

# Essentials Lab 9

## Exercise 1

- 1) Download and unzip the files for this lab.

[https://www.snowflakeuniversity.com/EL9/downloads/Essentials Lesson 9 Files.zip](https://www.snowflakeuniversity.com/EL9/downloads/Essentials_Lesson_9_Files.zip)

- 2) Load the script named **Lesson\_9\_File\_1.txt** into a new worksheet.

- 3) Run the statements one at a time or all together, based on your own preference.

**NOTE:** We've input the first two author records manually by assigning a 1 and 2 as the **AUTHOR\_UIDs**, but this is neither efficient nor practical, long term.

In the next step we create a “counter,” called a SEQUENCE, to supply the unique numbers to each new author record.

```
▶ Run All Queries | Saved a few seconds ago

1 // Create a new database and set the context to use the new database
2 CREATE DATABASE LIBRARY_CARD_CATALOG COMMENT = 'Essentials Lesson 9';
3 USE DATABASE LIBRARY_CARD_CATALOG;

4
5 // Create and Author table
6 CREATE OR REPLACE TABLE AUTHOR (
7     AUTHOR_UID NUMBER
8     , FIRST_NAME VARCHAR(50)
9     , MIDDLE_NAME VARCHAR(50)
10    , LAST_NAME VARCHAR(50)
11 );
12
13 // Insert the first two authors into the Author table
14 INSERT INTO AUTHOR(AUTHOR_UID,FIRST_NAME,MIDDLE_NAME, LAST_NAME)
15 Values
16 (1, 'Fiona', '', 'Macdonald')
17 ,(2, 'Gian','Paulo','Faleschini');
18
```



# Essentials Lab 9

## Exercise 2

- 1) Click **[Databases]** in the **Navigation Ribbon**.
- 2) Click **LIBRARY\_CARD\_CATALOG** to drill into that database.
- 3) Click the **[Sequences]** tab.
- 4) Click **[Create]**.

The screenshot shows the Snowflake web interface. At the top, there is a navigation bar with icons for Databases, Shares, Warehouses, Worksheets, and History. The 'Databases' icon is highlighted. Below the navigation bar, the path 'Databases > LIBRARY\_CARD\_CATALOG' is displayed. Underneath the path, there are tabs for Tables, Views, Schemas, Stages, File Formats, and Sequences. The 'Sequences' tab is currently selected and underlined. Below these tabs are several action buttons: '+ Create...', 'Clone...', 'Edit...', 'Drop...', and 'Transfer Ownership'. A main table is present with columns: Sequence, Schema, Creation Time ▾, Next Value, and Interval. The table is currently empty.

# Essentials Lab 9

## Exercise 3

- 1) Type in the name **SEQ\_AUTHOR\_UID**.
- 2) Leave the defaults in **[Schema Name]**, **[Initial Value]** and **[Interval]** fields.
- 3) Add a **[Comment]** to remind yourself of how you will use this Sequence. (OPTIONAL).
- 4) Click **[Finish]**.

The screenshot shows the Snowflake web interface. At the top, there's a navigation bar with icons for Databases, Shares, Warehouses, Worksheets, History, and Account. Below the navigation bar, the path 'Databases > LIBRARY\_CARD\_CATALOG' is visible. On the left, there's a sidebar with 'Tables' and a 'Create...' button. The main area is a modal dialog titled 'Create Sequence'. It contains the following fields:

- Name \*: SEQ\_AUTHOR\_UID
- Schema Name: PUBLIC
- Initial Value: 1
- Interval: 1
- Comment: Use this to fill in the AUTHOR\_UID everytime you add a row

At the bottom of the dialog, there are 'Show SQL', 'Cancel', and 'Finish' buttons. The 'Finish' button is highlighted with a blue background.

# Essentials Lab 9

## Exercise 4

- 1) Click **[Worksheets]** in the **Navigation Ribbon**.
- 2) Make sure the **Worksheet Context** is set to use the new **LIBRARY\_CARD\_CATALOG** database.
- 3) Type a **SELECT** statement like the one shown here, into a worksheet.
- 4) Click **[Run]**.
- 5) Note the **NEXTVAL** column in the **Results Pane**, then click **[Run]** again.

