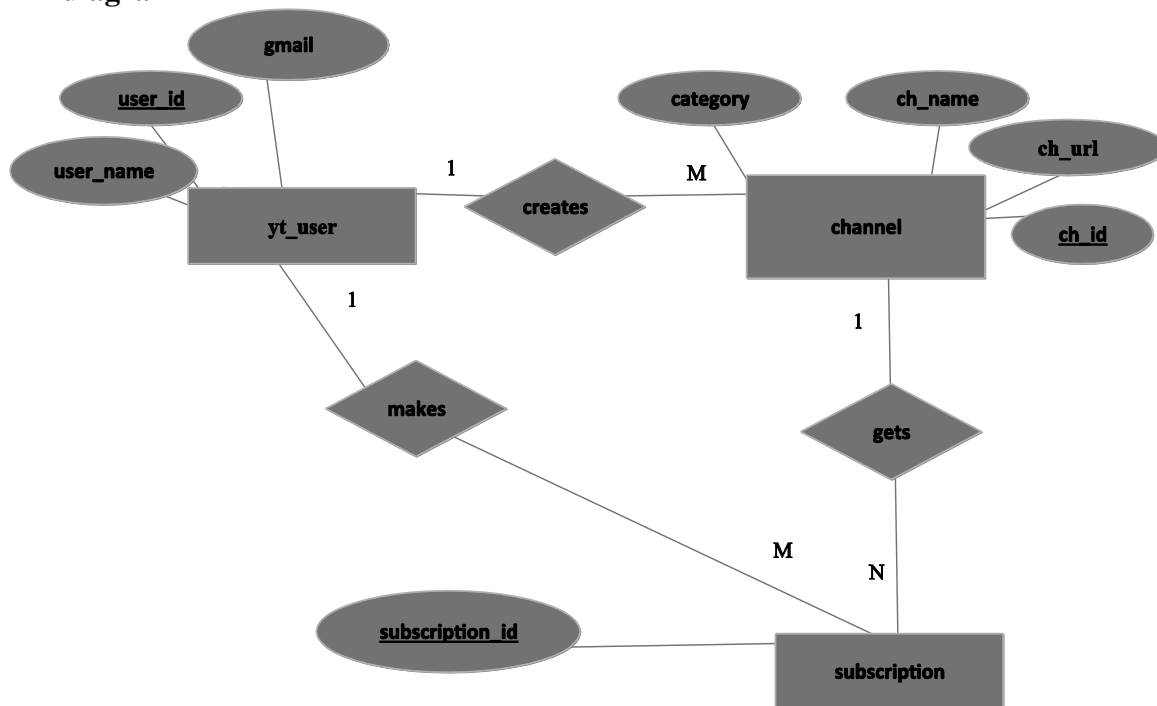


Lab 1- Assignment1 – SQL (7 marks) + (individual assessment 2 marks) = 9 marks

Here is the ER diagram of the Youtube database application explained in the first text book. Implement it by Sqldeveloper with the same data and try to write the queries to get the results shown for each query:

ER diagram



Creating Tables and populate data base

Creating yt_user (YouTube User Table), channel (Channel Table) and subscription (Subscription Table):

/* YouTube User Table*/

```
CREATE TABLE yt_user (
  user_id INTEGER PRIMARY KEY,
  user_name VARCHAR(100) NOT NULL,
  gmail VARCHAR(100) NOT NULL);
```

/*Channel Table */

```
CREATE TABLE channel (
  ch_id INTEGER PRIMARY KEY,
  ch_name VARCHAR(64) NOT NULL UNIQUE,
  ch_url VARCHAR(100),
  ch_category VARCHAR(20),
  user_id INTEGER REFERENCES yt_user(user_id));
```

