אלגברה רלציונית

לינוי בושרי

members (uid, name, occupation, birthPlace, gender, educatedAt, language, birthYear)

knessets (number, startYear, endYear)

memberInKnesset (number, uid, party)

In this exercise, "S = …" means I rename the relation to be S – used for long queries.

1. $\pi_{name}\left(\sigma_{educatedAt=\text{"Hebrew University of Jerusalem"} \wedge \text{birthYear}>1970}\text{members}\right)$
2. $\pi_{name,party}\left(\sigma_{number=1}(\text{members}\bowtie\text{memberInKnesset})\right)$
3. $\pi_{name,number}\left(\sigma_{startYear-birthYear>70}\left(\text{members}\bowtie\sigma_{party="Likud" \vee party="Meretz"}(\text{memberInKnesset})\bowtie\text{knessets}\right)\right)$
4. $S=\sigma_{gender=\text{"female"}\wedge occupation\neq"politician"}\text{members}$

   $\pi_{name}\left(\pi_{uid,name}(S\bowtie\sigma_{number=23}\text{memberInKnesset})\cap\pi_{uid,name}(S\bowtie\sigma_{number=24}\text{memberInKnesset})\right)$
5. $S=(\rho_{R(number0,uid0,party0)}\text{memberInKnesset})\bowtie_{uid=uid0\wedge number\neq number0}\text{memberInKnesset}$

   S contains all tuples of same member in two <u>different</u> Knesset, meaning it does not include uid if the member is only in one Knesset.

   $T=(\sigma_{birthPlace=\text{"Jerusalem"}}\text{members})\bowtie\text{memberInKnesset}$

   T contains all members who were born in Jerusalem and were in at least one Knesset, therefore:

   $\pi_{name}\left(\pi_{uid,name}(T)-\pi_{uid,name}(S)\right)$

   (also uid in case 2 different members have the same uid)
6. $S=\pi_{number}\left(\text{memberInKnesset}\bowtie_{\text{memberInKnesset.uid}=\text{members.uid}\wedge name=\text{"David Ben-Gurion"}\wedge party=\text{"Mapai"}}\text{members}\right)$

   S are the numbers of knessets in which David kihen in Mapai party.

   $T=\pi_{number,name}\left(\text{memberInKnesset}\bowtie_{\text{memberInKnesset.uid}=\text{members.uid}\wedge party=\text{"Mapai"}}\text{members}\right)$

   T contains all members' names when kihanu in mapai and the number of Knesset, therefore:

   $T\div S$

   for each name of member that kihen in mapai, we get only those who kihanu in each number of S == knessets that DBG kihen.
7. $S=\text{memberInKnesset}\bowtie\text{members}$

   $T=(\rho_{R(number0,uid0,party0,uid0..birthYear0)}S)\bowtie_{uid\neq uid0\wedge number=number0\wedge birthYear0<birthYear}S$

   In T table we won't get any (uid, number) in the right side when "uid" is the older in Knesset "number", while as for the rest, there will always be someone older than them. Therefore, we can take all the tuples appear in S and remove the right part of T:

   $\pi_{namber,name}\left(\pi_{number,name,uid}(S)-\pi_{number,name,uid}(T)\right)$