The screenshot shows a Snowflake Worksheet interface. At the top, there is a blue button labeled "Run". To its right are status indicators: "All Queries" (with a small icon), "Saved a minute ago", and "Context ▾". On the far right is a three-dot menu icon. Below the header, a query is listed: "1 SELECT SEQ\_AUTHOR\_UID.nextval;". A progress bar indicates the query is running. Underneath the query, there are tabs for "Results" (which is selected) and "Data Preview". The results pane shows a single row with the following data:

Query ID	SQL	373ms	1 rows
Filter result...			
Row			
1			

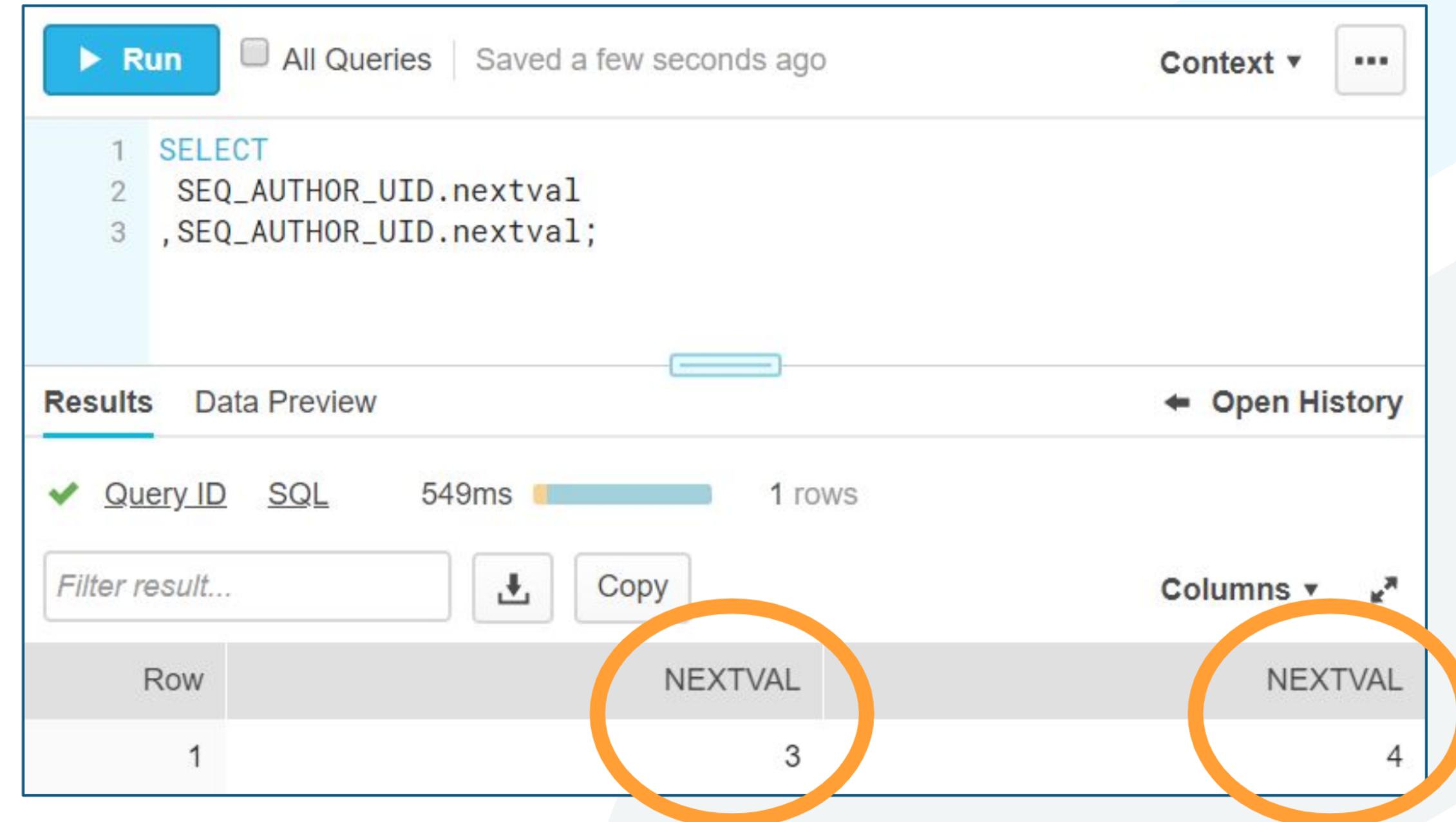
A "Columns ▾" button is located on the right side of the results pane. The "NEXTVAL" column is highlighted with an orange circle. The value in the first row of the "NEXTVAL" column is "2".

