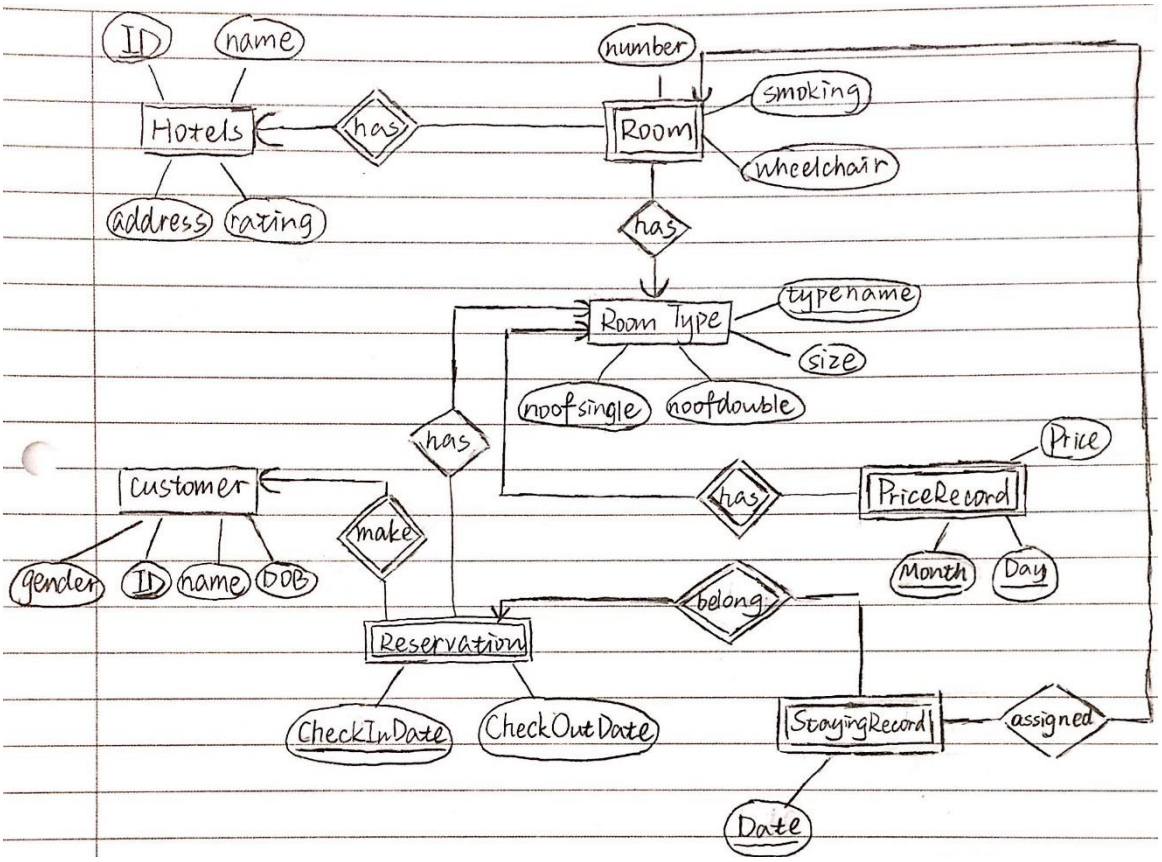
Solver: Huang Xuhui

Email Address: C150177@e.ntu.edu.sg

1. (a) (i)



(ii)

Hotels (<u>ID</u>, name, address, rating)
Room (<u>HotelID</u>, <u>number</u>, roomtype, smoking, wheelchair)
RoomType (<u>typename</u>, size, noOfSingle, noOfDouble)
PriceRecord (<u>typename</u>, <u>month</u>, <u>day</u>, price)
Customer (<u>ID</u>, gender, name, DOB)
Reservation (<u>customerID</u>, <u>checkInDate</u>, typename, checkOutDate)
StayingRecord (<u>date</u>, <u>customerID</u>, <u>checkInDate</u>, hotelID, roomNumber)

(b) (i)

R1 := $\sigma_{\text{source = "Seoul" and destination = "Busan"}}$ TRAIN

R2 := Π_{TID} R1

R3 := R2 $\bowtie$ TICKET

RESULT := $\gamma_{\text{count(TID)} \rightarrow \text{Passengers}}$ R3

(ii)

R1 := σ source = "Seoul" and destination = "Busan" TRAIN

R2 := σ source = "Busan" and destination = "Seoul" TRAIN

R3 := R1 $\bowtie$ R1.Date = R2.Date R2

RESULT := Π (TID, PID) TICKET $\div$ Π TID R3

(iii)

R1 := γ TID, count(PID) $\rightarrow$ numOfBooked TICKET

R2 := R1 $\bowtie$ R1.numOfBooked = TRAIN.NumberOfSeats and R1.TID = TRAIN.TID TRAIN

RESULT := Π TID R2

(iv)

R1 := γ PID, count(TID) $\rightarrow$ travelTimes TICKET

R2 := Π PID (σ travelTimes > 100 R1)

R3 := TICKET $\bowtie$ TICKET.PID = R2.PID R2

R4 := γ TID, count(PID) $\rightarrow$ numOfBooked R3

R5 := R4

R6 := R4 $\bowtie$ R4.TID <> R5.TID and R4. numOfBooked < R5. numOfBooked R5

R7:= Π TID R4 - Π TID R6

RESULT := Π source, destination (R7 $\bowtie$ TRAIN)

