Databases Foundations I
Season 2024-I
Workshop No. 2 — Relational Algebra

Eng. Carlos Andrés Sierra, M.Sc.
Computer Engineering
Universidad Distrital Francisco José de Caldas

Congratulations, you survive at your first challange. Now, it is time to keep pushing you because you could give more. As you undertstand how to design a relational database in a first but professional version, you next challenge is to define ways to extract meaningful information using relational algebra.

In this second challenge, *Alphabet* wants to see some information about the use of **YouTube**. In this way, for each one of the next requirements you must write the relational algebra expresion and draw the result table to validate your solution.

Alphabet asked you to deliver a document with the queries (and the result tables) described as follows:

1. Based on the table **Video** shows as follows:

VideoID	Title	Views	Likes	Dislikes	ChannelID
1	"How to make a cake"	8765	132	12	1
2	"How to make a pizza"	2547	432	24	1
3	"How to make a sandwich"	4765	654	37	2
4	"How to make a salad"	3213	432	34	2
5	"How to make a soup"	5876	651	58	3
6	"How to make a juice"	3456	325	23	3
7	"How to make a smoothie"	1436	754	65	4
8	"How to make a coffee"	800	323	43	4
9	"How to make a hotdog"	9987	565	9	5
10	"How to make a beer"	5422	543	87	5

Carlos Andrés Sierra, Computer Engineer, M.Sc. on Computer Engineering, Titular Professor at Universidad Distrital Francisco José de Caldas.

Any comment or concern related to this document could be send to Carlos A. Sierra at e-mail: *cavirguezs@udistrital.edu.co*

(a) Show the *Title* of the videos with more than 2000 views and less than 5000 views.

(b) Show the *Title* and *ChannelID* of the videos with more than 2000 views and less than 6000 views and more than 400 likes and less than 50 dislikes.

(c) Show the *Title*, *Likes* and *Views* of the videos with more than 3000 views and less than 4000 views and more than 340 likes and less than 55 dislikes and the *ChannelID* is 2.

(d) Show the *Title*, *Likes* and *Views* of the videos with more than 4567 views and more than 1000 likes, and called **MostPopulars**.

(e) Using next table called **VideoQuality**, show the *Title* and *Views* of the videos with less or equal than 40 dislikes with all possible video qualities.

VideoQualityID	Quality
1	240p
2	360p
3	480p
4	720p
5	1080p

2. Based on the table **Channel** shows as follows:

ChannelID	Name	Subscribers	Country
1	“Cooking with love”	12345	“USA”
2	“Healthy food”	32456	“UK”
3	“Natural drinks”	23445	“USA”
4	“Coffee and more”	3464	“Germany”
5	“Fast food”	18764	“USA”
6	“Mexican food”	23456	“Mexico”
7	“Italian food”	34567	“Italy”
8	“Molecular food”	45678	“USA”
9	“Vegan food”	56789	“India”
10	“Vegetarian food”	67890	“USA”

(a) Show the *Name* of the channels with more than 15000 subscribers.

(b) Show the *Name* and *Country* of the channels with more than 20000 subscribers and less than 40000 subscribers.

(c) Show the *Name*, *Country* and *Subscribers* of the channels with more than 10000 subscribers and less than 30000 subscribers and the *Country* is **USA**.

(d) Show all rows of the table **Channel** where there is a *food* word in the *Name*, and called **PureFoodChannels**.

(e) Show the *Name* of the channels with more than 30000 subscribers or the *Country* is not **USA**.

3. Based on the table **Playlists** shows as follows:

PlaylistID	**Name**	**Videos**	**ChannelID**
1	"Cooking"	23	1
2	"Healthy"	45	2
3	"Drinks"	38	3
4	"Coffee"	84	4
5	"Fast food"	17	5
6	"Mexican"	32	4
7	"Italian"	45	3
8	"Molecular"	23	4
9	"Vegan"	56	5
10	"Vegetarian"	67	5

   (a) Show the *Name* of the playlists with more than 35 videos.

   (b) Show the *Name*, *ChannelID* and *Videos* of the playlists with more than 20 videos and less than 40 videos and the *ChannelID* is 3.

   (c) Show the *Name*, *Videos* and *ChannelID* of the playlists with more than 35 videos and called **MostVideos**.

   (d) Show the *Name* of the playlists with more than 40 videos and the *ChannelID* is 3 or the *ChannelID* is 4 or the *ChannelID* is 5 or the *ChannelID* is 1.

   (e) Show the *PlaylistID* and *Name* of the playlists.

4. Based on the tables showed above, create an *ER Diagram* to show the relationships between the tables. If you think you need an additional entity you could add it.