/*Subscription Table*/

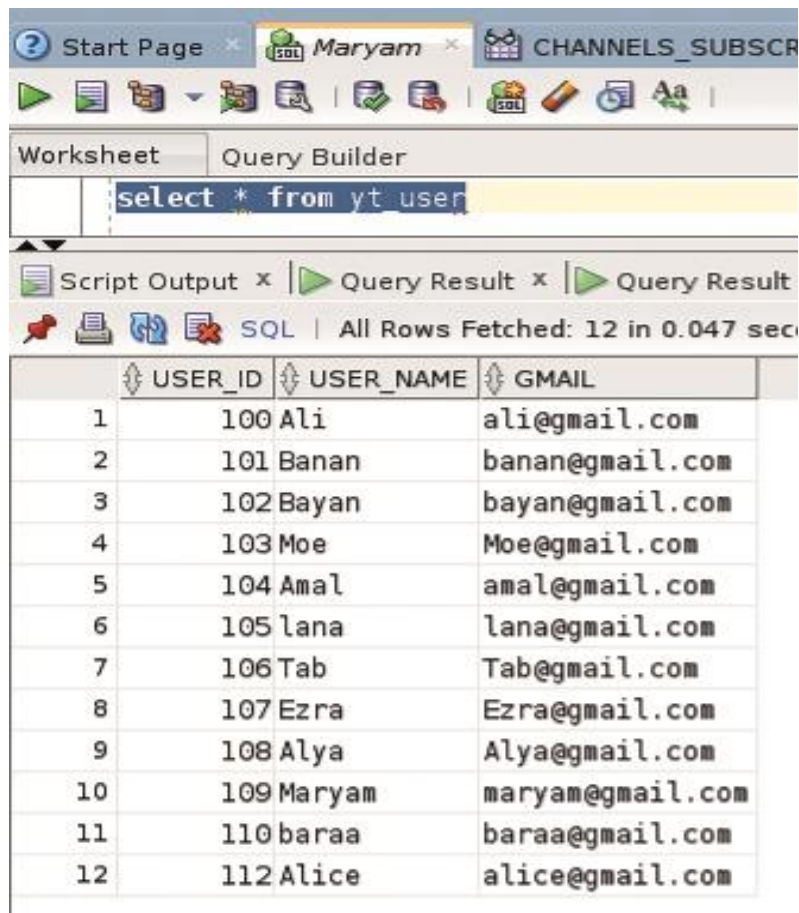
```
CREATE TABLE subscription (  
subscription_id INTEGER PRIMARY KEY,  
ch_id INTEGER REFERENCES channel(ch_id),  
user_id INTEGER REFERENCES yt_user(user_id));
```

Inserting

YouTube Users Table

We have inserted twelve records into Users Table.

```
INSERT INTO yt_user (user_id, user_name, gmail)VALUES (100, 'Ali', 'ali@gmail.com');  
INSERT INTO yt_user (user_id, user_name, gmail)VALUES (101, 'Banan',  
'banan@gmail.com');  
INSERT INTO yt_user (user_id, user_name, gmail)VALUES (102, 'Bayan',  
'bayan@gmail.com');  
INSERT INTO yt_user (user_id, user_name, gmail)VALUES (103, 'Moe', 'Moe@gmail.com');  
INSERT INTO yt_user (user_id, user_name, gmail)VALUES (104, 'Amal',  
'amal@gmail.com');  
INSERT INTO yt_user (user_id, user_name, gmail)VALUES (105, 'lana', 'lana@gmail.com');  
INSERT INTO yt_user (user_id, user_name, gmail)VALUES (106, 'Tab', 'Tab@gmail.com');  
INSERT INTO yt_user (user_id, user_name, gmail)VALUES (107, 'Ezra', 'Ezra@gmail.com');  
INSERT INTO yt_user (user_id, user_name, gmail)VALUES (108, 'Alya', 'Alya@gmail.com');  
INSERT INTO yt_user (user_id, user_name, gmail)VALUES (109, 'Maryam',  
'maryam@gmail.com');  
INSERT INTO yt_user (user_id, user_name, gmail)VALUES (110, 'baraa',  
'baraa@gmail.com');  
INSERT INTO yt_user (user_id, user_name, gmail) VALUES (112, 'Alice',  
'alice@gmail.com');
```



The screenshot shows a SQL query builder window with the following components:

- Query Builder:** The query entered is `select * from yt user`.
- Script Output:** Shows the query execution status: `SQL | All Rows Fetched: 12 in 0.047 sec`.
- Query Result:** A table displaying 12 rows of data.

	USER_ID	USER_NAME	GMAIL
1	100	Ali	ali@gmail.com
2	101	Banan	banan@gmail.com
3	102	Bayan	bayan@gmail.com
4	103	Moe	Moe@gmail.com
5	104	Amal	amal@gmail.com
6	105	lana	lana@gmail.com
7	106	Tab	Tab@gmail.com
8	107	Ezra	Ezra@gmail.com
9	108	Alya	Alya@gmail.com
10	109	Maryam	maryam@gmail.com
11	110	baraa	baraa@gmail.com
12	112	Alice	alice@gmail.com

