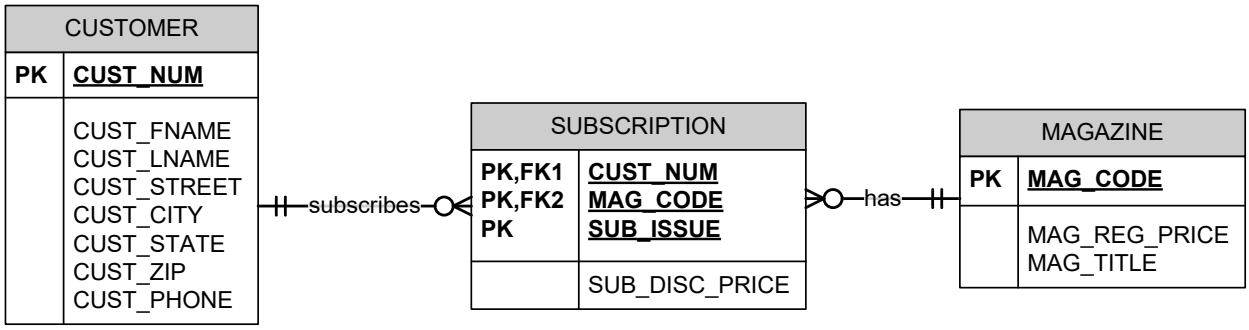
Assignment 2

INSTRUCTION:

- You should upload your completed assignment to myclasses.
- **Do it by yourself!!**

1. **Study the following conceptual data model about magazine subscription. (20 points)**



Note: If you already have a table named "CUSTOMER", you may need to use a different table name such as "CUSTOMER2".

(1) Write down three CREATE TABLE statements

- You may need to make a reasonable assumption regarding to the data type of each attribute. (See the SQL standard below)
- You must include all the **PK (entity integrity) and FK (referential integrity) constraints**. In particular, you may need to review how to handle the associative entities.

Numeric	integer	A 31-bit signed binary value
	smallint	A 15-bit signed binary value
	float(p)	A scientific format number of p binary digits precision
	decimal(p,q)	A packed decimal number of p digits total length; q decimal places to the right of the decimal point may be specified
String	char(n)	A fixed length character string of n characters
	varchar(n)	A variable length character string up to n characters
	text	A variable-length character string of up to 65,535 characters
Date/time	date	Date in the form yyyymmdd
	time	Time in the form hhmmss
	timestamp	A combination of date and time to the nearest microsecond

	time with time zone	Same as time, with the addition of an offset from UTC of the specified time
	timestamp with time zone	Same as timestamp, with the addition of an offset from UTC of the specified time

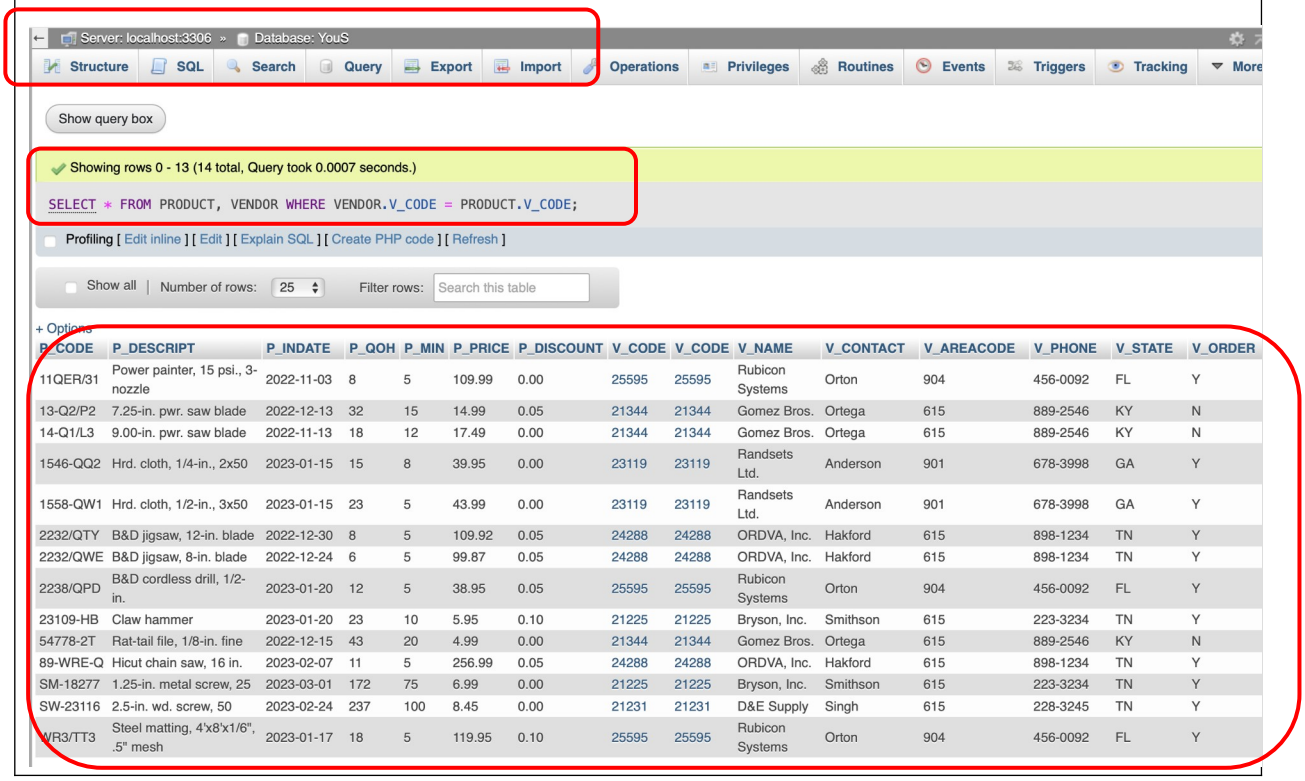
Entity name	SQL Statement
CUSTOMER	CREATE TABLE CUSTOMER2 (CUST_NUM INTEGER NOT NULL, CUST_FNAME VARCHAR(15), CUST_LNAME VARCHAR(15), CUST_STREET VARCHAR(15), CUST_CITY VARCHAR(15), CUST_STATE CHAR(2), CUST_ZIP CHAR(5), CUST_PHONE CHAR(8), PRIMARY KEY (CUST_NUM));
SUBSCRIPTIO N	CREATE TABLE SUBSCRIPTION (SUB_ISSUE VARCHAR(10) NOT NULL, SUB_DISC_PRICE INTEGER, CUST_NUM INTEGER NOT NULL, MAG_CODE INTEGER NOT NULL, PRIMARY KEY (SUB_ISSUE, CUST_NUM, MAG_CODE), FOREIGN KEY (CUST_NUM) REFERENCES CUSTOMER2(CUST_NUM), FOREIGN KEY (MAG_CODE) REFERENCES MAGAZINE(MAG_CODE));
MAGAZINE	CREATE TABLE MAGAZINE (MAG_CODE INTEGER NOT NULL, MAG_REG_PRICE INTEGER, MAG_TITLE VARCHAR(15), PRIMARY KEY (MAG_CODE));

