2.a. $(r_1 \cap r_2) \bowtie s = (r_1 \bowtie s) \cap (r_2 \bowtie s)$

Let $L \equiv (r_1 \cap r_2) \bowtie s$ & $R \equiv (r_1 \bowtie s) \cap (r_2 \bowtie s)$

We have to show, $L \subseteq R$ and $R \subseteq L$

$L \subseteq R$

For any tuple $t \in L$, we have

$$\exists\, t_{r_1} \in r_1 \wedge \exists\, t_{r_2} \in r_2 \mid (t_{r_1}[A] = t_{r_2}[A] = t[A]) \wedge (t_{r_1}[B] = t_{r_2}[B] = t[B]) \wedge \exists\, t_s \in s \mid t_s[B] = t[B]$$

$\Rightarrow \exists\, t_{r_1} \in r_1 \wedge \exists\, t_s \in s \mid (t_{r_1}[B] = t[B] = t_s[B])$ $\hookrightarrow (r_1 \bowtie s)$

$\wedge \exists\, t_{r_2} \in r_2 \mid (t_{r_2}[B] = t[B] = t_s[B])$ $\hookrightarrow (r_2 \bowtie s)$

$\wedge (t_{r_1}[A] = t_{r_2}[A] = t[A])$ $\hookrightarrow (r_1 \bowtie s) \cap (r_2 \bowtie s)$ [$\because$ intersection is associative]

