```
/* SQL Assignment 5: SQL sub-queries
   Due Date: Sunday 4/26 at midnight
   Please submit the completed SQL code and a spreadsheet showing the result of each
SQL query */

--  example: List total price, total number of trips, average price, highest price
and lowest price of each trip.
use adventure;

Select sum(tripprice*numPersons+OtherFees) as totalprice,
        count(reservationID) as NumberTrips,
            round(avg(tripprice*numPersons+OtherFees)) averagePrice,
        max(tripprice*numPersons+OtherFees) as highestPrice,
        min(tripprice*numPersons+OtherFees) as LowestPrice
from reservation;

/*query 1. Find all reservation with number of persons in that trip less than the
average number of persons in all trips.
                   Display reservationID, trip ID, trip name and number of persons
in each trip and order by trip ID. */

select r.ReservationID, r.TripID, t.TripName, r.NumPersons
from reservation r
inner join trip t on r.TripID=t.TripID
where NumPersons< (select avg(NumPersons)
                                          from reservation)
order by TripID;

/* query 1 partial result
1600013 8       Black Pond      1
1600015 10      Mt. Cardigan - Firescrew        1
1600026 12      Cadillac Mountain Ride  2
1600014 12      Cadillac Mountain Ride  2
1600002 21      Long Pond       2
*/


/* query 2. Find all reservations that have a total trip price greater than the
average total trip price of all trips.
            Display reservationID, trip ID, trip name and trip price. Order by
tripID (in descending order) and reservationID*/

select r.ReservationID, r.TripID, t.TripName, (tripprice*numPersons+OtherFees) as
total_price
from reservation r
inner join trip t on r.TripID=t.TripID
where (tripprice*numPersons+OtherFees)> (select
avg((tripprice*numPersons+OtherFees))
                                          from reservation)
order by TripID desc, ReservationID desc;
```

```
/* query 2 partial result:
1600005 39      Welch and Dickey Mountains Hike 275.00
1600009 38      Sawyer River Ride       220.00
1600018 38      Sawyer River Ride       355.00
1600025 38      Sawyer River Ride       185.00
1600021 32      Northern Forest Canoe Trail     290.00
*/

/* query 3. Find all customers who liv in MA and have made at least two
reservations.
            Display Customer ID, customer name and number of reservation.
            order by number of reservation in descending order*/


select r.CustomerNum, concat(FirstName, ' ', LastName) as CustomerName,
count(r.ReservationID) as Num_Reserve
from customer c
inner join reservation r on c.CustomerNum=r.CustomerNum
where State='MA'
group by CustomerName
having Num_Reserve>=2
order by CustomerName, Num_Reserve desc;


/* query 3 result
104     Ryan Goff       2
102     Arnold Ocean    2
*/

/* query 4. Find all trips that have a total trip price greater than the average
total trip price of all trips.
          Display tripID, trip name and total trip price. Order by TripID*/


select r.TripID, t.TripName, sum(tripprice*numPersons+OtherFees) as total_price
from reservation r
inner join trip t on r.TripID=t.TripID
group by r.TripID, TripName
having total_price> (select avg(total_price)
                                        from
                    (select TripID, sum(tripprice*numPersons+OtherFees) as
total_price
                                          from reservation
                      group by TripID) t1)
order by TripID;
```

```
/* query 4 partial result.
4       Bradbury Mountain Ride  445.00
11      Chocorua Lake Tour      465.00
15      Crawford Path Presidentials Hike        375.00
21      Long Pond       310.00
22      Long Pond Tour  610.00
*/


/* query 5. For each customer, find all reservations that have a total trip cost
greater than the average trip cost of that customer.
            Display customer name, reservationID, trip name, trip date and total
trip cost. order by customer name*/

select concat(c.FirstName, ' ', c.LastName) as CustomerName, r.ReservationID,
t.TripName, r.TripDate, (TripPrice*NumPersons+OtherFees) as total_price
from reservation r
inner join trip t on r.TripID=t.TripID
inner join customer c on r.CustomerNum=c.CustomerNum
where TripPrice*NumPersons+OtherFees> (select avg(TripPrice*NumPersons+OtherFees)
                                        from reservation r2
                        where r2.CustomerNum=r.CustomerNum)
order by TripDate;




/* query 5 partial result
Liam Northfold  1600002 Long Pond       2018-06-08      190.00
Clement Chau    1600022 Long Pond       2018-06-08      120.00
Siam Bretton-Borak      1600016 Chocorua Lake Tour      2018-07-23      465.00
Ryan Goff       1600030 Crawford Path Presidentials Hike        2018-07-25
375.00
Karen Busa      1600019 Mount Battie Ride       2018-08-29      245.00
*/

/* query 6. Find all guides who guided the total number of trips is greater than the
average number of trips for all guides.
