Mid2 Solution

Exercise 1:

Consider the two following tables T1 and T2 .

P	Q	R
10	**a**	**5**
15	**b**	**8**
25	**a**	**6**

A	B	C
10	b	6
25	c	3
10	b	5

Show the results of the following operations:

a) $T1 \bowtie_{T1.P = T2.A} T2$ b) $T1 \bowtie_{T1.Q = T2.B} T2$

c) T1 U T2 d) $T1 \bowtie_{(T1.P = T2.A \textbf{ AND } T1.R = T2.C)} T2$

(a)	(b)
P Q R A B C 10 a 5 10 b 6 10 a 5 10 b 5 25 a 6 25 c 3	P Q R A B C 15 b 8 10 b 6 15 b 8 10 b 5
(c)	(d)
P Q R 10a 5 15 b 8 25 a 6 10b 6 25 c 3 10b 5	P Q R A B C 10 a 5 10 b 5

Question 2:

Consider the following relations:

Student(id, name, address, major)
Course(code, title)
Registered(id,code)

1. The titles of courses for which no student is registered.

π_{name} ((π_{code} (Course) - π_{code} (Registered)) ∞ Course)

2. Names of students and the titles of courses they registered to.

$\pi_{name,title}$ (Student ∞ Registered ∞ Course)

3. Ids of students who are registered for ‘Database Systems’ or ‘Operating Systems’.

π_{ssn} (Student ∞ Registered ∞ ($\sigma_{title='Database\ Systems'}$ Course)) ∪
π_{ssn} (Student ∞ Registered ∞ ($\sigma_{title='Operating\ Systems'}$ Course))

4. List of courses in which all students are registered.

$\pi_{code,\ id}$ (Registered) / π_{id} (Student)

Question 3:

Consider the Sailors-Boats-Reserves DB:
Sailors(sid: integer, sname: string, rating: integer, age: real)
Boats(bid: integer, bname: string, color: string)
Reserves(sid: integer, bid: integer, day: date)

Write the following queries in SQL :

1. *Find the names of sailors who have reserved at least one boat.*

```
SELECT sname
FROM Sailors S, Reserves R
WHERE S.sid = R.sid
```

2. *Find the names and ratings of sailor whose rating is better than some sailor called “Ali”.*

```
SELECT S.sname, S.rating
FROM Sailors S
WHERE S.rating > ANY ( SELECT S2.rating
FROM Sailors S2
WHERE S2.sname = ‘Ali’ )
```

3. *Find the name and the age of the youngest sailor.*

```
SELECT S.sname, S.age
FROM Sailors S
WHERE S.age = (SELECT MIN(S2.age)
FROM Sailors S2 )
```

4. *Find the average age of sailors for each rating level that has at least two sailors.*

```
SELECT S.rating, AVG(S.age) AS avg_age
FROM Sailors S
GROUP BY S.rating
HAVING COUNT(*) > 1
```