

# ITCS 393 Database System Lab

## Assignment #6

Solve the following 7 questions using SQL SET Operations

**a. Write a query to show SlipID that does not request for service.**

The screenshot shows a SQL Developer window with a query editor and a results pane. The query editor contains the following SQL code:

```
USE ALEXAMARA;

-- Name: Mangkhaes Ngamjaruskotchakorn
-- ID: 6188055 Section: 1 Date: 3/8/2021

-- Write a query to show SlipID that does not request for service
SELECT s.SlipID, T.ServiceID
FROM MarinaSlip s
OUTER APPLY
(
    SELECT sr.ServiceID
    FROM ServiceRequest sr
    WHERE s.SlipID = sr.SlipID AND sr.ServiceID IS NULL
) AS T;

-- Write a query to display Distinct Status using SET operation WHERE Status is not "Open".
SELECT DISTINCT Status
FROM ServiceRequest
EXCEPT
(
    SELECT Status
    FROM ServiceRequest
    WHERE Status = 'Open'
);

-- List all distinct City and State of Owner using SET operation in Descending Order.
SELECT DISTINCT City, T.State
```

The results pane shows the output of the first query, displaying a list of SlipIDs and their corresponding ServiceIDs. The results are as follows:

SlipID	ServiceID
1	NULL
2	NULL
3	NULL
4	NULL
5	NULL
6	NULL
7	NULL
8	NULL
9	NULL
10	NULL
11	NULL

b. Write a query to display Distinct Status using SET operation WHERE Status is not "Open".

U6188055\_Assignm...EXAMARA (sa (51))\* SQLQuery1.sql - D...LEXAMARA (sa (60))\*

```

SELECT s.SlipID, T.ServiceID
FROM MarinaSlip s
OUTER APPLY
(
    SELECT sr.ServiceID
    FROM ServiceRequest sr
    WHERE s.SlipID = sr.SlipID AND sr.ServiceID IS NULL
) AS T;

-- Write a query to display Distinct Status using SET operation WHERE Status is not "Open".
SELECT DISTINCT Status
FROM ServiceRequest
EXCEPT
(
    SELECT Status
    FROM ServiceRequest
    WHERE Status = 'Open'
);

-- List all distinct City and State of Owner using SET operation in Descending Order.
SELECT DISTINCT City, T.State
FROM Owner
CROSS APPLY
(
    SELECT DISTINCT State
    FROM Owner
) AS T;
  
```

100 %

Results Messages

	Status
1	Overlap has been created. Installation has been sc...
2	Scheduled
3	Service call has been scheduled.
4	Technician has completed the initial filling of the op...
5	Technician has determined that the exchanger is fa...
6	Technician has scheduled appointment with owner ...
7	Technician has scheduled repair.
8	Technician has verified the problem. Air conditionin...
9	Technician not able to replicate the problem. Air co...
10	Technician suspects problem with either propeller or...

c. List all distinct City and State of Owner using SET operation in Descending Order.

```
U6188055_Assignm...EXAMARA (sa (51))* SQLQuery1.sql - D...LEXAMARA (sa (60))*  
FROM ServiceRequest  
WHERE Status = 'Open'  
);  
  
-- List all distinct City and State of Owner using SET operation in Descending Order.  
SELECT DISTINCT City, T.State  
FROM Owner  
CROSS APPLY  
(  
    SELECT DISTINCT State  
    FROM Owner  
) AS T  
ORDER BY City DESC;  
  
-- For each Owner and Marina, list all City using SET operation in Ascending Order  
SELECT OwnerNum as NumID, City  
FROM Owner  
UNION  
SELECT MarinaNum, City  
FROM Marina  
ORDER BY OwnerNum;  
  
-- For each Owner, get additional column with SlipID and BoatType using APPLY operator  
SELECT o.FirstName + ' ' + o.LastName as OwnerName,  
T.SlipID,  
T.BoatType  
FROM Owner o  
CROSS APPLY (SELECT * FROM ServiceRequest WHERE OwnerNum = o.OwnerNum) T
```

100 %

Results Messages

	City	State
1	Rivard	FL
2	Lewiston	FL
3	Kaleva	FL
4	Glander Bay	FL
5	Bowton	FL
6	Bay Shores	FL

d. For each Owner and Marina, list all City using SET operation in Ascending Order.

```
U6188055_Assignm...EXAMARA (sa (51))* SQLQuery1.sql - D...LEXAMARA (sa (60))*  
SELECT DISTINCT State  
FROM Owner  
) AS T  
ORDER BY City DESC;  
  
-- For each Owner and Marina, list all City using SET operation in Ascending Order  
SELECT OwnerNum as NumID, City  
FROM Owner  
UNION  
SELECT MarinaNum, City  
FROM Marina  
ORDER BY OwnerNum;  
  
-- For each Owner, get additional column with SlipID and BoatType using APPLY operator  
SELECT o.FirstName + ' ' + o.LastName as OwnerName,  
T.SlipID,  
T.BoatType  
FROM Owner o  
OUTER APPLY  
(  
    SELECT SlipID, BoatType  
    FROM MarinaSlip s  
    WHERE o.OwnerNum = s.OwnerNum  
) AS T;  
  
-- For each Marina, get Name of Marina and Max rental fee  
SELECT Name, MAX(T.RentalFee) as MaxRentalFee
```

100 %

Results Messages

	NumID	City
1	1	Brinman
2	2	W Brinman
3	AD57	Bowton
4	AN75	Glander Bay
5	BL72	Bowton
6	EL25	Rivard
7	FE82	Kaleva
8	JU92	Rivard
9	KE22	Bowton
10	NO27	Lewiston
11	SM72	Glander Bay
12	TR72	Bay Shores

e. For each Owner, get additional column with SlipID and BoatType using APPLY operator.