$\Rightarrow t \in (r_1 \bowtie s) \cap (r_2 \bowtie s)$

$R \subseteq L$

Similarly we can prove this direction. So VALID

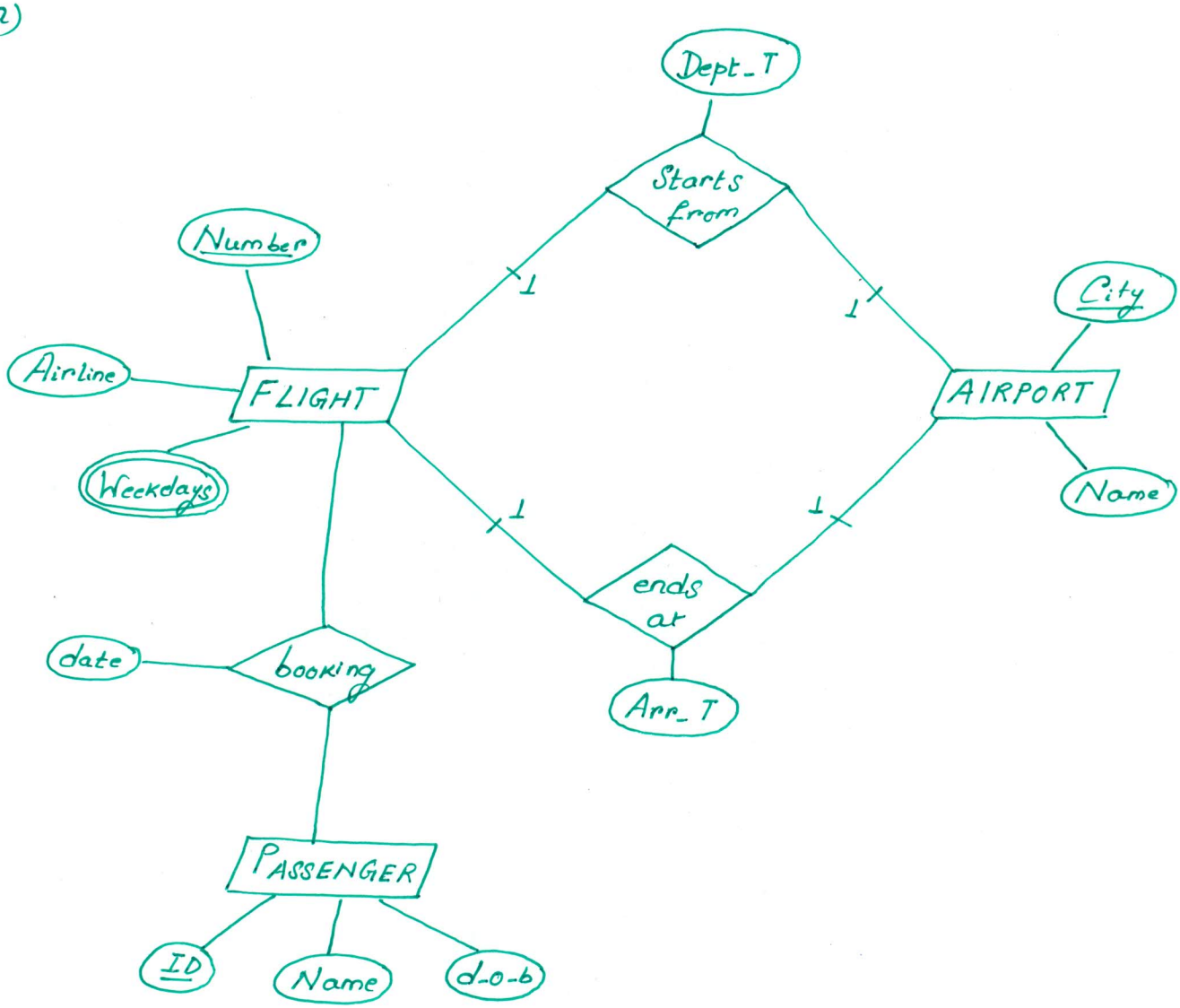
c) INVALID → Counter example is —

r_1		r_2		$(r_1 - r_2)$	$\Pi_A(r_1 - r_2)$	$\Pi_A(r_1) - \Pi_A(r_2)$
a_1	b_1	a_1	b_1	a_2 b_2	a_2	NULL.
a_2	b_2	a_2	b_3	a_1 b_3	a_1	
a_1	b_3	a_1	b_2			
a_2	b_3					

DATABASE MANAGEMENT SYSTEMS

Class Test - 1

1. a)



TABLES:

1. (b)

(i) FLIGHT (Number, Airline, Dept-C, Dept-T, Arr-C, Arr_T)

(ii) AIRPORT (City, Name)

(iii) DAY_OF_OP (Number, Day)

** This table is needed because "Weekdays" is a multivalued attribute.

(iv) BOOKINGS (ID, Number, date)

** For 'date' attribute we can keep one CONSTRAINT that will check if the corresponding flight operates on that particular day. In that case 'date' can be a multivalued attribute with date, month, year and day.

1. (c) i)

```
SELECT Number FROM FLIGHT, DAY-OF-OP
WHERE FLIGHT. Number = DAY-OF-OP. Number
    AND FLIGHT. Dep_C = 'Kolkata'
    AND FLIGHT. Arr_C = 'Delhi'
    AND DAY-OF-OP = 'Sunday';
```

ii)

```
SELECT F. AirLine FROM FLIGHT as F
WHERE NOT EXISTS
    (SELECT A. City FROM AIRPORT as A EXCEPT
    ((SELECT DISTINCT C. Dep_C FROM FLIGHT as C
        WHERE C. AirLine = F. AirLine)
    UNION
    (SELECT DISTINCT D. Arr_C FROM FLIGHT as D
        WHERE D. AirLine = F. AirLine)))
```

1. (c) iii) For no stop :

```
SELECT Number FROM Flight , DAY_OF_OP
  WHERE FLIGHT. Number = DAY_OF_OP. Number
     AND DAY_OF_OP. Day = 'Tuesday'
     AND FLIGHT. Dep_C = 'Kolkata'
     AND FLIGHT. Arr_C = 'Pune';
```

For one stop :- Answer will be a pair of Flight numbers

```
SELECT A.Number, B.Number
FROM FLIGHT as A, FLIGHT as B, DAY_OF_OP as S1, DAY_OF_OP as S2
WHERE S1. Day = 'Tuesday' AND S2. Day = 'Tuesday'
   AND A. Number = S1. Number AND B. Number = S2. Number
   AND A. Dep_C = 'Kolkata' AND B. Arr_C = 'Pune'
   AND A. Arr_C = B. Dep_C AND B. Dep_T > A. Arr_T ;
```

iv) In similar fashion as (iii) is done.