2. (a) FDs: AF → D, B → C, BE→ D, BDA → F, CE → A, D→ E
B is not in right hand side of FDs, B must be contained in the key
{B}+ = {B, C},
{A, B}+ = {A, B, C},
{B, C}+ = {B, C},
{B, D}+ = {B, C, D, E, A, F},
{B, E}+ = {B, E, C, A, D, F},
{B, F}+ = {B, F, C},
{A, B, C}+ = {A, B, C},
{A, B, F}+ = {A, B, F, D, E, C},
{B, C, F}+ = {B, C, F}
Key: BD, BE, ABF

AF → D violated BCNF. {A, F, D}+ = {A, ,F, D, E}
R (A, B, C, D, E, F) → R1 (A, F, D, E) and R2 (A, F, D, B, C)

B → C violated BCNF. {B, C}+ = {B, C}
R2 (A, F, D, B, C) → R3 (B, C) and R4 (A, F, D, B)

D → E violated BCNF. {D, E}+ = {D, E}
R1(A, F, D, E) → R5 (A, F, D) and R6 (D, F)
For R4 (A, F, D, B)

{A}+ = {A},
{B}+ = {B, C},
{D}+ = {D, E},
{F}+ = {F},
{A, B}+ = {A, B, C},
{A, D}+ = {A, D, E},
{A, F}+ = {A, F, D}
R4 (A, F, D, B) → R7 (A, F, B) AND R8 (A, F, D)

FDs that is not preserved : C, E→ A

(b) FDs: AF → D, B → C, BE→ D, BDA → F, CE → A, D→ E
Try to remove AF → D, {A, F}+ = {A, F}, AF → D cannot be remove.
Try to remove B → C, {B}+ = {B}, B → C cannot be remove.
Try to remove BE→ D, {B, E}+ = {B, E, C, A}, BE→ D cannot be remove.
Try to remove BDA → F, {B, D, A}+ = {B, D, A, C, E}, BDA → F cannot be remove.
Try to remove CE → A, {C, E}+ = {C, E}, CE → A cannot be remove.
Try to remove D→ E, {D}+ = {D}, D→ E cannot be remove.

Next step try to remove the redundant attribute of left hand side.
Only A can be removed from BDA → F
Now the FDs: AF → D, B → C, BE→ D, BD → F, CE → A, D→ E
Finally, R (A, B, C, E, F) is normalized into R1 (A, F, D), R2 (B, C), R3 (B, E, D), R4 (B, D, F)
R5 (C, E, A)

3. (a) (i)

```
select categoryname from Category as ct1 where not exists (select
categoryname from Category as ct2 where ct1.categoryname =
ct2.belingsto)
```

(ii)

```
select Borrowed.ReaderNr from(select Loan.ReaderNr,Loan.ISBN from
Loan, Book where Loan.ISBN = Book.ISBN and Book.author = 'Jiawei
Han' group by Loan.ReaderNr,Loan.ISBN) as Borrowed group by
Borrowed.ReaderNr having count(Borrowed.ReaderNr)= (select
COUNT(ISBN) from Book where author  = 'Jiawei Han')
```

Note: question (ii) may has a simpler solution, the above one is the one I write in the exam

(iii)

```
select DISTINCT ISBN from Copy where ISBN in (select Rest.ISBN
from(select Copy.ISBN,Copy.copynumber from Copy except select
ISBN,copy from Loan) AS Rest)
```

(b)
CHECK: Create ASSERTION noMoreThan20 AS Check(NOT exists((select count(ReaderNr) from Loan group by ReaderNr) >20));

TRIGGER:

```
Create TRIGGER noMoreThan20
AFTER INSERT ON Loan
REFERENCING NEW ROW AS new FOR
EACH ROW
WHEN (select count(ReaderNr) from Loan group by ReaderNr) >20
ROLLBACK
```

(c)

```
Update Loan
  SET ReturnDate = DATEADD(day, 30, ReturnDate)
  Where ReturnDate < ’15.03.2013’
```

4. (a)Left outer join

R ⟕L S =

$$(R \bowtie S) \cup ((R - \pi_{r_1, r_2, \ldots, r_n}(R \bowtie S)) \times \{(\omega, \ldots \omega)\})$$

r1, r2…rn are the attributes of R, the ω is the null value (those that are not attributes of *R*)

Right outer join

R ⟖R S =

$$(R \bowtie S) \cup (\{(\omega, \ldots, \omega)\} \times (S - \pi_{s_1, s_2, \ldots, s_n}(R \bowtie S)))$$

s1, s2…sn are the attributes of R, the ω is the null value (those that are not attributes of *S*)

Full outer join

R ⟗Full S = (R ⟕L S) ∪ (R ⟖R S)

More details can be found:
https://en.wikipedia.org/wiki/Relational_algebra#Left_outer_join_.28.E2.9F.95.29

(b) Let's use A, B, C, D, E to present the Student_ID, Date_Enrolled, Course_ID, Room NR, Professor respectively. Now we have R (A, B, C, D, E), R1 (A, B, C) and R2 (B, D, E). From description, we have C → E, C → D and A, C → B

(i) R

A	B	C	D	E
1	2	1	2	3
3	2	2	3	4

R1

A	B	C
1	2	1
3	2	2

R2

B	D	E
2	2	3
2	3	4

R1 ⋈ R2

A	B	C	D	E
1	2	1	2	3
1	2	1	3	4
3	2	2	2	3
3	2	2	3	4

(ii) R1 (C, D, E), R2 (A, B, C)
C is super key of R1, so it's lossless decomposition

(c) (i)

1	1100
2	2200
3	3300
4	4400

(ii)

1	4400
2	4400
3	4400
4	4400

(d) (i) //*[@choiceNum = "1"]/[@meritScore > 800]
(ii) //*[@applicant = //* [@name = "Doreen"]/@appNum]/@code

Note: Redo this paper a few months after my exam, if there are any errors please send me an email.

Thank you and all the best for your final exam