Niyati - 47943319

QUES 1

A)Normalisation is a process of organising the dataset and removing data redundancy .It involves removing update, insertion and deletion.

The following data is in the **2nd NF** as

1) 1ST NF Exist -Dataset doesn't have any **repeated groups** i.e. attributes are single values not multivalued.

2) there is only 1 column in the Primary key i.e. ENTRY ATTEMPT ID , there is **no partial dependency** (all columns are dependent on the FULL primary key only)

3) There exist many transitive dependencies so not in 3NF

{entry attempt id } -> {staff id } ->{given name, family name ,staff card id }

{entry attempt id } -> {staff card id}->{card issue date}

{staff id} -> {staff card id} ->{card issue date}

{entry attempt id } ->{permit reservation id } -> {permit from date,permit to date ,permit for number plate}

B)For BCNF

a) First need to convert into **3rd NF** – Check **Transitive dependency**

There exist many transitive dependencies like

{entry attempt id } -> {staff id } ->{given name, family name ,staff card id }

{entry attempt id } -> {staff card id}->{card issue date}

{staff id} -> {staff card id} ->{card issue date}

{entry attempt id } ->{permit reservation id } -> {permit from date,permit to date ,permit for number plate}

Therefore table can be decomposed into

1. Staff Table

Primary Key: Staff ID

Attributes: Given Names, Family Name

2. Staff Card Table

Primary Key: Staff Card ID

Attributes: Card Issue Date

3. Permit Reservation Id

Primary Key: Permit Reservation ID

Attributes: Permit From Date, Permit To Date, Permit For Number Plate

4. Entry Attempt

Primary Key: Entry Attempt ID

Attributes: Parking Area ID, Entry Datetime, Exit Datetime

5. Parking Entry Information Table

Primary Key: Entry Attempt ID

Foreign Keys: Staff ID, Permit Reservation ID, Staff Card ID

b) Secondly for **BCNF** all the tables Attributes need to be dependent only on a **super key** that is left-hand side of the functional dependency stated should be the super key of the respective table.

{staff id },{staff card id },{entry attempt id},{permit reservation id } are super keys of there respective table. Therefore, BCNF exists in above decomposed table only.

QUES 2

Create table and insert values are directly in sql template

QUES 3

A)List each staff number and the number of swipe card records that exist for each staff number. Sort by the largest number of records first to the smallest number of records being last.

```
1 •  SELECT `staff number`,COUNT(`card id`) AS `no. of card id ` FROM `swipe card`
2    GROUP BY `staff number`
3    ORDER BY `no. of card id ` DESC
4    ;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

staff number	no. of card id
WILL45	4
WILL13	4
MEET94	4
POMO92	4
ELLE50	4
LEON94	3
ALEX87	3
JESS88	3
CRIS89	3

B)List the distinct staff names and staff numbers (from the staff card) of all staff who have 2 or more spot reservations. Sort by staff surname in ascending dictionary order.

```
1 •  SELECT  DISTINCT `name on card`,`staff number` FROM `swipe card`
2    JOIN `spot reservation` ON `swipe card`.`card id`=`spot reservation`.`card id`
3    GROUP BY `spot reservation`.`card id`
4    HAVING COUNT(`spot reservation`.`card id`)>=2
5    ORDER BY SUBSTRING_INDEX (`name on card`, ' ', -1) ASC ;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

name on card	staff number
Amelia Elle	AMEL54
Isla Emmy	ISLA75
Niyati Niyati	NIYA62
Ram Ram	RAMR64
Alex Right	ALEX87
JK Wills	WILL13

C)Show all entry attempts for a chosen swipe card (use **where `card id` =** to choose the card id you want to use for your query). Leave a comment above your query indicating which card id you want the marker to test with. (e.g. 2487)

```
1  SELECT * FROM `spot area entry attempt`
2  WHERE `card id`=2487;
3  -- all the attempt to enter for card id =2487--
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

attempt id	card id	area id	date and time of entry
2004	2487	1358	2020-02-16 08:42:00
2109	2487	1368	2021-02-17 09:04:00
2118	2487	1361	2021-08-18 09:04:00
2198	2487	1358	2021-02-17 09:04:00
2216	2487	1366	2022-08-16 09:15:00
2304	2487	1361	2023-02-16 08:55:00
2309	2487	1367	2023-02-17 09:04:00
2315	2487	1360	2023-03-16 09:13:00
2328	2487	1358	2023-08-19 09:35:00
NULL	NULL	NULL	NULL

D)List the details of each Car Park and the total number of numbered parking spots in each car park.

