

## Part 1 - Joins

1.

$$T1 \bowtie_{T1.A=T2.A} T2$$

A	Q	R	A	B	C
20	a	5	20	b	6
20	a	5	20	b	5

2.

$$T1 \bowtie_{T1.Q=T2.B} T2$$

A	Q	R	A	B	C
25	b	8	20	b	6
25	b	8	20	b	5

3.

$$T1 \bowtie T2$$

A	Q	R	B	C
20	a	5	b	6
20	a	5	b	5

4.

$$T1 \bowtie_{T1.A=T2.A \& \& T1.R=T2.C} T2$$

A	Q	R	A	B	C
20	a	5	20	b	5

## Part 2 - Chess Queries

1. Find the names of any player with an Elo rating of 2850 or higher.

$\pi_{\text{name}}(\sigma_{\text{Elo} \geq 2850}(\text{Players}))$

2. Find the names of any player who has ever played a game as white.

$\pi_{\text{name}}(\text{Players} \bowtie_{\text{wpID}=\text{pID}} \text{Games})$

3. Find the names of any player who has ever *won* a game as white.

$\pi_{\text{name}}(\sigma_{\text{Result}="1-0"}(\text{Players} \bowtie_{\text{wpID}=\text{pID}} \text{Games}))$

4. Find the names of any player who played any games in 2018.

$\pi_{\text{name}}(\sigma_{\text{year}=2018}(\text{Players} \bowtie_{\text{wpID}=\text{pID} \vee \text{bpID}=\text{pID}} (\text{Games} \bowtie \text{Events})))$

5. Find the names and dates of any event in which Magnus Carlsen lost a game.

$\rho(\text{blackLosses}, \pi_{\text{elID}}(\sigma_{\text{Name}="Magnus Carlsen"}(\text{Players} \bowtie_{\text{bpID}=\text{pID} \wedge \text{result}="1-0"} \text{Games})))$

$\rho(\text{whiteLosses}, \pi_{\text{elID}}(\sigma_{\text{Name}="Magnus Carlsen"}(\text{Players} \bowtie_{\text{wpID}=\text{pID} \wedge \text{result}="0-1"} \text{Games})))$

$\pi_{\text{Name}, \text{year}}(\text{Events} \bowtie (\text{blackLosses} \cup \text{whiteLosses}))$

6. Find the names of all opponents of Magnus Carlsen. An opponent is someone who he has played a game against. **Hint:** Both Magnus and his opponents could play as white or black.

$\rho(\text{blackIDs}, \pi_{\text{bpID}}(\sigma_{\text{Name}="Magnus Carlsen"}(\text{Players} \bowtie_{\text{pID}=\text{wpID}} \text{Games})))$

$\rho(\text{whiteIDs}, \pi_{\text{wpID}}(\sigma_{\text{Name}="Magnus Carlsen"}(\text{Players} \bowtie_{\text{pID}=\text{bpID}} \text{Games})))$

$\rho(\text{blackNames}, \pi_{\text{Name}}(\text{Players} \bowtie_{\text{bpID}=\text{pID}} \text{blackIDs}))$

$\rho(\text{whiteNames}, \pi_{\text{Name}}(\text{Players} \bowtie_{\text{wpID}=\text{pID}} \text{whiteIDs}))$

$\pi_{\text{Name}}(\text{blackNames} \bowtie \text{whiteNames})$

## Part 3 - LMS Queries

1.

a) Provide the relation that is the result of the following query. Your relation should be in the form of a table, and should include the schema.

$$\rho(C, \pi_{sID}(\sigma_{Grd=C}(Enrolled))) \\ \pi_{Name}((\pi_{sID}(Enrolled) - C) \bowtie Students)$$

Name
Hermione
Harry

b) Provide a simple English description of what the query is searching for. Your description should be in general terms (remember that the original LMS instance data may change).

Searching for all students that don't have a "C" as any of their grades.

2.

a) Provide the relation that is the result of the following query. Your relation should be in the form of a table, and should include the schema.

$$\rho(S1, Students) \\ \rho(S2, Students) \\ \pi_{S2.Name}(\sigma_{S1.Name==Ron \& \& S1.DOB==S2.DOB \& \& S2.Name!=Ron}(S1 \times S2))$$

Name
Hermione

b) Provide a simple English description of what the query is searching for. Your description should be in general terms (remember that the original LMS instance data may change).

Searching for all students with the same DOB value as Ron's, not including Ron (the query could be changed to be more general, but right now 'Ron' is hardcoded).

3.

a) Provide the relation that is the result of the following query. Your relation should be in the form of a table, and should include the schema.

$$\pi_{Name}((\pi_{cID, sID}(Enrolled) / \pi_{sID}(Students)) \bowtie Courses)$$

Name

b) Provide a simple English description of what the query is searching for. Your description should be in general terms (remember that the original LMS instance data may change).

Searching for all classes that have every student in the students table enrolled in them. There are no such classes in this case.

4. Provide a relational algebra query that uses the divide operator to find the names of all students who are taking all of the 3xxx-level classes.

$$\rho(3000lv, \pi_{cID}(\pi_{cID}(\sigma_{Name==3500}(Enrolled)) \bowtie \pi_{cID}(\sigma_{Name==3810}(Enrolled))))$$

$$\pi_{Name}((\pi_{sID, cID}(Enrolled) / 3000lv) \bowtie (Students))$$