(2) Run the SQL statements above to create the tables in your database.

- You may need to follow the sequence of creating tables, considering the foreign key constrains

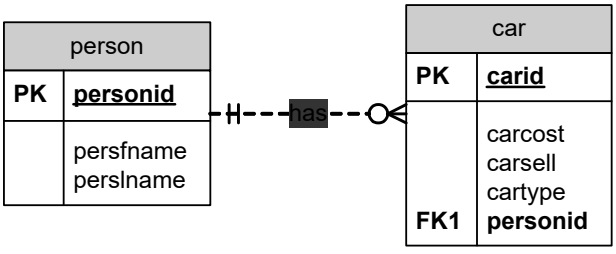
NOTE: SQL Query Problem

You must show both the query and the results of the query for each problem. This should be done by capturing screenshot from the "print view" of your query result as shown in the following sample:



P_CODE	P_DESCRIPT	P_INDATE	P_QOH	P_MIN	P_PRICE	P_DISCOUNT	V_CODE	V_CODE	V_NAME	V_CONTACT	V_AREACODE	V_PHONE	V_STATE	V_ORDER
11QER/31	Power painter, 15 psi., 3-nozzle	2022-11-03	8	5	109.99	0.00	25595	25595	Rubicon Systems	Orton	904	456-0092	FL	Y
13-Q2/P2	7.25-in. pwr. saw blade	2022-12-13	32	15	14.99	0.05	21344	21344	Gomez Bros.	Ortega	615	889-2546	KY	N
14-Q1/L3	9.00-in. pwr. saw blade	2022-11-13	18	12	17.49	0.00	21344	21344	Gomez Bros.	Ortega	615	889-2546	KY	N
1546-QQ2	Hrd. cloth, 1/4-in., 2x50	2023-01-15	15	8	39.95	0.00	23119	23119	Randsets Ltd.	Anderson	901	678-3998	GA	Y
1558-QW1	Hrd. cloth, 1/2-in., 3x50	2023-01-15	23	5	43.99	0.00	23119	23119	Randsets Ltd.	Anderson	901	678-3998	GA	Y
2232/QTY	B&D jigsaw, 12-in. blade	2022-12-30	8	5	109.92	0.05	24288	24288	ORDVA, Inc.	Hakford	615	898-1234	TN	Y
2232/QWE	B&D jigsaw, 8-in. blade	2022-12-24	6	5	99.87	0.05	24288	24288	ORDVA, Inc.	Hakford	615	898-1234	TN	Y
2238/QPD	B&D cordless drill, 1/2-in.	2023-01-20	12	5	38.95	0.05	25595	25595	Rubicon Systems	Orton	904	456-0092	FL	Y
23109-HB	Claw hammer	2023-01-20	23	10	5.95	0.10	21225	21225	Bryson, Inc.	Smithson	615	223-3234	TN	Y
54778-2T	Rat-tail file, 1/8-in. fine	2022-12-15	43	20	4.99	0.00	21344	21344	Gomez Bros.	Ortega	615	889-2546	KY	N
89-WRE-Q	Hicut chain saw, 16 in.	2023-02-07	11	5	256.99	0.05	24288	24288	ORDVA, Inc.	Hakford	615	898-1234	TN	Y
SM-18277	1.25-in. metal screw, 25	2023-03-01	172	75	6.99	0.00	21225	21225	Bryson, Inc.	Smithson	615	223-3234	TN	Y
SW-23116	2.5-in. wd. screw, 50	2023-02-24	237	100	8.45	0.00	21231	21231	D&E Supply	Singh	615	228-3245	TN	Y
WR3/TT3	Steel matting, 4'x8'x1/6", .5" mesh	2023-01-17	18	5	119.95	0.10	25595	25595	Rubicon Systems	Orton	904	456-0092	FL	Y

2. A car dealership wants to keep track of car sales. They want to track a person's first and last name, and for each car bought by the person, the wholesale cost of the car, the selling price, and the type of car. One person can buy more than one car. A data model showing entities attributes, identifiers, and the relationship between the entities is given in the following ERD.



