{Employer ID}→{Employer Address}
{Employer Address}→{Location}
{Candidate Name, Candidate DoB, Job ID}→{Application Date}
{Candidate Name, Candidate DoB, Job ID}→{Status}
}
Thus, get
R_1 = {Candidate Name, Candidate DoB,Candidate Email}
R_2 = {Candidate Name, Candidate DoB,Candidate Education}
R_3 = {Job ID, Position}
R_4 = {Job ID, Closing Date}
R_5 = {Job ID, Type}
R_6= {Job ID, Employer ID}
R_7 = {Employer ID, Employer Name}
R_8 = {Employer ID, Employer Address}
R_9 = {Employer, Address, Location}
R_{10} = {Candidate Name, Candidate DoB, Job ID, Application Date}
R_{11} = {Candidate Name, Candidate DoB, Job ID, Status}
R_0 = {Candidate Name, Candidate DoB, Job ID} (omited)
The above result can not been refined.
Thus the 3NF decomposition is given by $\{R_i \mid i = 1,2...,11\}$ with FD keeping in Minimal cover.

2.2

Consider, if we keep some FDs which violates the BCNF preserving which means arbitrary our decomposition algorithm will not cause these FDs' losses.
If we change the decomposition order since ,it will not cause FD losses.Thus, each relation schemes is well defined by FDs.Since the FDs don't disappear, the order doen't matter the final results.Thus, we could first consider what FDs we would like to keep before we apply the decomposition and how many different combinations of FDs preserving or FD losses is the actual how many different results we could get.

Clearly,FD1-FD5 doesn't satisfy the BCNF.And {Employer ID} → {Employer Name, Employer Addresss}→ {Location} which means location could also be deduced by {Employer ID} in FD4.

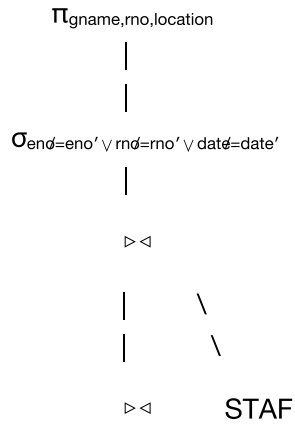
(a)

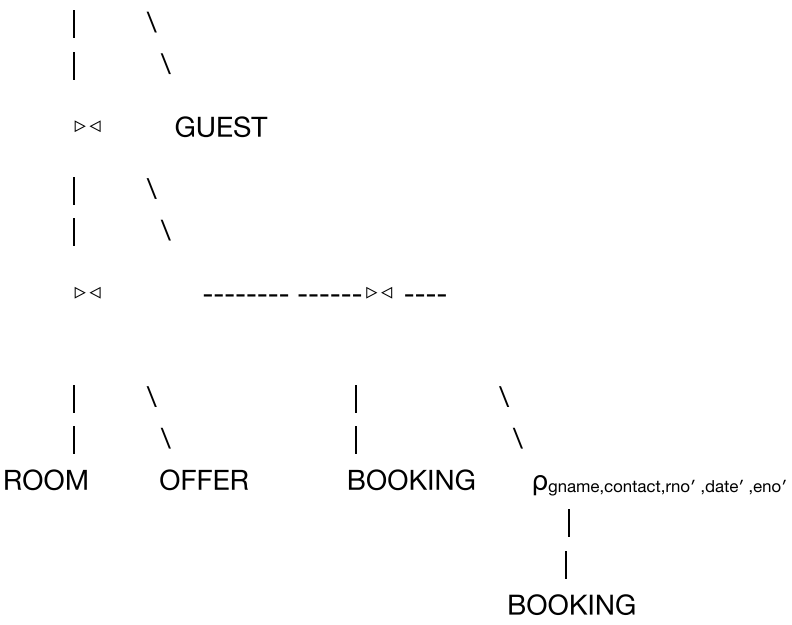
Consider FD1-FD5. FD2,FD4,FD5 has relation with FD3 which means if we apply BCNF decomposition along any of these before dealing along FD3 would make FD3 unavailable. While decomposition along FD2 would affect FD4. Besides, decomposition along FD4 would affect FD5. So basically there are only 2 different results for each scenario (keeping FD3 or not, keeping FD4 or not , keeping FD5 or not).
Thus,there are 2^3 which are 8 kinds of different decomposition relation schemes.
(b)
Consider decomposition along FD3 , FD5 , FD4 ,FD2,FD1 orderly.
Thus, we get as follows :
R_1 = {Position, Type, Location, Employer ID, Employer Address,Salary} with FD3
R_2 = {Employer Address,Location} with FD5
R_3 = {Employer ID,Employer Name, Employer Address} with FD4
R_4 = {Job ID,Position,Type,Closing Date, Employer ID} with FD2
R_5 = {Candidate Name, Candidate DoB,Candidate Email, Candidate Education} with FD1
R_5 ={Candidate Name, Candidate DoB,Job ID,Application Date, Status} with FD6
Above decomposition keeps all FD thus are lossless.

3.1

```
SELECT G.gname, B.rno ,R.location
FROM ROOM AS R
JOIN OFFER AS O ON R.rno=O.rno
JOIN (SELECT B1.gname,B1.contact,B1.eno,B1.rno,B1.date,B2.eno AS eno1,B2.rno
AS rno1, B2.date AS date1
FROM BOOKING AS B1
JOIN BOOKING AS B2 ON B1.gname = B2.gname AND B1.contact = B2.contact )
AS B ON O.rno=B.rno AND O.date=B.date
JOIN GUEST AS G ON B.gname=G.gname
JOIN STAFF AS S ON B.eno=S.eno
WHERE B.eno <> B.eno1 OR B.rno <> B.rno1 OR B.date <> B.date1
```

3.2





3.3

(a)

gname	contact	rno	date	eno	rno’	date’	eno’
Pepe	Hotsforskunksville	11	29/04/2016	1	11	29/04/2016	1
Coyote	Ringroad	17	29/04/2016	1	17	29/04/2016	1
Pepe	Hotsforskunksville	13	13/04/2016	1	13	13/04/2016	1
Pepe	Hotsforskunksville	11	29/04/2016	1	13	13/04/2016	1
Pepe	Hotsforskunksville	13	13/04/2016	1	11	29/04/2016	1

(b)

gname	rno	location
Pepe	11	level0
Pepe	13	level13

3.4

$$\pi_{gname,rno,location}(\pi_{rno,location}(ROOM) \triangleright\triangleleft \pi_{gname,rno}(\sigma_{eno' = eno \vee rno' = rno \vee date' = date} (BOOKING \triangleright\triangleleft \rho_{gname,contact,rno',date',eno'}(Booking))))$$