1 Examples from Week 1

Given three sets:

$D_1 = \{1, 2, 3\}$,

$D_2 = \{4, 5\}$,

$D_3 = \{6, 7\}$

Their *cartesian product*, written $D_1 \times D_2 \times D_3$ is a set of 3-tuples:

$D_1 \times D_2 \times D_3 = \{(1,4,6), (1,4,7), (1,5,6), (1,5,7), (2,4,6), (2,4,7), (2,5,6), (2,5,7),$
$(3,4,6), (3,4,7), (3,5,6), (3,5,7)\}$

Note that the size of $D_1 \times D_2 \times D_3$, is the product of the sizes of D_1, D_2, D_3 i.e. it is $3 \times 2 \times 2 = 12$

A *relation*, is a subset of the cartesian product of a number of sets $D_1, D_2, \ldots, D_n$. Relations can be viewed as *tables* e.g. for the above example:

Col_1	Col_2	Col_3
1	4	6
1	4	7
1	5	6
1	5	7
2	4	6
2	4	7
2	5	6
2	5	7
3	4	6
3	4	7
3	5	6
3	5	7

Don't confuse the mathematical operator "cartesian product" with the relational algebra operator "product". The latter forms the cartesian product of two relations and "flattens" each pair of tuples into a single tuple of atomic values.

For example, give these two tables:

Students		
sid	name	age
5000	Mary Brown	25
6000	Joe Smith	30

Enrolled		
sid	cid	examMark
5000	BSCDB1	35
5000	BSCOPS	55
5000	BSCHCI	60
6000	BSCDB2	38
6000	BSCINT	70

Their *product*, Students × Enrolled, is:

sid	name	age	sid	cid	examMark
5000	Mary Brown	25	5000	BSCDB1	35
5000	Mary Brown	25	5000	BSCOPS	55
5000	Mary Brown	25	5000	BSCHCI	60
5000	Mary Brown	25	6000	BSCDB2	38
5000	Mary Brown	25	6000	BSCINT	70
6000	Joe Smith	30	5000	BSCDB1	35
6000	Joe Smith	30	5000	BSCOPS	55
6000	Joe Smith	30	5000	BSCHCI	60
6000	Joe Smith	30	6000	BSCDB2	38
6000	Joe Smith	30	6000	BSCINT	70

The *theta join* Students $\bowtie_C$ Enrolled with the condition $C = (age > 25)\ OR\ (mark > 60)$ is the following subset of their product:

sid	name	age	sid	cid	examMark
5000	Mary Brown	25	6000	BSCINT	70
6000	Joe Smith	30	5000	BSCDB1	35
6000	Joe Smith	30	5000	BSCOPS	55
6000	Joe Smith	30	5000	BSCHCI	60
6000	Joe Smith	30	6000	BSCDB2	38
6000	Joe Smith	30	6000	BSCINT	70

The *equi-join* Students $\bowtie_{Student.sid=Enrolled.sid}$ Enrolled is the following subset of their product:

sid	name	age	sid	cid	examMark
5000	Mary Brown	25	5000	BSCDB1	35
5000	Mary Brown	25	5000	BSCOPS	55
5000	Mary Brown	25	5000	BSCHCI	60
6000	Joe Smith	30	6000	BSCDB2	38
6000	Joe Smith	30	6000	BSCINT	70

Their *natural join* over their common column "sid" is similar, but without the duplicated "sid" column:

sid	name	age	cid	examMark
5000	Mary Brown	25	BSCDB1	35
5000	Mary Brown	25	BSCOPS	55
5000	Mary Brown	25	BSCHCI	60
6000	Joe Smith	30	BSCDB2	38
6000	Joe Smith	30	BSCINT	70

Relational Algebra Exercises

Given the relations Students, Courses and Enrolled above, write relational algebra queries which:

1. Find the name of each student who is enrolled on either course BSCDB1 or course BSCDB2:

   $\pi_{name}\ ((\sigma_{cid='BSCDB1'\ OR\ cid='BSCDB2'} Enrolled) \bowtie Students)$

   OR

   $\pi_{name}\ ((\sigma_{cid='BSCDB1'} Enrolled) \bowtie Students) \cup \pi_{name}\ ((\sigma_{cid='BSCDB2'} Enrolled) \bowtie Students)$

2. Find the id of each student who is enrolled on both course BSCDB1 and course BSCDB2.

   $\pi_{E1.sid}\ (\sigma_{E1.cid='BSCDB1'}(\rho(E1(1 \rightarrow E1.sid, 2 \rightarrow E1.cid), Enrolled)) \bowtie_{E1.sid=E2.sid}$
   $\sigma_{E2.cid='BSCDB2'}(\rho(E2(1 \rightarrow E2.sid, 2 \rightarrow E2.cid), Enrolled)))$

   OR

   $\pi_{name}\ ((\sigma_{cid='BSCDB1'} Enrolled) \bowtie Students)\ \cap$
   $\pi_{name}\ ((\sigma_{cid='BSCDB2'} Enrolled) \bowtie Students)$

SQL Exercises Given the relations Students, Courses and Enrolled above, write SQL queries which:

1. Find the name of each student who is enrolled on either course BSCDB1 or course BSCDB2.

```
SELECT DISTINCT name
FROM   Students, Enrolled
WHERE  Enrolled.sid = Students.sid AND
       (Enrolled.cid = 'BSCDB1' OR
        Enrolled.cid = 'BSCDB2')
```

2. Find the name of each student who is enrolled on both course BSCDB1 and course BSCDB2.

```
SELECT name
FROM   Students, Enrolled
WHERE  cid = 'BSCDB1' AND
       Students.sid = Enrolled.sid AND EXISTS
```

```
          (SELECT *
           FROM  Enrolled E2
           WHERE cid = 'BSCDB2' AND
                 E2.sid = Students.sid)
```

OR:

```
SELECT name
FROM Students
WHERE sid IN
         (SELECT sid
          FROM Enrolled
          WHERE cid = 'BSCDB1'
          INTERSECT
          SELECT sid
          FROM Enrolled
          WHERE cid = 'BSCDB2')
```

OR:

```
SELECT name
FROM   Students, Enrolled E1, Enrolled E2
WHERE  E1.sid = Students.sid AND
       E2.sid = Students.sid AND
       E1.cid = 'BSCDB1' AND
       E2.cid = 'BSCDB2'
```