Before answering to the questions, you should create two tables and populate the data for each table in the database (**acadweb5.salisbury.edu/phpMyAdmin**) Use the following SQL statements.

```
DROP TABLE IF EXISTS car;
```

```
DROP TABLE IF EXISTS person;

create table person
(personid integer not null,
persfname varchar(25),
perslname varchar(25),
constraint pk_person primary key (personid));

create table car
(carid integer not null,
carcost decimal(7,2),
carsell decimal(7,2),
cartype varchar(10),
personid integer,
constraint pk_car primary key (carid),
constraint fk_soldby foreign key (personid) references person(personid));

insert into person values (21,'Sheila','O''Hara');
insert into person values (2,'Gigi','Garfield');
insert into person values (63,'Barbara','Capelli');
insert into person values (74,'James','Haley');
insert into person values (5,'Nolan','Haley');
insert into person values (16,'Kwok-Kee','Tan');
insert into person values (7,'Macedonio','Gomez');
insert into person values (8,'Bruce','Bush');
insert into person values (99,'Sue','Lim');
insert into person values (10,'Kendra','Haley');

insert into car values (1,500,795,'sedan',5);
insert into car values (2,15500,14750,'coupe',8);
insert into car values (3,1255,1355,'sports',8);
insert into car values (4,950,2000,'sedan',8);
insert into car values (5,7500,9000,'sports',2);
insert into car values (6,5400,6000,'sedan',99);
insert into car values (7,10300,12000,'sedan',99);
insert into car values (8,5700,9000,'coupe',99);
insert into car values (9,9000,12000,'sports',74);
insert into car values (10,6000,6500,'sedan',63);
```

(4 points)
(1) Run the following SQL statement:

```
insert into car values (1,7000,8000,'sedan',5);
```

What is the error message? Copy and paste the error message here:

#1062 - Duplicate entry '1' for key 'PRIMARY'

In terms of **Entity Integrity Constraint**, explain the error message. Do not restate the definition. Instead, be specific with the example given above.

This error indicates a violation of entity integrity constraint, specifically related to the primary key in the table 'car'. The constraint mandates that every row in a table must be uniquely identifiable, such as with a non-null primary key. In this case, the SQL statement attempts to insert a row with the same identifying value (pk) of '1' as another row, which would compromise the primary key as a unique identifier. A duplication of a primary key, as in this example, results in an error to uphold uniqueness and entity integrity, as if this were to succeed, not only would it violate the rules of a primary key and entity integrity constraint, but it would result in an inability to identify the duplicated rows as their unique identifiers would no longer be unique. The current primary entry of '1' follows: (1,500,795,'sedan',5), such that if (1,7000,8000,'sedan',5) was added, we can not uniquely identify each entity as pk '1' is now associated with two entities of different values.

(4 points)
(2) Run the following SQL statement:

```
insert into car values (12,7000,8000,'sedan',55);
```

What is the error message? Copy and paste the error message here:

#1452 - Cannot add or update a child row: a foreign key constraint fails (`TranfagliaK`.`car`, CONSTRAINT `fk_soldby` FOREIGN KEY (`personid`) REFERENCES `person` (`personid`))

In terms of **Referential Integrity Constraint**, explain the error message. Do not restate the definition. Instead, be specific with the example given above.

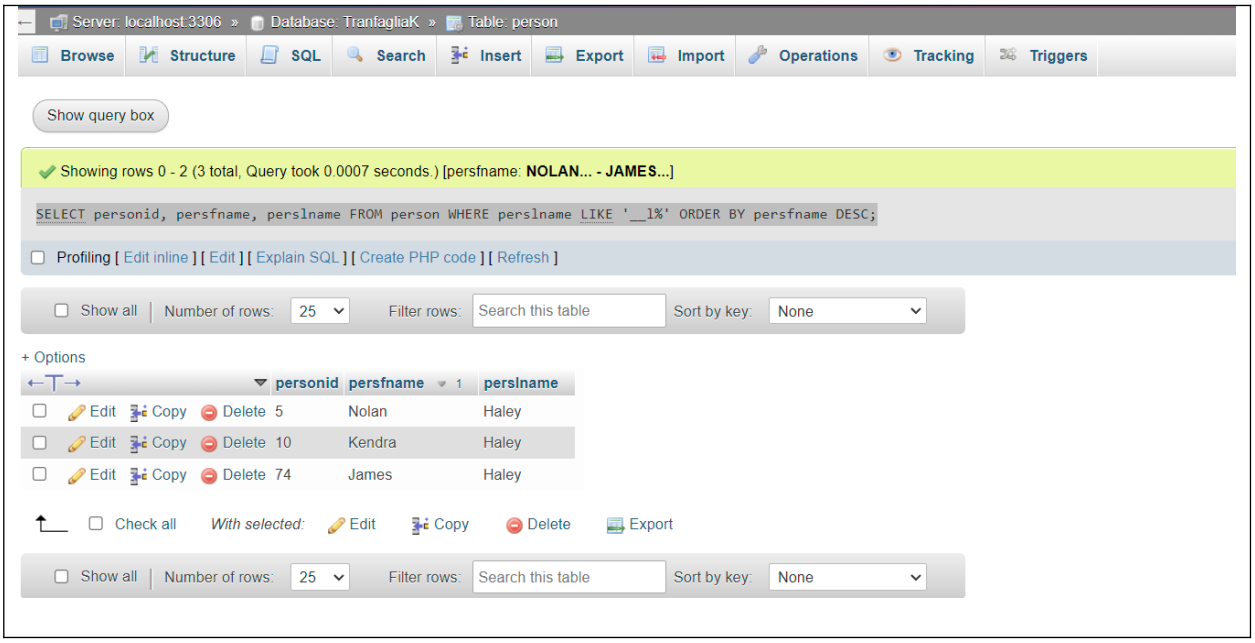
This error indicates a violation of referential integrity constraint, specifically related to the foreign key in the table 'car' in reference to the primary key of 'person'. The constraint mandates that a table maintains validity of foreign key relationships by enforcing foreign key constraints and referenced value existence. In this example, the error occurs as there is an attempt to insert the values (12,7000,8000,'sedan',55) into 'car' where '12' is the foreign key value for 'personid' (child) referencing the primary key in 'person' but the value '12' does not exist in 'person.' If this statement was successfully ran, then there would be a foreign key '12' in 'car' that does not correspond to a valid existing primary key value in 'person.' In summary, this error occurs as the value foreign key value '12' being inserted in 'car' does not match an existing primary key value in 'person,' diminishing the relationship between the tables.

(8 points)

(3) List details of sales persons (personid, persfname, perslname) with a last name has "l" as the third letter. Sort the results by the first name in descending order.