U6188055\_Assignm...EXAMARA (sa (51))\* SQLQuery1.sql - D...LEXAMARA (sa (60))\*

```

SELECT OwnerNum as NumID, City
FROM Owner
UNION
SELECT MarinaNum, City
FROM Marina
ORDER BY OwnerNum;

-- For each Owner, get additional column with SlipID and BoatType using APPLY operator
SELECT o.FirstName + ' ' + o.LastName as OwnerName,
T.SlipID,
T.BoatType
FROM Owner o
OUTER APPLY
(
    SELECT SlipID, BoatType
    FROM MarinaSlip s
    WHERE o.OwnerNum = s.OwnerNum
) AS T;

-- For each Marina, get Name of Marina and Max rental fee
SELECT m.Name, MAX(T.RentalFee) as MaxRentalFee
FROM Marina m
OUTER APPLY
(
    SELECT RentalFee
    FROM MarinaSlip s
    WHERE m.MarinaNum = s.MarinaNum
) AS T;

```

100 %

Results Messages

	OwnerName	SlipID	Boat Type
1	Bruce and Jean Adney	6	Dolphin 25
2	Bill Anderson	1	Sprite 4000
3	Bill Anderson	5	Sprite 3000
4	Mary Blake	9	Dolphin 28
5	Sandy and Bill Elend	2	Ray 4025
6	Daniel Feenstra	7	Dolphin 22
7	Maria Juarez	4	Dolphin 28
8	Alyssa Kelly	3	Sprite 4000
9	Peter Norton	10	Dolphin 40
10	Becky and Dave Smeltz	8	Dolphin 25
11	Ashton Trent	11	Ray 4025

f. For each Marina, get Name of Marina and Max rental fee.

U6188055\_Assignm...EXAMARA (sa (51))\* SQLQuery1.sql - D...LEXAMARA (sa (60))\*

```
OUTER APPLY
(
    SELECT SlipID, BoatType
    FROM MarinaSlip s
    WHERE o.OwnerNum = s.OwnerNum
) AS T;

-- For each Marina, get Name of Marina and Max rental fee
SELECT m.Name, MAX(T.RentalFee) as MaxRentalFee
FROM Marina m
OUTER APPLY
(
    SELECT RentalFee
    FROM MarinaSlip s
    WHERE m.MarinaNum = s.MarinaNum
) AS T
GROUP BY m.Name;

-- For every Slip, list the marina number, slip number, boat name, owner numbe
SELECT s.SlipID, s.MarinaNum, s.SlipNum, s.BoatName,
T.OwnerNum, T.FirstName, T.LastName
FROM MarinaSlip s
CROSS APPLY
(
    SELECT OwnerNum, FirstName, LastName
    FROM Owner o
    WHERE o.OwnerNum = s.OwnerNum
)
```

100 %

Results Messages

	Name	MaxRentalFee
1	Alexamara Central	4200.00
2	Alexamara East	3800.00

*g. For every Slip, list the marina number, slip number, boat name, owner number, owner's first name, and owner's last name.*

U6188055\_Assignm...EXAMARA (sa (51))\* SQLQuery1.sql - D...LEXAMARA (sa (60))\*

```

FROM Marina m
OUTER APPLY
(
    SELECT RentalFee
    FROM MarinaSlip s
    WHERE m.MarinaNum = s.MarinaNum
) AS T
GROUP BY m.Name;

-- For every Slip, list the marina number, slip number, boat name, owner number, owner's f
SELECT s.SlipID, s.MarinaNum, s.SlipNum, s.BoatName,
T.OwnerNum, T.FirstName, T.LastName
FROM MarinaSlip s
CROSS APPLY
(
    SELECT OwnerNum, FirstName, LastName
    FROM Owner o
    WHERE s.OwnerNum = o.OwnerNum
) AS T;

```

100 %

Results Messages

	SlipID	MarinaNum	SlipNum	BoatName	OwnerNum	FirstName	LastName
1	1	1	A1	Anderson II	AN75	Bill	Anderson
2	2	1	A2	Our Toy	EL25	Sandy and Bill	Elend
3	3	1	A3	Escape	KE22	Alyssa	Kelly
4	4	1	B1	Gypsy	JU92	Maria	Juarez
5	5	1	B2	Anderson III	AN75	Bill	Anderson
6	6	2	1	Bravo	AD57	Bruce and Jean	Adney
7	7	2	2	Chinook	FE82	Daniel	Feenstra
8	8	2	3	Listy	SM72	Becky and Dave	Smeltz
9	9	2	4	Mermaid	BL72	Mary	Blake
10	10	2	5	Axxon II	NO27	Peter	Norton
11	11	2	6	Karvel	TR72	Ashton	Trent