(Sujatro Majumder SQL Assignment TAS056)

```
CREATE DATABASE sql_assgn;
USE sql_assgn;

SELECT COUNT(*) FROM airbnb_calendar_backup; -- '1308890'
-- dataset has 3585 listings each for 365 days. 1 entry has been duplicated. Hence total 3586 *
365 = 1308890 entries.

-- ans 1

SELECT DATEDIFF(MAX(date), MIN(date)) + 1 FROM airbnb_calendar; -- 365 days
-- start time
SELECT date FROM airbnb_calendar ORDER BY date LIMIT 1; -- '2016-09-06'
-- end time
SELECT date FROM airbnb_calendar ORDER BY date DESC LIMIT 1; -- '2017-09-05'

-- ans 2

SELECT COUNT(DISTINCT listing_id) FROM airbnb_calendar; -- '3585' - unique ids
-- How many properties have duplicate entries
SELECT *, COUNT(*) FROM airbnb_calendar GROUP BY listing_id, date, available, price
HAVING COUNT(*)>1;


-- delete duplicate rows
CREATE TABLE temp_airbnb LIKE airbnb_calendar;
INSERT INTO temp_airbnb
SELECT DISTINCT * FROM airbnb_calendar;
DROP TABLE airbnb_calendar;
RENAME TABLE temp_airbnb TO airbnb_calendar;


WITH calendar AS (
SELECT ROW_NUMBER() OVER (PARTITION BY listing_id, date, available, price)
AS RN FROM airbnb_calendar)
DELETE FROM calendar WHERE RN > 1;
-- '365' rows deleted, '1308525' remaining.


DELETE FROM airbnb_calendar WHERE available='t';

-- ans 3
```

```
-- list id with number of available day
CREATE TABLE calendar_days
SELECT listing_id, SUM(available='t') AS days_available, SUM(available='f')
AS days_unavailable, SUM(available='t')/COUNT(*) AS fraction_available
FROM airbnb_calendar GROUP BY listing_id;
SELECT * FROM calendar_days;
-- ans 4
-- How many properties were available on more than 50% of the days?
SELECT COUNT(*) FROM calendar_days WHERE fraction_available > 0.50;
-- How many properties were available on more than 75% of the days?
SELECT COUNT(*) FROM calendar_days WHERE fraction_available > 0.75;

-- ans 5
CREATE TABLE calendar_prices_helper
SELECT listing_id, CAST(SUBSTRING(price,2,10) AS DECIMAL(10,2)) AS prices
FROM airbnb_calendar WHERE price!='';
-- Create a table with max, min and average price of each property
CREATE TABLE calendar_prices SELECT listing_id, MAX(prices) AS max,
MIN(prices) AS min, ROUND(AVG(prices),2) AS avg
FROM calendar_prices_helper GROUP BY listing_id;
-- Extract properties with an average price of more than $500
SELECT listing_id, avg FROM calendar_prices WHERE avg>500;
```