If your answer is correct, you will get the following query result:

personid	persfname	perslname
5	Nolan	Haley
10	Kendra	Haley
74	James	Haley



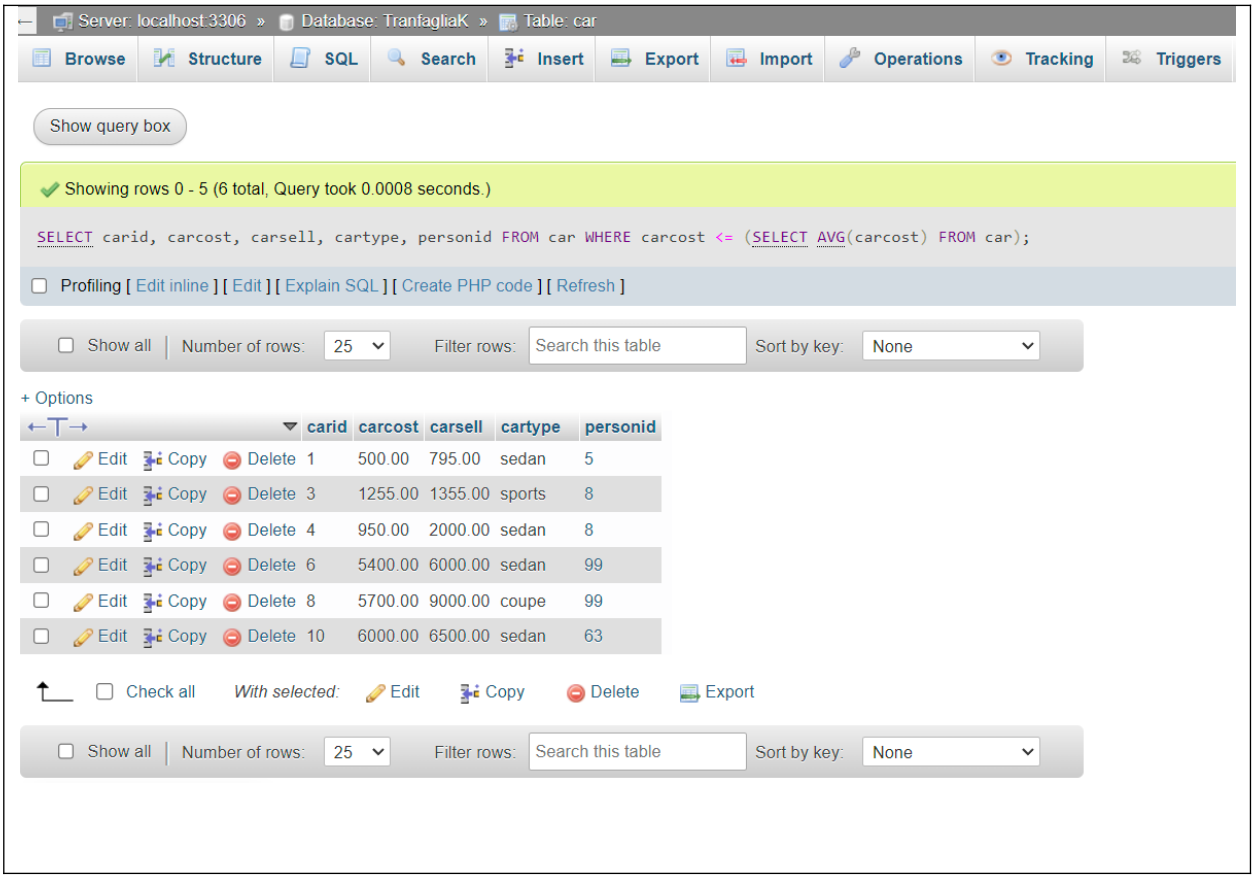
(8 points)

(4) List details of the car sales (carid, carcost, carsell, cartype) with the cost less than or equal to the average car cost (Hint: Subquery).

If your answer is correct, you will get the following query result:

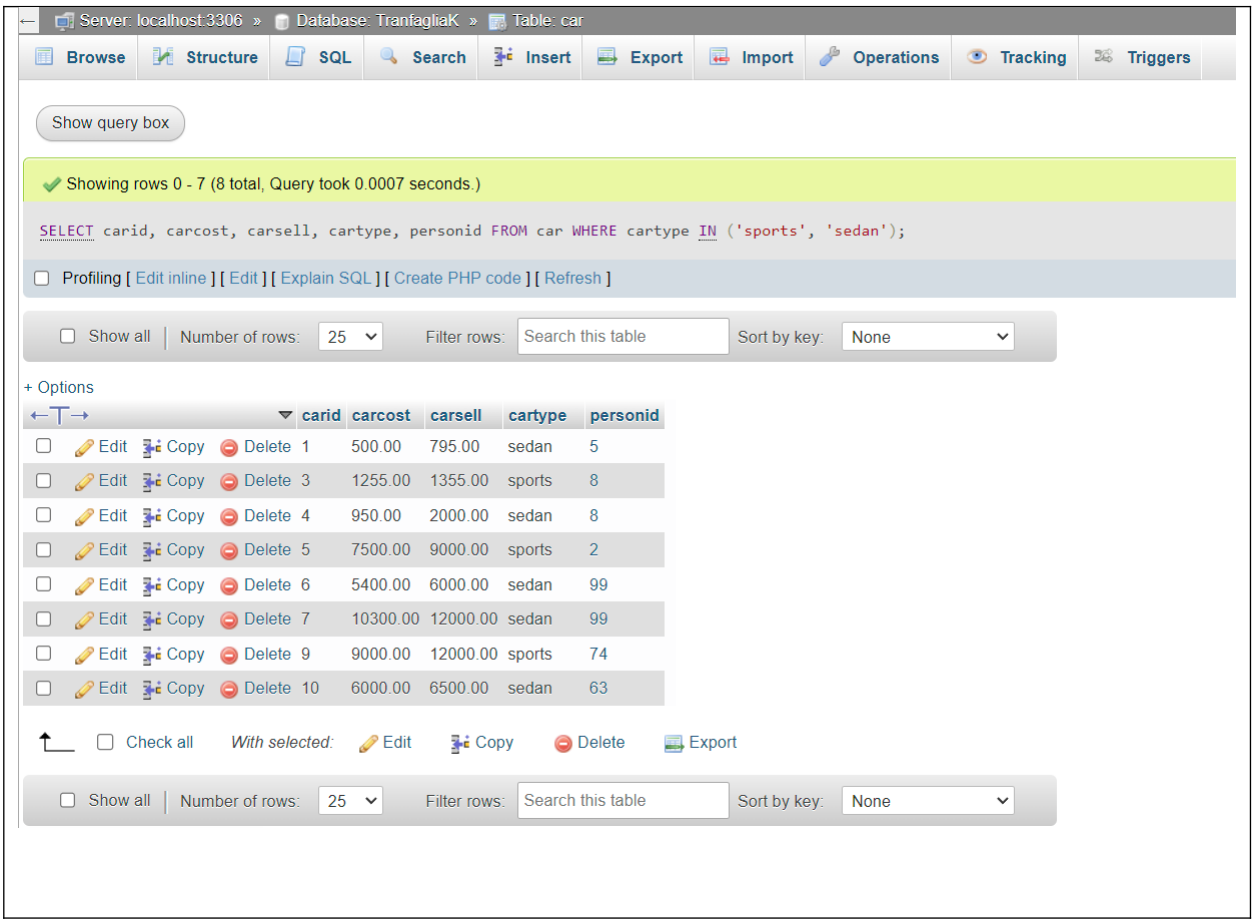
carid	carcost	carsell	cartype	personid
1	500.00	795.00	sedan	5 [->]
3	1255.00	1355.00	sports	8 [->]
4	950.00	2000.00	sedan	8 [->]
6	5400.00	6000.00	sedan	99 [->]
8	5700.00	9000.00	coupe	99 [->]