Channel Table

Inserting fifteen records into Channel Table.

```

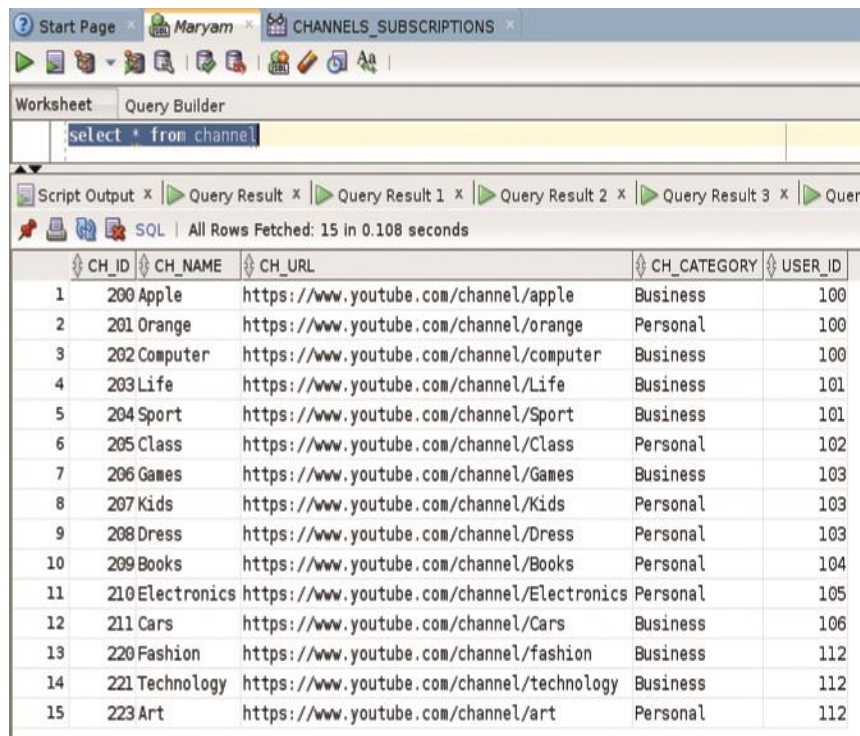
INSERT INTO channel (ch_id, ch_name, ch_url, ch_category, user_id) VALUES (200,
'Apple', 'https://www.youtube.com/channel/apple', 'Business', 100 );
INSERT INTO channel (ch_id, ch_name, ch_url, ch_category, user_id) VALUES (201,
'Orange', 'https://www.youtube.com/channel/orange', 'Personal', 100 );
INSERT INTO channel (ch_id, ch_name, ch_url, ch_category, user_id) VALUES (202,
'Computer', 'https://www.youtube.com/channel/computer', 'Business', 100 );
INSERT INTO channel (ch_id, ch_name, ch_url, ch_category, user_id) VALUES (203,
'Life', 'https://www.youtube.com/channel/Life', 'Business', 101 );
INSERT INTO channel (ch_id, ch_name, ch_url, ch_category, user_id) VALUES (204,
'Sport', 'https://www.youtube.com/channel/Sport', 'Business', 101 );
INSERT INTO channel (ch_id, ch_name, ch_url, ch_category, user_id) VALUES (205,
'Class', 'https://www.youtube.com/channel/Class', 'Personal', 102 );
INSERT INTO channel (ch_id, ch_name, ch_url, ch_category, user_id) VALUES (206,
'Games', 'https://www.youtube.com/channel/Games', 'Business', 103 );
INSERT INTO channel (ch_id, ch_name, ch_url, ch_category, user_id) VALUES (207,
'Kids', 'https://www.youtube.com/channel/Kids', 'Personal', 103 );
INSERT INTO channel (ch_id, ch_name, ch_url, ch_category, user_id) VALUES (208,
'Dress', 'https://www.youtube.com/channel/Dress', 'Personal', 103 );
INSERT INTO channel (ch_id, ch_name, ch_url, ch_category, user_id) VALUES (209,
'Books', 'https://www.youtube.com/channel/Books', 'Personal', 104 );

```

```

INSERT INTO channel (ch_id, ch_name, ch_url, ch_category, user_id) VALUES (210,
'Electronics', 'https://www.youtube.com/channel/Electronics', 'Personal', 105 );
INSERT INTO channel (ch_id, ch_name, ch_url, ch_category, user_id) VALUES (211,
'Cars', 'https://www.youtube.com/channel/Cars', 'Business', 106 );
INSERT INTO channel (ch_id, ch_name, ch_url, ch_category, user_id) VALUES (220,
'Fashion', 'https://www.youtube.com/channel/fashion', 'Business', 112);
INSERT INTO channel (ch_id, ch_name, ch_url, ch_category, user_id) VALUES (221,
'Technology', 'https://www.youtube.com/channel/technology', 'Business', 112);
INSERT INTO channel (ch_id, ch_name, ch_url, ch_category, user_id) VALUES (223,
'Art', 'https://www.youtube.com/channel/art', 'Personal', 112);

```



The screenshot shows a web-based database interface with a query result table. The query executed is 'select * from channel'. The result table has 5 columns: CH_ID, CH_NAME, CH_URL, CH_CATEGORY, and USER_ID. It displays 15 rows of data, including channels like Apple, Orange, Computer, Life, Sport, Class, Games, Kids, Dress, Books, Electronics, Cars, Fashion, Technology, and Art.