```
1  SELECT `carpark id`,`map reference`,`description`,COUNT(`parking spot id`) AS `number of parking spot`
2  FROM `car park`
3  JOIN `spot reservation parking area` ON `car park`.`carpark id` = `spot reservation parking area`.`car park :
4  JOIN `numbered parking spot` ON `spot reservation parking area`.`area id` = `numbered parking spot`.`area id`
5  GROUP BY `car park id`
6  ORDER BY `number of parking spot` ASC
7  ;
8
9
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

carpark id	map reference	description	number of parking spot
2349	2O7W	Hadenfeld Avenue	24
2348	6N6W	Link Road	27
1237	22K6E	Science Road	29
5678	12E3N	Gymnasium road	37
4567	21S2S	Eastern Road	62

E) How many cars has each swipe card ever been associated with? List each swipe card id and count of ***different*** number plates.

```
SELECT `swipe card`.`card id`,COUNT(`spot reservation`.`number plate`) AS `number of car associated`
FROM `swipe card`
JOIN `spot reservation` ON `swipe card`.`card id` = `spot reservation`.`card id`
JOIN `car` ON `spot reservation`.`number plate` = `car`.`number plate`
GROUP BY `card id`
ORDER BY `number of car associated` DESC
;
```

Result Grid

card id	number of car associated
2487	4
2313	4
9862	2
3954	2
9475	2
9864	2

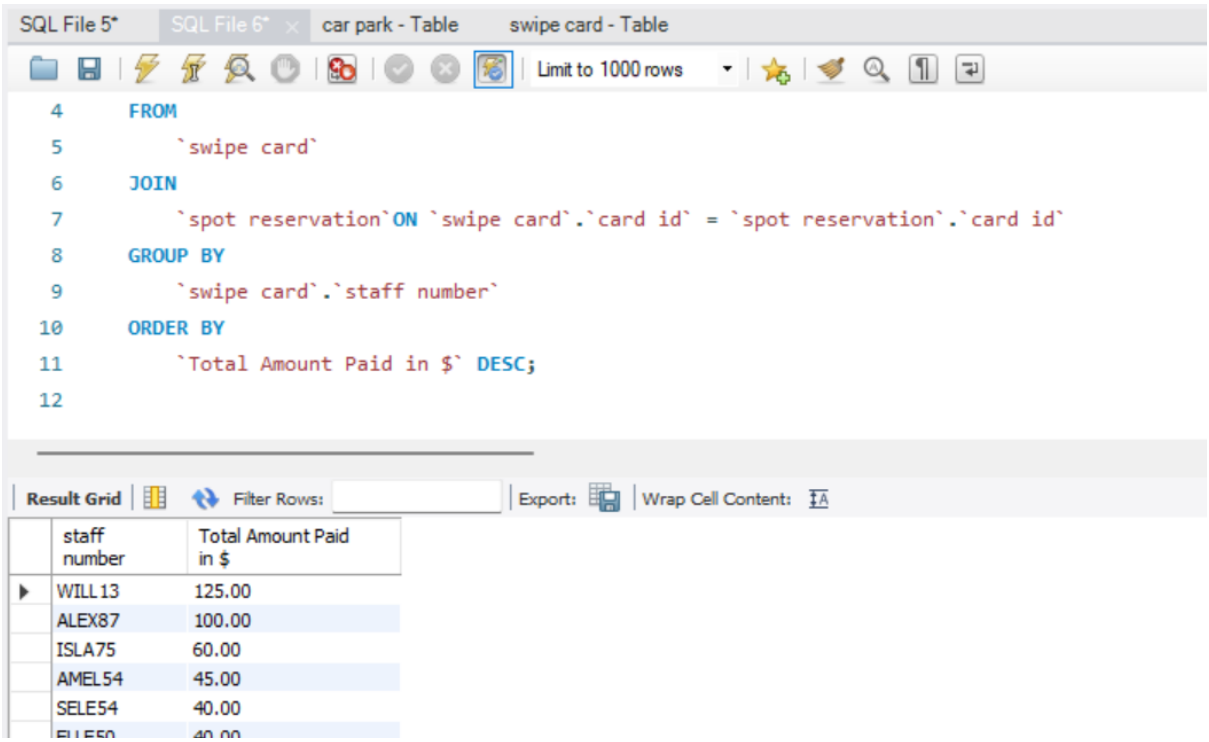
F)For a given timeslot (your choice of year and semester), how many parking spots are not allocated in each car park? Leave a comment above your query indicating which year and semester you want the marker to test with. (e.g. 2021, s2)
I did 2021,2

```
    LEFT JOIN
        `allocation` ON `numbered parking spot`.`parking spot id` = `allocation`.`parking spot id`
        AND `allocation`.`Timeslot_Year` = '2021'
        AND `allocation`.`Timeslot_Semester` = '2'
    WHERE
        `allocation`.`Timeslot_Year` IS NULL
        AND `allocation`.`Timeslot_Semester` IS NULL
    GROUP BY
        `car park`.`carpark id`
    ORDER BY
        `car park`.`carpark id`;
```

Result Grid

carpark id	Unallocated Parking Spots
2347	0
2348	27
2349	24
4567	60
5678	36
5679	0
5680	0

G)What is the total $ amount that each staff member has paid for parking during the lifetime of this system? In the query, list the staff number and the total $ amount for that staff member.



H)How much revenue (payments total) has each car park brought in each year? List the car park id, year, and total $ amount for that car park for that year.