carid	carcost	carsell	cartype	personid
10	6000.00	6500.00	sedan	63 [->]



(**8 points)**
(5) List all the car sales (carid, carcost, carsell, cartype) with the car type ‘sports’ or ‘sedan’. (Hint: IN/NOT IN)
If your answer is correct, you will get the following query result:

carid	carcost	carsell	cartype	personid
1	500.00	795.00	sedan	5 [->]
3	1255.00	1355.00	sports	8 [->]
4	950.00	2000.00	sedan	8 [->]
5	7500.00	9000.00	sports	2 [->]
6	5400.00	6000.00	sedan	99 [->]
7	10300.00	12000.00	sedan	99 [->]
9	9000.00	12000.00	sports	74 [->]
10	6000.00	6500.00	sedan	63 [->]

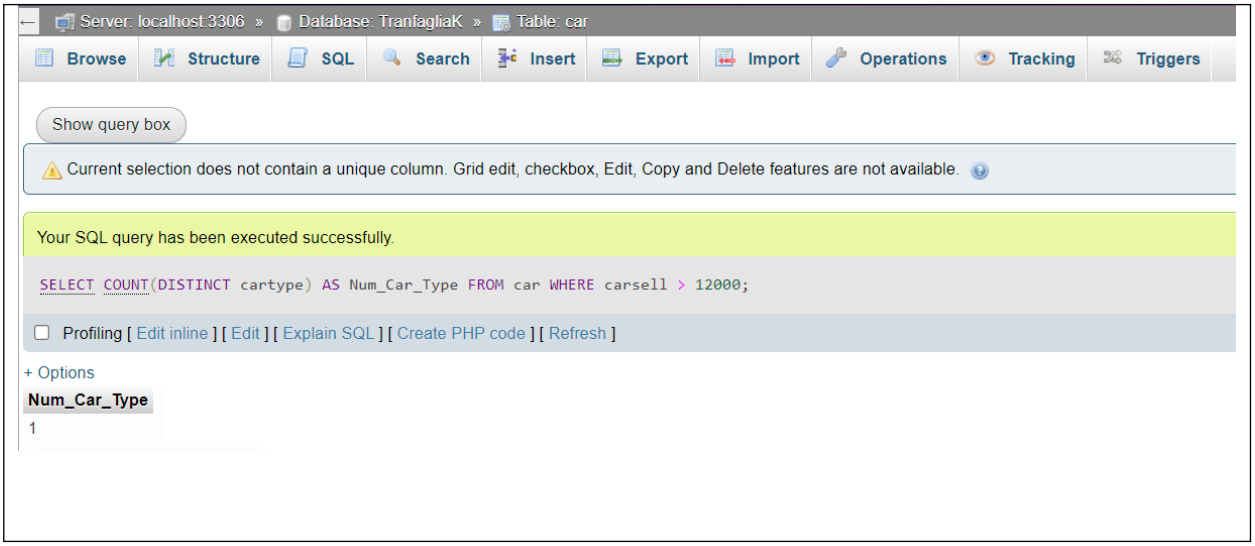


(**8 points)**

(6) Find the number of **different** car types with the selling price greater than $12,000 (Hint: DISTINCT). Use the alias "Num_Car_Type".

If your answer is correct, you will get the following query result:

Num_Car_Type
1

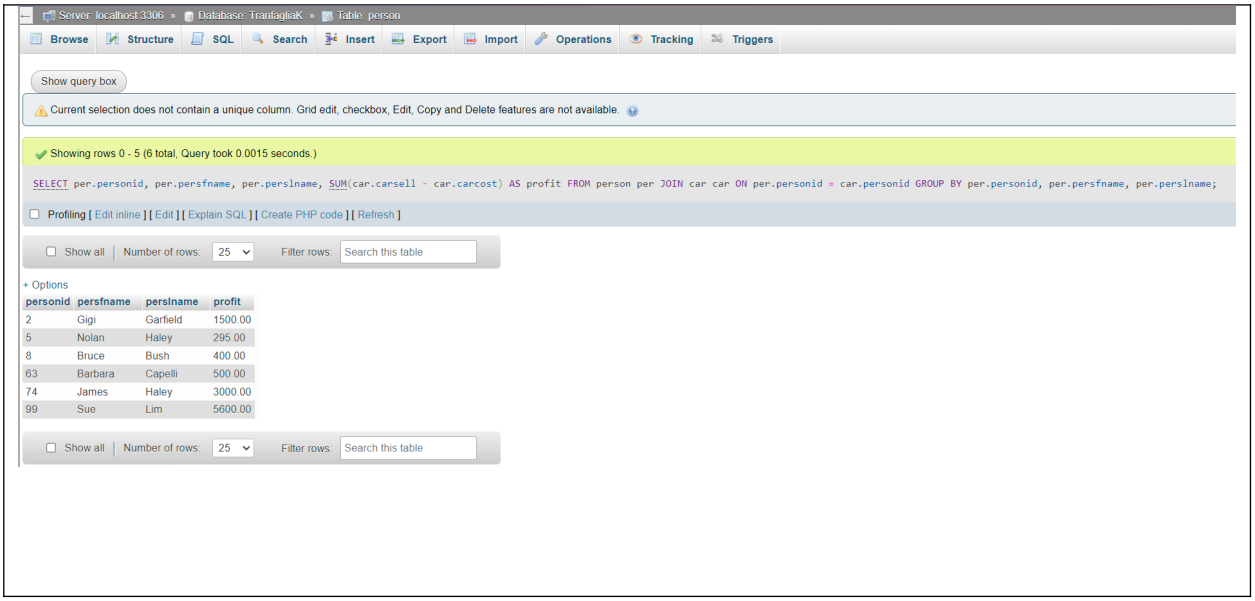

Server: localhost:3306 » Database: TranfagliaK » Table: car
Browse
Structure
SQL
Search
Insert
Export
Import
Operations
Tracking
Triggers
Show query box
Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.
Your SQL query has been executed successfully.
SELECT COUNT(DISTINCT cartype) AS Num_Car_Type FROM car WHERE carsell > 12000;
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
+ Options
Num_Car_Type
1

(**8 points)**

(7) List sales persons (personid, persfname, perslname) with the sum of gross profit (selling price minus cost price) for each person. Use the alias "profit". [Hint: JOIN, GROUP By]

If your answer is correct, you will get the following query result:

personid	persfname	perslname	profit
2 [->]	Gigi	Garfield	1500.00
5 [->]	Nolan	Haley	295.00
8 [->]	Bruce	Bush	400.00
63 [->]	Barbara	Capelli	500.00
74 [->]	James	Haley	3000.00
99 [->]	Sue	Lim	5600.00

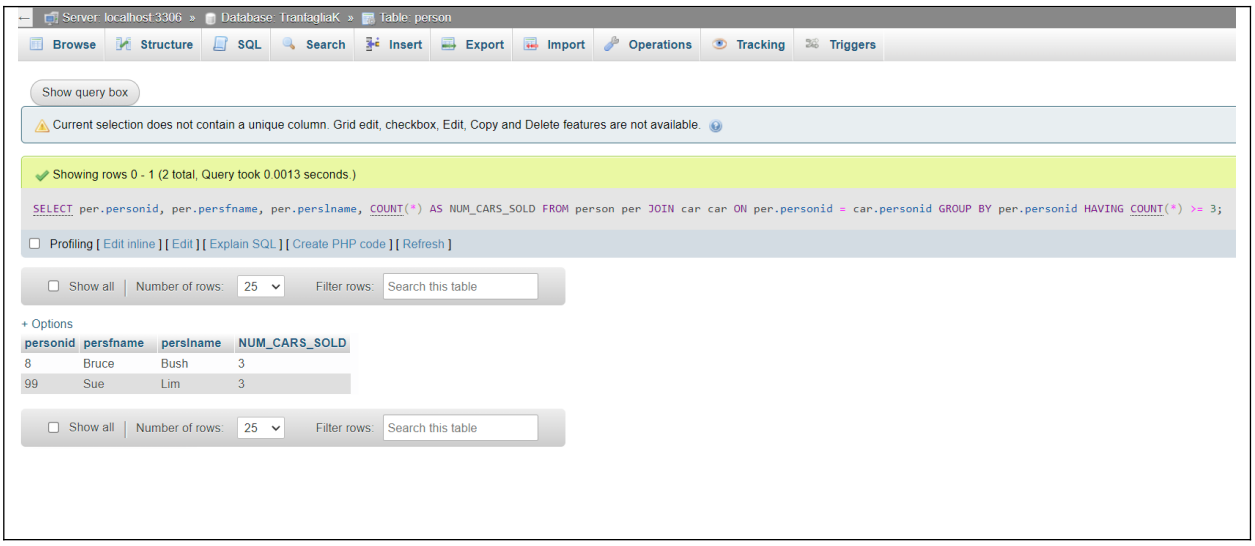


(**8 points)**

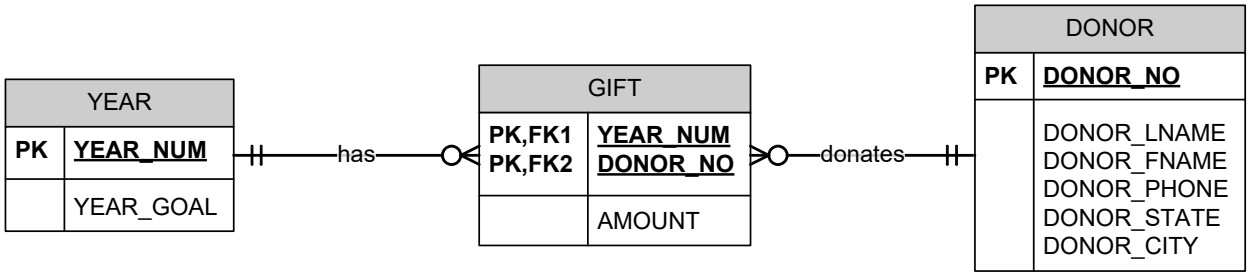
(8) Report the number of cars sold by each salesperson who sold at least three cars.