**NOTE:** Every time you refer to the **Sequence's Next Value** in a select statement, it will increment. You can run this statement as many times as you'd like. We will reset the counter before we use the value in a table.

# Essentials Lab 9

## Exercise 5

- 1) Change the select statement so that you refer to the **Sequence's Next Value** twice in the same select statement.
- 2) Click **[Run]**.
- 3) Click **[Run]** as many times as you'd like, in order to understand the effect of using the **Sequence's Next Value** in this way.



The screenshot shows a SQL query interface. At the top, there is a 'Run' button, a status bar indicating 'All Queries' and 'Saved a few seconds ago', and a 'Context' dropdown. Below the code editor, there are tabs for 'Results' and 'Data Preview', with 'Results' selected. The results table has columns for 'Query\_ID', 'SQL', 'Time', and 'Rows'. The first row shows a result of 3. The second row shows a result of 4. Both the '3' and '4' values are circled in orange.

```
1 SELECT
2   SEQ_AUTHOR_UID.nextval
3 , SEQ_AUTHOR_UID.nextval;
```

Results Data Preview

Query\_ID SQL 549ms 1 rows

Row	NEXTVAL
1	3
	4

**NOTE:** Two **Sequence Next Values** used in the same query may skip over some values and leave them unused. This is a safety feature to make sure no value is used more than once. When used one at a time, gaps almost never occur but are NOT guaranteed not to occur.



# Essentials Lab 9

## Exercise 6

- 1) Add the keyword **AS** to each column and assign names to the output values such as **Tally\_One** and **Tally\_Two** shown here.
- 2) Click **[Run]**.
- 3) Notice the assigned names are used as column headings in the **Results Pane**.
- 4) Click **[Run]** again. Repeat as many times as you'd like in order to understand the effect of using the **Sequence's Next Value** in this way.

The screenshot shows a query editor interface with the following details:

- Top Bar:** Contains a **Run** button, a **All Queries** dropdown, and a note that the query was **Saved a few seconds ago**. There is also a **Context** dropdown and a three-dot menu icon.
- Query Editor:** Displays the following SQL code:

```
1 SELECT
2 SEQ_AUTHOR_UID.nextval as Tally_One
3 , SEQ_AUTHOR_UID.nextval as Tally_Two;
```
- Results Pane:** Shows the results of the query. It has tabs for **Results** (which is selected) and **Data Preview**.
  - Filter result...** button
  - Download** and **Copy** buttons
  - Performance Metrics:** Query ID, SQL, 802ms execution time, and 1 row returned.
  - Column Headers:** Row, TALLY\_ONE, TALLY\_TWO
  - Data Rows:** Row 1 has values 6 and 7 respectively.
- Right Sidebar:** Includes a **Open History** link and a **Columns** dropdown.



# Essentials Lab 9

## Exercise 7

- 1) Load the file

**Lesson\_9\_File\_2.txt**

into a new worksheet.

- 2) Run each statement separately or as a group, according to your preference.

Run All Queries | Saved a few seconds ago

```
1 USE DATABASE LIBRARY_CARD_CATALOG;
2
3 //Drop and recreate the counter so that it starts at 3 (so we can add the
4 //other author records)
5 CREATE OR REPLACE SEQUENCE "LIBRARY_CARD_CATALOG"."PUBLIC"."SEQ_AUTHOR_UID"
6 START 3
7 INCREMENT 1
8 COMMENT = 'Use this to fill in the AUTHOR_UID everytime you add a row';
9
10 //Add the remaining author records and use the nextval function instead
11 //of putting in the numbers
12 INSERT INTO AUTHOR(AUTHOR_UID,FIRST_NAME,MIDDLE_NAME, LAST_NAME)
13 Values
14 (SEQ_AUTHOR_UID.nextval, 'Laura', 'K', 'Egendorf')
15 ,(SEQ_AUTHOR_UID.nextval, 'Jan', '', 'Grover')
16 ,(SEQ_AUTHOR_UID.nextval, 'Jennifer', '', 'Clapp')
17 ,(SEQ_AUTHOR_UID.nextval, 'Kathleen', '', 'Petelinsek');
```



# Essentials Lab 9

## Exercise 8

1) Preview the **AUTHOR** table.

