

## PART 2: RELATIONAL ALGEBRA

1)

$$\Pi_{id, name, surname, tel} \text{ Member}$$

By the use of projection we display the required columns

2)

$$\Pi_{name, creationdate} ( \sigma_{type=Unlisted, createdby=1333} \text{ Group} )$$

Firstly we perform the selection operation to get the full table of the group with only the required type and information over who created it. Lastly we use the projection to display the required columns

3)

$$T1 = \sigma_{type=Standart, date \leq 1st \text{ January } 2022} \text{ Group}$$

$$T2 = \text{Member} \bowtie_{memberID = created \text{ by}} T1$$

$$T3 = \Pi_{name, surname, email} T2$$

Firstly we perform the selection operation to exclude the tuples where the type is not standard and the date is earlier than the 1st of January 2022. Secondly, we combine it with the member table where the member Id is equal to the id of the person who created the group. Thirdly we use projection to display the fields that were required

4)

$$T1 = \sigma_{member="Yeliz \text{ Yesilada}"} \text{ Connection List}$$

$$T2 = \text{Post} \bowtie_{postedby=conectionmemembrID} T1$$

$$T3 = \Pi_{content, title} T2$$

Firstly, we need to find all the member connections from the connection list for the given member name. Secondly we combine the output of the first operation to the Post table using the Equijoin where the posted by id is equivalent to the conectionId to find all the post that the connected members of the given user have made. Lastly, we use projection to display the relevant columns.

5)

$$T1 = \sigma_{\text{companyName}=\text{"Curiosity"}} \text{Company}$$
$$T2 = \text{Member} \bowtie_{\text{CompWorksfor}=\text{compID}} T1$$
$$T3 = T2 \bowtie_{\text{member}=\text{createdby}} \text{Group}$$
$$T4 = T3 \bowtie_{\text{manager}=\text{memberID}} \text{Member}$$
$$T5 = \Pi_{\text{name, surname, groupname, managername, managersurname}} T4$$

Firstly, we have to find all the information for the company in the company relation. Secondly we combine it with the member relation to find the members that work in that family by the use of equijoin based on the Company Id. Thirdly, we connect the member we found by the use of equijoin with the Group relation to see the groups member of that company have created. Next, we combine the result with the member table again to get the names and the surnames of the manager by the use of equijoin where managers id is equal to members id. Lastly we do the projection to display the relative columns.

6)

$$T1 = \sigma_{\text{name}=\text{"Introduction to Python"}} \text{Assessment}$$
$$T2 = \text{assessmentTaken} \bowtie_{\text{assessmentID}=\text{AssessmentID}} T1$$
$$T3 = \text{Member} \bowtie_{\text{memberID}=\text{memberID}} T2$$
$$T4 = \Pi_{\text{name, surname}} T3$$
$$T5 = \text{name, surname} \Join_{\text{count}} T4$$
$$T6 = \rho_{\text{name, surname, times-taken}} T5$$

Firstly, we find the information over the assessment in the table by the use of selection. Next, we have to combine it by the use of equijoin to the assessment taken to see the member ids that have taken the assessment. Then, we combine the result with the member table to get the full information of the members that have taken the assessment. Next, we do the projection to reduce the amount of data we are currently storing. Next, we have to count the number of the attempts of each member, that is why we use aggregation with grouping by the name and surname. Lastly we name the counted column for the ease of reading.