CH_ID	CH_NAME	CH_URL	CH_CATEGORY	USER_ID
1	200 Apple	https://www.youtube.com/channel/apple	Business	100
2	201 Orange	https://www.youtube.com/channel/orange	Personal	100
3	202 Computer	https://www.youtube.com/channel/computer	Business	100
4	203 Life	https://www.youtube.com/channel/Life	Business	101
5	204 Sport	https://www.youtube.com/channel/Sport	Business	101
6	205 Class	https://www.youtube.com/channel/Class	Personal	102
7	206 Games	https://www.youtube.com/channel/Games	Business	103
8	207 Kids	https://www.youtube.com/channel/Kids	Personal	103
9	208 Dress	https://www.youtube.com/channel/Dress	Personal	103
10	209 Books	https://www.youtube.com/channel/Books	Personal	104
11	210 Electronics	https://www.youtube.com/channel/Electronics	Personal	105
12	211 Cars	https://www.youtube.com/channel/Cars	Business	106
13	220 Fashion	https://www.youtube.com/channel/fashion	Business	112
14	221 Technology	https://www.youtube.com/channel/technology	Business	112
15	223 Art	https://www.youtube.com/channel/art	Personal	112

Subscription Table

We have inserted 31 records into Channel Table. The below statement is an example:

```

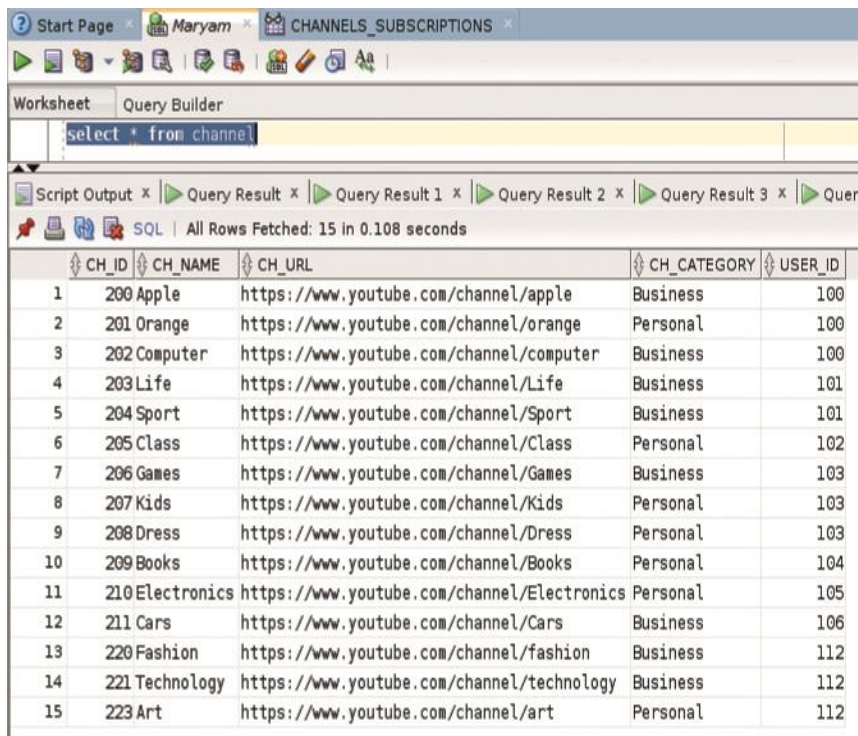
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (300,201,100);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (301,202,100);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (302,200,101);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (303,202,101);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (304,200,102);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (305,200,103);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (306,201,103);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (307,202,103);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (308,203,103);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (309,204,103);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (310,205,104);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (311,206,104);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (312,207,105);

```

```

INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (313,208,105);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (314,209,106);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (315,210,107);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (316,211,107);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (317,220,108);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (318,221,109);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (319,221,100);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (320,221,101);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (321,221,102);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (322,221,103);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (324,200,110);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (325,200,107);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (326,201,106);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (327,209,107);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (328,209,108);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (329,205,105);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (330,205,106);
INSERT INTO subscription (subscription_id, ch_id, user_id) VALUES (331,205,107);

```



The screenshot shows a database query tool interface. The 'Query Builder' tab is active, displaying the query 'select * from channel'. Below the query, the 'Query Result' tab shows the results of the query. The results are displayed in a table with 5 columns: CH_ID, CH_NAME, CH_URL, CH_CATEGORY, and USER_ID. The table contains 15 rows of data, numbered 1 through 15. The data is as follows:

CH_ID	CH_NAME	CH_URL	CH_CATEGORY	USER_ID
200	Apple	https://www.youtube.com/channel/apple	Business	100
201	Orange	https://www.youtube.com/channel/orange	Personal	100
202	Computer	https://www.youtube.com/channel/computer	Business	100
203	Life	https://www.youtube.com/channel/Life	Business	101
204	Sport	https://www.youtube.com/channel/Sport	Business	101
205	Class	https://www.youtube.com/channel/Class	Personal	102
206	Games	https://www.youtube.com/channel/Games	Business	103
207	Kids	https://www.youtube.com/channel/Kids	Personal	103
208	Dress	https://www.youtube.com/channel/Dress	Personal	103
209	Books	https://www.youtube.com/channel/Books	Personal	104
210	Electronics	https://www.youtube.com/channel/Electronics	Personal	105
211	Cars	https://www.youtube.com/channel/Cars	Business	106
220	Fashion	https://www.youtube.com/channel/fashion	Business	112
221	Technology	https://www.youtube.com/channel/technology	Business	112
223	Art	https://www.youtube.com/channel/art	Personal	112

Queries (in total 7 marks – each query 1 mark)

1- /*List of channels and their IDs for nonbusiness channels*/

The screenshot shows a database query tool interface. At the top, there are tabs for 'Start Page', 'Maryam', and 'CHANNELS_SUBSCRIPTIONS'. Below the tabs is a toolbar with various icons and a timer showing '0.73799998 seconds'. The main area is divided into 'Worksheet' and 'Query Builder' tabs. The 'Query Builder' tab is active, showing the query: '/* list of channels and their IDs for non Business channels*/'. Below the query, there is a yellow rectangular box. The bottom section of the interface shows 'Script Output', 'Query Result 5', and 'Query Result 6' tabs. The 'Query Result 6' tab is active, displaying a table with 7 rows and 2 columns: 'CH_NAME' and 'CH_ID'. The table contains the following data:

	CH_NAME	CH_ID
1	Orange	201
2	Class	205
3	Kids	207
4	Dress	208
5	Books	209
6	Electronics	210
7	Art	223

2-/*List of personal channels and their creators*/

Worksheet Query Builder

/*list of Personal channels and their creators*/

Script Output x Query Result 5 x Query Result 6 x Query Re

SQL | All Rows Fetched: 7 in 0.043 seconds

	USER_NAME	CH_NAME
1	Alice	Art
2	Amal	Books
3	Bayan	Class
4	Moe	Dress
5	lana	Electronics
6	Moe	Kids
7	Ali	Orange