2) Notice that the

`SEQ_AUTHOR_UID.nextval` function has resulted in a sequential series of values for the AUTHOR\_UID in each record.

Results **Data Preview** ← Open History

Table: LIBRARY\_CARD\_CATALOG.PUBLIC.AUTHOR Data Details

Filter result... Columns ▾ ↗

Preview Data X

Row	AUTHOR_UID	FIRST_NAME	MIDDLE_NAME	LAST_NAME
1	1	Fiona		Macdonald
2	2	Gian	Paulo	Faleschini
3	3	Laura	K	Egendorf
4	4	Jan		Grover
5	5	Jennifer		Clapp
6	6	Kathleen		Petelinsek

Data Type

NUMBER(38,0)  
VARCHAR(50)  
VARCHAR(50)  
VARCHAR(50)



# Essentials Lab 9

## Exercise 9

- 1) Load the file  
**Lesson\_9\_File\_3.txt**  
into a new worksheet.
- 2) Run each statement separately  
or as a group, according to your  
preference.

**NOTE:** By defining a **DEFAULT**  
value for the **BOOK\_UID** field, and  
setting it to your **Sequence's Next  
Value**, you make sure that each  
time a row is added to the table, it  
will get an automatically generated  
**Unique ID** assigned by the  
**Sequence**.

```
▶ Run All Queries | Saved a few seconds ago

1 // Create a new sequence, this one will be a counter for the book table
2 CREATE OR REPLACE SEQUENCE "LIBRARY_CARD_CATALOG"."PUBLIC"."SEQ_BOOK_UID"
3 START 1
4 INCREMENT 1
5 COMMENT = 'Use this to fill in the BOOK_UID everytime you add a row';
6
7 // Create the book table and use the NEXTVAL as the
8 // default value each time a row is added to the table
9 CREATE OR REPLACE TABLE BOOK
10 ( BOOK_UID NUMBER DEFAULT SEQ_BOOK_UID.nextval
11 ,TITLE VARCHAR(50)
12 ,YEAR_PUBLISHED NUMBER(4,0)
13 );
14
15 // Insert records into the book table
16 // You don't have to list anything for the
17 // BOOK_UID field because the default setting
18 // will take care of it for you
19 INSERT INTO BOOK(TITLE, YEAR_PUBLISHED)
20 VALUES
21 ('Food', 2001)
22 ,('Food', 2006)
23 ,('Food', 2008)
24 ,('Food', 2016)
25 ,('Food', 2015);
26
```