If your answer is correct, you will get the following query result:

personid	persfname	perslname	NUM_CARS_SOLD
8 [->]	Bruce	Bush	3
99 [->]	Sue	Lim	3


Server: localhost:3306 » Database: TranfagliaK » Table: person
Browse
Structure
SQL
Search
Insert
Export
Import
Operations
Tracking
Triggers
Show query box
Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.
Showing rows 0 - 1 (2 total, Query took 0.0013 seconds.)
SELECT per.personid, per.persfname, per.perslname, COUNT(*) AS NUM_CARS_SOLD FROM person per JOIN car car ON per.personid = car.personid GROUP BY per.personid HAVING COUNT(*) >= 3;
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all | Number of rows: 25 Filter rows: Search this table
+ Options
personid persfname perslname NUM_CARS_SOLD
8 Bruce Bush 3
99 Sue Lim 3
Show all | Number of rows: 25 Filter rows: Search this table

3. Answer the following questions based on the following data model.



Before answering to the questions, you should create two tables and populate the data for each table in the database (https://acadweb5.salisbury.edu/phpmyadmin). Use the following SQL statements.

```
DROP TABLE IF EXISTS GIFT;
DROP TABLE IF EXISTS YEAR;
DROP TABLE IF EXISTS DONOR;

create table DONOR
(DONOR_NO integer not null,
DONOR_LNAME varchar(15),
DONOR_FNAME varchar(15),
DONOR_PHONE decimal(4),
DONOR_STATE char(2),
DONOR_CITY varchar(15),
constraint pk_donor primary key (DONOR_NO));

create table YEAR
(YEAR_NUM integer not null,
YEAR_GOAL decimal(9),
constraint pk_year primary key (YEAR_NUM));

create table GIFT
(AMOUNT decimal(8) not null,
YEAR_NUM integer not null,
DONOR_NO integer not null,
constraint pk_gift primary key (YEAR_NUM,DONOR_NO),
constraint fk_donatedin foreign key (YEAR_NUM) references YEAR(YEAR_NUM),
constraint fk_donatedby foreign key (DONOR_NO) references DONOR(DONOR_NO));

insert into DONOR values (101,'Abrams','Louis',9018,'GA','London');
insert into DONOR values (102,'Aldinger','Dmitry',1521,'GA','Paris');
insert into DONOR values (103,'Beckman','Gulsen',8247,'WA','Sao Paulo');
insert into DONOR values (104,'Berdahl','Samuel',8149,'WI','Sydney');
insert into DONOR values (105,'Borneman','Joanna',1888,'MD','Bombay');
insert into DONOR values (106,'Brock','Scott',2142,'AL','London');
insert into DONOR values (107,'Buyert','Aylin',9355,'AK','New York');
insert into DONOR values (108,'Cetinsoy','Girwan',6346,'AZ','Rome');
insert into DONOR values (109,'Chisholm','John',4482,'MA','Oslo');
insert into DONOR values (110,'Crowder','Anthony',6513,'NC','Stockholm');
```

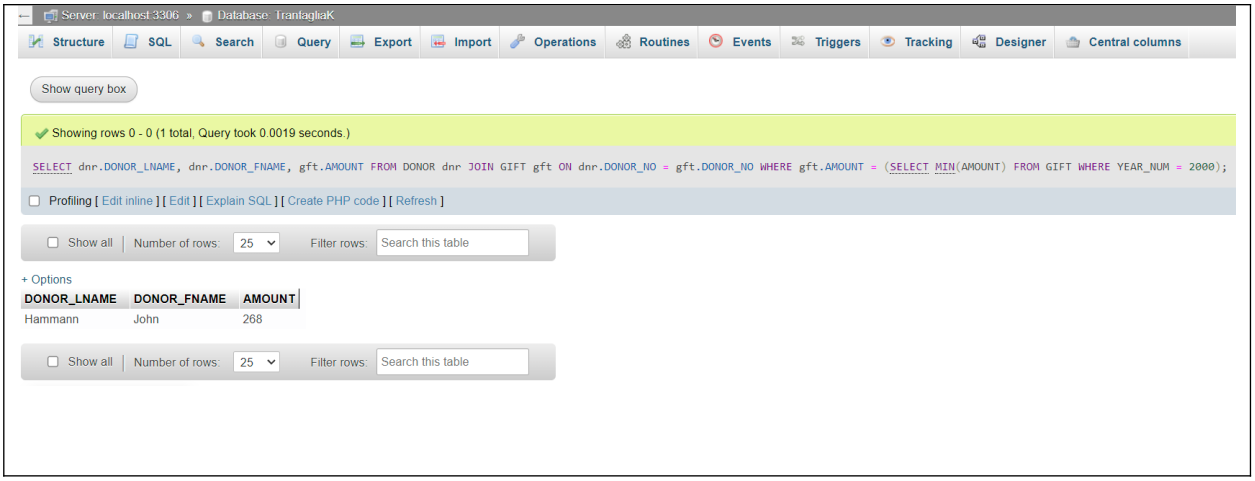
```
insert into DONOR values (111,'Dishman','Michelle',3903,'NC','Helsinki');
insert into DONOR values (112,'Duke','Peter',4939,'FL','Tokyo');
insert into DONOR values (113,'Evans','Ann',4336,'GA','Singapore');
insert into DONOR values (114,'Frawley','Todd',4785,'MN','Perth');
insert into DONOR values (115,'Guo','John',6247,'MN','Moscow');
insert into DONOR values (116,'Hammann','John',5369,'ND','Kabaul');
insert into DONOR values (117,'Hays','Cami',1352,'SD','Lima');
insert into DONOR values (118,'Herskowitz','Thomas',6872,'MT','London');
insert into DONOR values (119,'Jefts','Robert',8103,'ME','Oslo');
insert into YEAR values (1999,5000);
insert into YEAR values (2000,5000);
insert into YEAR values (2001,5500);
insert into YEAR values (2002,5000);
insert into GIFT values (939,2000,101);
insert into GIFT values (899,2000,102);
insert into GIFT values (111,2001,102);
insert into GIFT values (373,1999,101);
insert into GIFT values (543,1999,102);
insert into GIFT values (1185,1999,103);
insert into GIFT values (1362,2000,103);
insert into GIFT values (5208,2001,103);
insert into GIFT values (1865,2002,103);
insert into GIFT values (667,2000,105);
insert into GIFT values (60,2002,106);
insert into GIFT values (332,2001,107);
insert into GIFT values (674,2000,108);
insert into GIFT values (155,2001,108);
insert into GIFT values (838,1999,109);
insert into GIFT values (499,2001,109);
insert into GIFT values (582,1999,110);
insert into GIFT values (297,2000,110);
insert into GIFT values (84,2001,110);
insert into GIFT values (823,2002,110);
insert into GIFT values (887,1999,111);
insert into GIFT values (332,2000,111);
insert into GIFT values (882,2001,111);
insert into GIFT values (666,1999,112);
insert into GIFT values (812,2002,112);
insert into GIFT values (560,2001,113);
insert into GIFT values (223,1999,114);
insert into GIFT values (835,2001,114);
insert into GIFT values (558,2000,115);
insert into GIFT values (268,2000,116);
insert into GIFT values (345,2001,116);
insert into GIFT values (265,2002,116);
insert into GIFT values (82,1999,117);
insert into GIFT values (657,2002,117);
insert into GIFT values (17,2002,118);
insert into GIFT values (186,1999,119);
insert into GIFT values (772,2000,119);
```