3-/* Personal channels belonging to Alice*/

Worksheet Query Builder

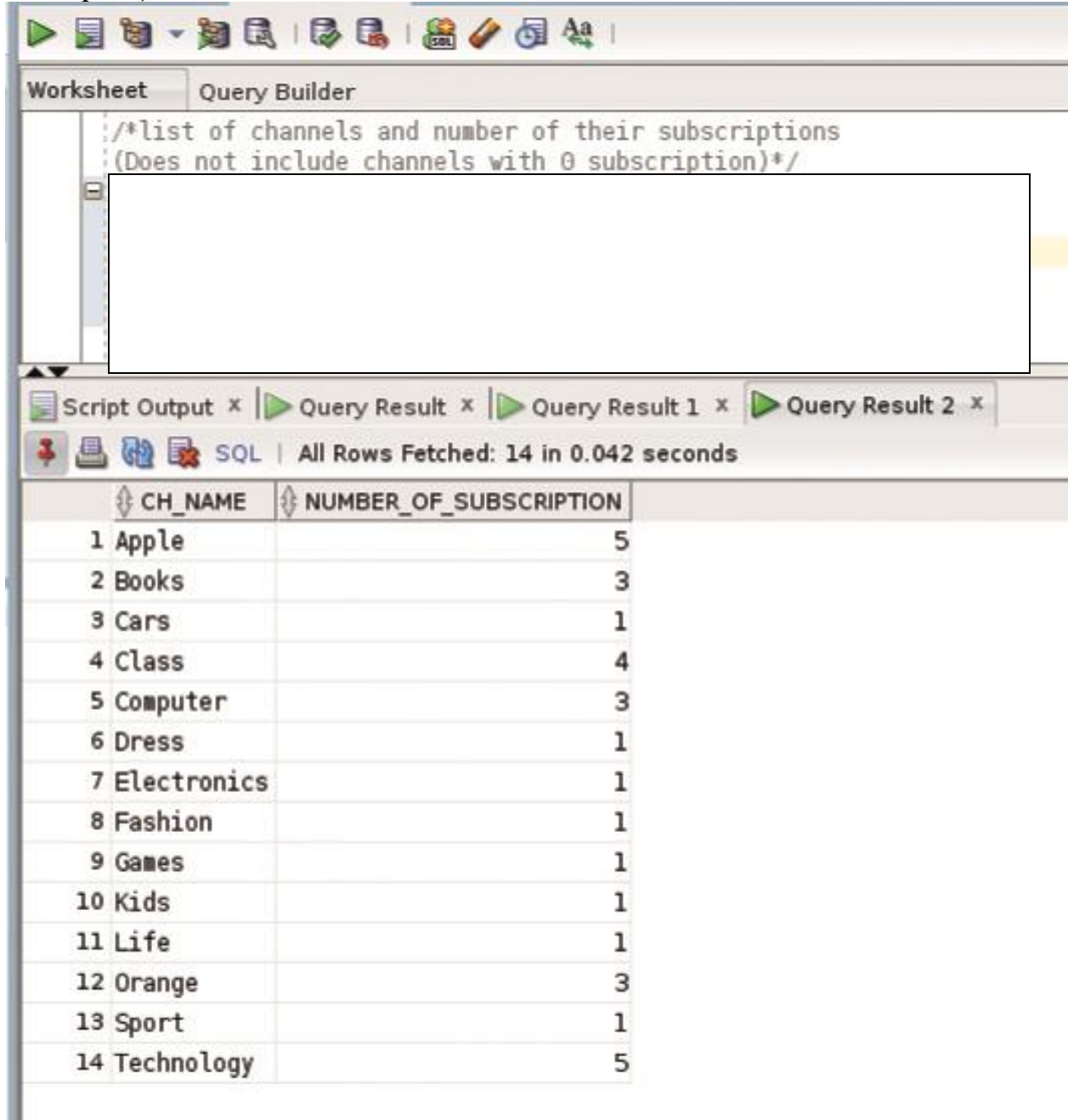
/* Personal channels belonging to Alice*/

Script Output x Query Result x

SQL | All Rows Fetched: 1 in 0.04 seconds

	USER_ID	USER_NAME	CH_NAME
1	112	Alice	Art

4-/*List of channels and number of their subscriptions (does not include channels with zero subscription)*/



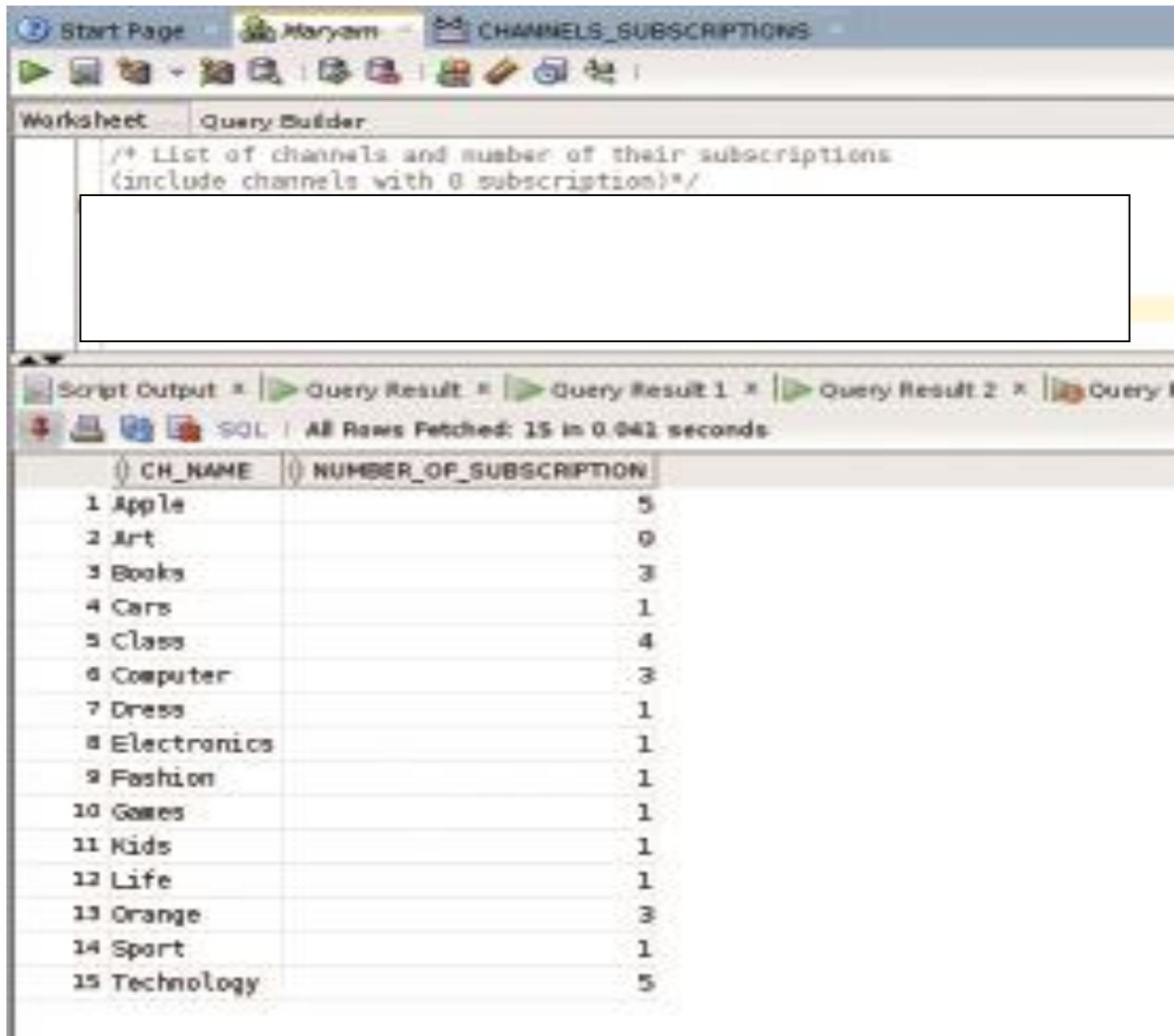
The screenshot shows a database query builder interface. The top toolbar contains various icons for file operations, database connections, and query execution. Below the toolbar, there are tabs for 'Worksheet' and 'Query Builder'. The 'Query Builder' tab is active, displaying a SQL query in a text area:

```
/*list of channels and number of their subscriptions  
(Does not include channels with 0 subscription)*/
```