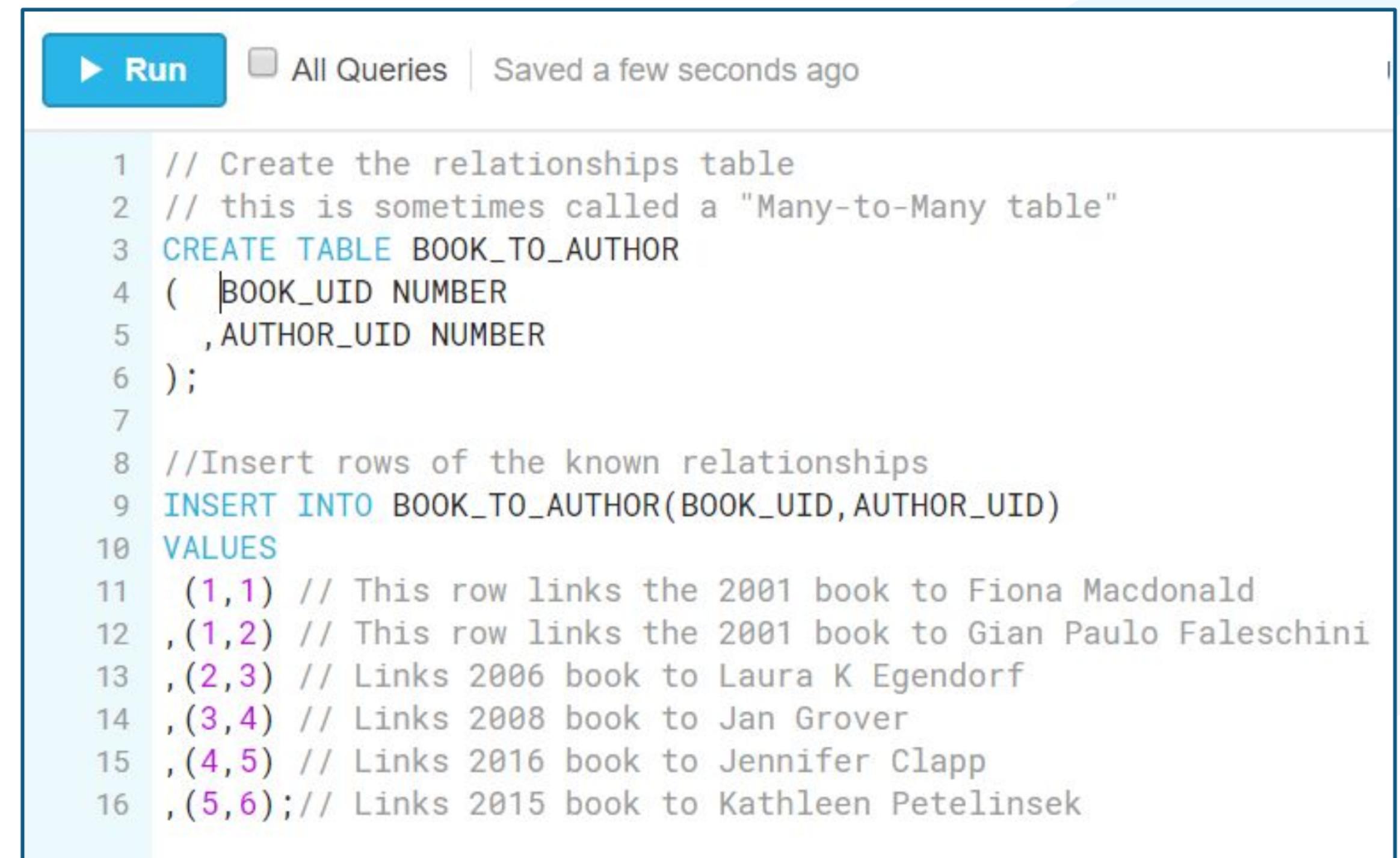


# Essentials Lab 9

## Exercise 10

- 1) Load the file **Lesson\_9\_File\_4.txt** into a new worksheet.
- 2) Run each statement separately or as a group, according to your preference.
- 3) Navigate to the **BOOK\_TO\_AUTHOR** table in the Navigation Tree of the Worksheets area and click **PREVIEW** to see the **BOOK\_TO\_AUTHOR** table.

**NOTE:** You have now created a small, normalized relational database. If you are new to the concept of 3NF database design, you may want to research the concept further.



The screenshot shows a Snowflake query editor interface. At the top, there is a blue button labeled "Run" with a play icon, followed by a checkbox labeled "All Queries" and the text "Saved a few seconds ago". The main area contains the following SQL code:

```
1 // Create the relationships table
2 // this is sometimes called a "Many-to-Many table"
3 CREATE TABLE BOOK_TO_AUTHOR
4 (
5     BOOK_UID NUMBER
6     ,AUTHOR_UID NUMBER
7 );
8 //Insert rows of the known relationships
9 INSERT INTO BOOK_TO_AUTHOR(BOOK_UID,AUTHOR_UID)
10 VALUES
11     ,(1,1) // This row links the 2001 book to Fiona Macdonald
12     ,(1,2) // This row links the 2001 book to Gian Paulo Faleschini
13     ,(2,3) // Links 2006 book to Laura K Egendorf
14     ,(3,4) // Links 2008 book to Jan Grover
15     ,(4,5) // Links 2016 book to Jennifer Clapp
16     ,(5,6); // Links 2015 book to Kathleen Petelinsek
```