            Display guideNum, guide name and number of trips for each guide. */


select r.GuideNum, concat(FirstName, ' ', LastName) as GuideName,
count(ReservationID) as NumTrips
from guide g
inner join reservation r on g.GuideNum=r.GuideNum
group by r.GuideNum
having NumTrips> (select avg(NumTrips)
                                        from
                        (select count(ReservationID) as NumTrips
                        from reservation
```

```
                    group by GuideNum) t);

/* query 6 partial result
AM01    Miles Abrams    4
GZ01    Zach Gregory    4
KS01    Susan Kiley     3
UG01    Glory Unser     6
*/

/* query 7. Find most recent trip date for each customer.
            Display customer number, customer name, most recent trip date and total
trip cost of that trip.
            order by trip cost in descending order*/

-- this is wrong in general
select c.CustomerNum, concat(FirstName, ' ', LastName) as CustomerName,
                        max(r.TripDate) as 'most recent date',
(TripPrice*NumPersons+OtherFees) as total_cost
from customer c
inner join reservation r on c.CustomerNum=r.CustomerNum
group by c.CustomerNum, CustomerName
order by total_cost desc;



/* query 7 partial result
107     Quinn Marchand  2018-07-09      610.00
121     Siam Bretton-Borak      2018-09-11      465.00
126     Brianne Brown   2018-10-01      355.00
112     Laura Jones     2018-06-11      290.00
105     Kyle McLean     2018-06-25      275.00
 */

/* query 8. find most popular trip type in each season. Display season, trip type
and total prices of that type. */

select Season, Type, total_price
from
(select Season, Type, sum((r.TripPrice*r.NumPersons)+r.OtherFees) as total_price,
       rank() over(partition by Season order by
sum((TripPrice*NumPersons)+OtherFees) desc) as trip_rank
from  trip t
inner join reservation r on t.TripID=r.TripID
group by Season, Type) t
where trip_rank=1
order by Type;

/* query 8 partial result
Early Fall      Biking  1715.00
Early Spring    Hiking  110.00
```

```
Late Fall       Hiking  195.00
Late Spring     Hiking  75.00
Summer  Paddling        1465.00
*/

/* query 9. Find top 3 trips in each customer state based on total trip prices.
            Display customer state, trip id, trip name, total trip prices and rank.
            order by state and then by rank*/


select State, TripID, TripName, total_price, state_rank
from
(select c.State, r.TripID, TripName, sum((r.TripPrice*r.NumPersons)+r.OtherFees) as
total_price,
        rank() OVER(PARTITION BY c.State order by
sum((TripPrice*NumPersons)+OtherFees) desc) AS state_rank
from  reservation r
join customer c on r.CustomerNum=c.CustomerNum
join trip t on r.TripID=t.TripID
group by c.State, r.TripID) t
where state_rank<=3
order by State, state_rank;



/* query 9 partial result
CT      38      Sawyer River Ride       355.00  1
CT      25      Mount Battie Ride       245.00  2
CT      28      Mount Garfield Hike     115.00  3
MA      15      Crawford Path Presidentials Hike        375.00  1
MA      32      Northern Forest Canoe Trail     290.00  2
MA      26      Mount Cardigan Hike     195.00  3
*/

/* query 10. Find top guide in each month based on total number of persons led by
each guide.
             Display month, guide name, number of trips, and number of persons.
order by month */


select OneMonth, GuideName, num_trips, num_people
from
(select month(TripDate) as OneMonth, concat(FirstName, ' ', LastName) as GuideName,
count(TripID) as num_trips, sum(NumPersons) as num_people,
       rank() over(partition by month(TripDate) order by sum(NumPersons) desc) as
guide_rank
from  guide g
inner join reservation r on g.GuideNum=r.GuideNum
group by OneMonth, r.GuideNum, GuideName, NumPersons) t
where guide_rank=1
```

```
order by OneMonth;


/* query 10 partial result
3       Miles Abrams    1       2
5       Zach Gregory    1       3
6       Lori Stevens    1       5
7       Glory Unser     2       14
8       Susan Kiley     1       2
8       Harley Devon    1       2
*/
```