Below the query text area, there is a large empty rectangular box. At the bottom of the interface, there is a 'Script Output' tab and a 'Query Result' tab. The 'Query Result' tab is active, showing a table with 14 rows and 2 columns. The table headers are 'CH_NAME' and 'NUMBER_OF_SUBSCRIPTION'. The data is as follows:

	CH_NAME	NUMBER_OF_SUBSCRIPTION
1	Apple	5
2	Books	3
3	Cars	1
4	Class	4
5	Computer	3
6	Dress	1
7	Electronics	1
8	Fashion	1
9	Games	1
10	Kids	1
11	Life	1
12	Orange	3
13	Sport	1
14	Technology	5

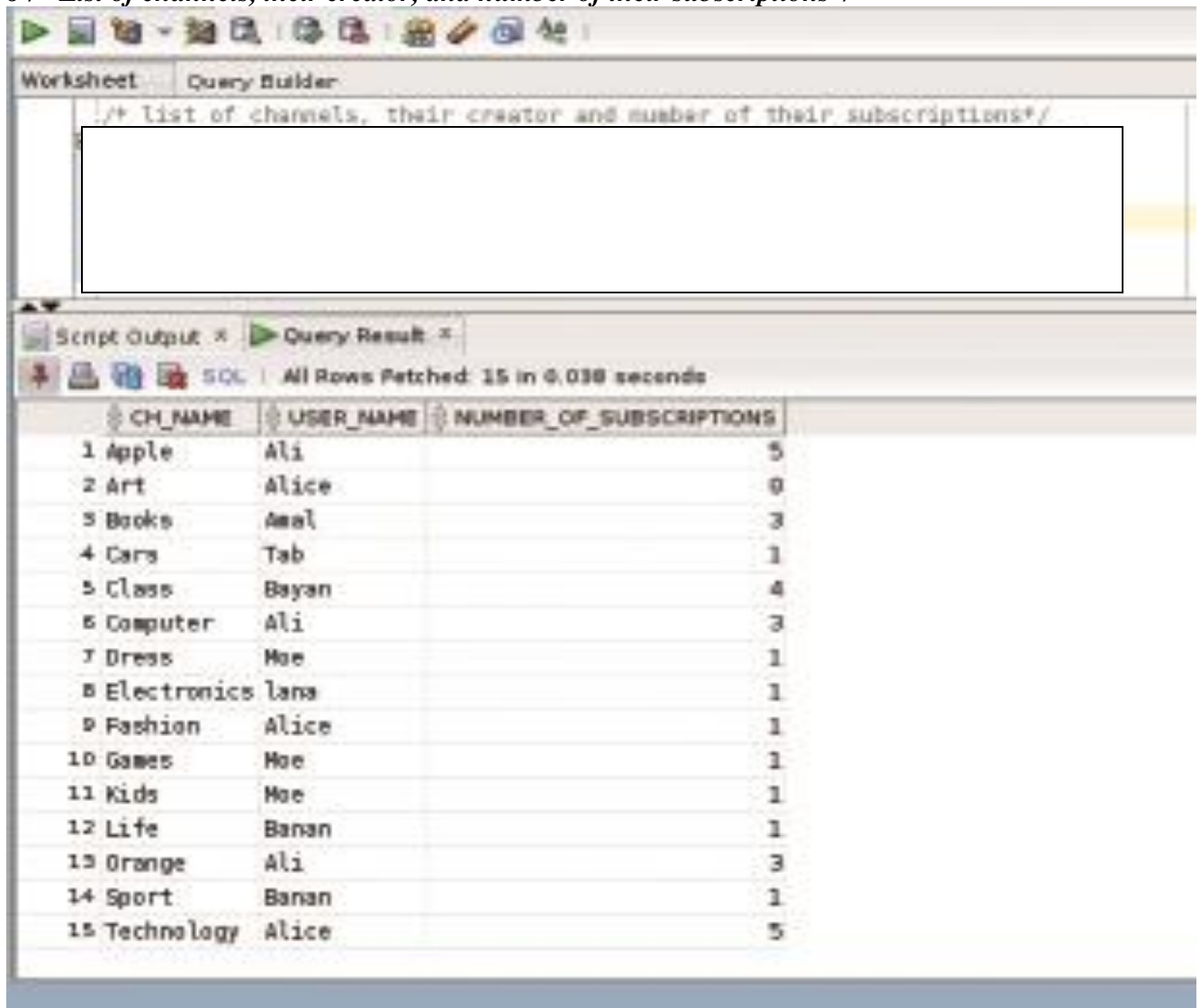
5-/* List of channels and number of their subscriptions (include channels with zero subscription)*/