(8 points)

(1) What is the first and last name of the person who made the smallest donation in 2000? [Hint: Join + Subquery]

If your answer is correct, you will get the following query result:

DONOR_LNAME	DONOR_FNAME	AMOUNT
Hammann	John	268

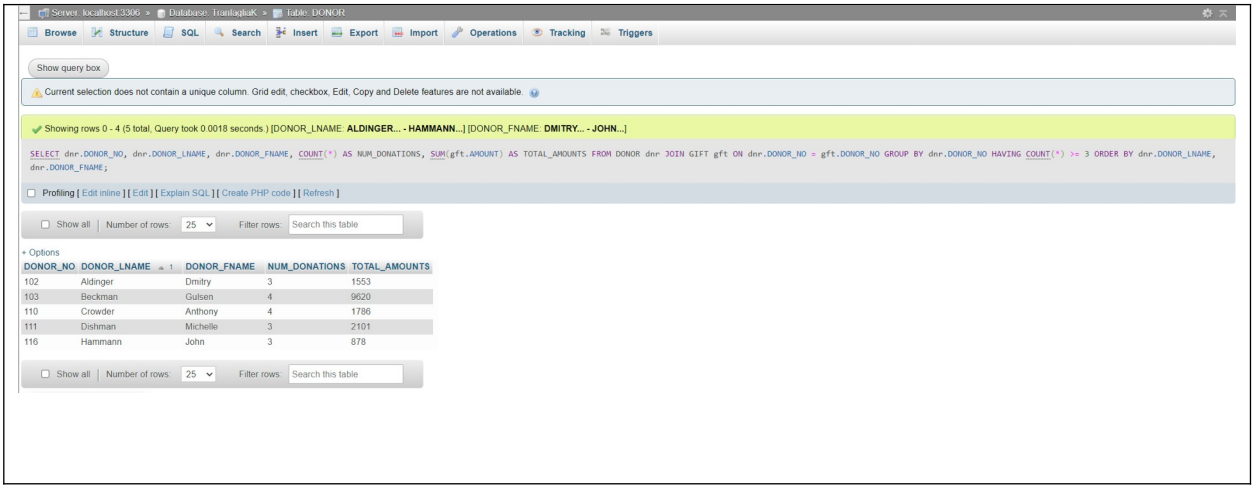


(8 points)

(2) List the donor's number, name, total number of donations, and total amount given by each donor who donates at least 3 times; sort the report by the donor's last name and first name. (Hint: GROUP BY DONOR_NO)

If your answer is correct, you will get the following query result:

DONOR_LNAME	DONOR_FNAME	NUM_DONATIONS	TOTAL_AMOUNTS
Aldinger	Dmitry	3	1553
Beckman	Gulsen	4	9620
Crowder	Anthony	4	1786
Dishman	Michelle	3	2101
Hammann	John	3	878


Server: localhost:3306 » Database: franlagleK » Table: DONOR
Browse
Structure
SQL
Search
Insert
Export
Import
Operations
Tracking
Triggers
Show query box
Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.
Showing rows 0 - 4 (5 total, Query took 0.0018 seconds.) [DONOR_LNAME: ALDINGER... - HAMMANN...] [DONOR_FNAME: DMITRY... - JOHN...]
SELECT dnr.DONOR_NO, dnr.DONOR_LNAME, dnr.DONOR_FNAME, COUNT(*) AS NUM_DONATIONS, SUM(gft.AMOUNT) AS TOTAL_AMOUNTS FROM DONOR dnr JOIN GIFT gft ON dnr.DONOR_NO = gft.DONOR_NO GROUP BY dnr.DONOR_NO HAVING COUNT(*) >= 3 ORDER BY dnr.DONOR_LNAME, dnr.DONOR_FNAME;
Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]
Show all
Number of rows: 25
Filter rows: Search this table
+ Options
DONOR_NO DONOR_LNAME DONOR_FNAME NUM_DONATIONS TOTAL_AMOUNTS
102 Aldinger Dmitry 3 1553
103 Beckman Gulsen 4 9620
110 Crowder Anthony 4 1786
111 Dishman Michelle 3 2101
116 Hammann John 3 878
Show all
Number of rows: 25
Filter rows: Search this table

(8 points)

(3) In which year, was the goal NOT achieved? Report year, year goal, and total donated amounts for that year.

[Hint: Correlated subquery. You may want to modify the following query:

```
SELECT YEAR.YEAR_NUM, YEAR_GOAL, SUM(AMOUNT)
FROM YEAR,GIFT
WHERE YEAR.YEAR_NUM = GIFT.YEAR_NUM
GROUP BY YEAR.YEAR_NUM
```

]

If your answer is correct, you will get the following query result:

YEAR_NUM	YEAR_GOAL	SUM(AMOUNT)
2002 [->]	5000	4499