The screenshot shows a database query tool interface. At the top, there's a toolbar with icons for Start Page, Maryam, and CHANNELS_SUBSCRIPTIONS. Below the toolbar, there's a tab labeled 'Query Builder'. The main area contains a query: `/* List of channels and number of their subscriptions (include channels with 0 subscription)*/`. Below the query, there's a large empty rectangular box. At the bottom, there's a 'Script Output' tab with a 'Query Result' tab selected. The query result shows a table with two columns: 'CH_NAME' and 'NUMBER_OF_SUBSCRIPTION'. The table contains 15 rows of data, numbered 1 to 15. The data is as follows:

	CH_NAME	NUMBER_OF_SUBSCRIPTION
1	Apple	5
2	Art	0
3	Books	3
4	Cars	1
5	Class	4
6	Computer	3
7	Dress	1
8	Electronics	1
9	Fashion	1
10	Games	1
11	Kids	1
12	Life	1
13	Orange	3
14	Sport	1
15	Technology	5

6-/* List of channels, their creator, and number of their subscriptions*/



The screenshot shows a database query tool interface. At the top, there is a toolbar with various icons. Below it, a tab labeled 'Query Builder' is active. The query text area contains the SQL statement: `/* List of channels, their creator and number of their subscriptions*/`. Below the query area, there is a 'Script Output' tab and a 'Query Result' tab. The 'Query Result' tab is active, showing a table with 15 rows and 3 columns: CH_NAME, USER_NAME, and NUMBER_OF_SUBSCRIPTIONS. The table data is as follows:

	CH_NAME	USER_NAME	NUMBER_OF_SUBSCRIPTIONS
1	Apple	Ali	5
2	Art	Alice	0
3	Books	Amal	3
4	Cars	Tab	1
5	Class	Bayan	4
6	Computer	Ali	3
7	Dress	Moe	1
8	Electronics	Iana	1
9	Fashion	Alice	1
10	Games	Moe	1
11	Kids	Moe	1
12	Life	Banan	1
13	Orange	Ali	3
14	Sport	Banan	1
15	Technology	Alice	5

7- */*The total number of subscriptions*/*

The screenshot shows a database query tool interface. At the top, there are two tabs: 'Worksheet' and 'Query Builder'. The 'Query Builder' tab is active, displaying the query: `/*The total number of Subscription*/`. Below the query editor, there is a large empty rectangular box. At the bottom of the interface, there is a status bar with several icons (a red pushpin, a printer, a blue folder, and a red X) followed by the text 'SQL | All Rows Fetched: 1 in 0.038 seconds'. Below the status bar, there is a table with one column header 'TOTAL_NUMBER_OF_SUBSCRIPTIONS' and one row of data with the value '31'.

TOTAL_NUMBER_OF_SUBSCRIPTIONS
31