Demystifying SQL for Internal Auditors

Kate Head, CPA, CFE, CISA, CIG
University of South Florida

Joselyn De La Cruz-Rameau, Ed. D./ET
University of Texas-San Antonio

1

1

Introduction of Speakers

Kate Head, CPA, CFE, CISA, CIG	**Joselyn De La Cruz-Rameau, Ed. D./ET**
• Associate Director of Internal Audit, University of South Florida • BS in Accounting • Over 20 years experience with Data Analytics (ACL) and 30 years of Audit & Investigation experience	• Data Analyst for the University of Texas at San Antonio Audit/Cons. • BA in Business Administration, MS in Info. Systems Mgmt., Doctorate in Education with spec/Educational Technology. • Data Science for Business Certificate from Harvard Online. • Over 20 years in database programming, reporting, and analytics

2

2

Session Objectives

- Gain an understand of how to use SQL code.
- Learn the process for reviewing SQL code.
- Understand how to write and document SQL code.
- Explore best practice or use cases through interactions with others.

3

3

What is SQL?

SQL (pronounced “sequel”) stands for Structured Query Language. The language was originally developed by IBM in the 1970s.

The language is intended to mimic the human language and standardize the way a user will interact with a database.

SQL is used for communicating with databases and is the most commonly used language for manipulating data.

SQL can be used to insert, retrieve, modify, and delete data within a database and can be used to query data and create management reports.

4

4

SQL Categories

SQL is divided into the following categories:

Data Definition Language (DDL) – Defines the data objects (e.g., tables, columns, keys, indexes, etc.) within the database based on the defined data model.

Data Manipulation Language (DML) – Used to insert, update, or delete data within a database.

Data Query Language (DQL) – Used to query or select data from a database based on the defined data model. **This training will only focus on this category.**

5

5

Polling Question

What does the acronym SQL stand for in data analytics?

a. Structure Quasi Linguistics
b. Strategic Query Language
c. Structured Query Language
d. All of the above

6

6

Gain an understand on how to use SQL?

- Acquire a better understanding of existing queries
- Confirm the accuracy of query logic for key monitoring reports
- Test existing query output to validate accuracy and completeness
- Identify data sources for data analytic efforts
- Familiarize yourself with the structure and syntax of SQL

7

7

Learn the Process for reviewing SQL code

- Identify the table, report, or process to be reviewed
- Obtain the SQL Code and any supporting documentation:
  - User Service Request or Business Case for developing the SQL Code,
  - An external or internal data requirements,
  - Entity Relationship diagram,
  - Data Flow Diagram.
- Obtain a copy of the report or table produced by the SQL Code
- Dissects the SQL code
- Document your understand
- Confirm your results

8

8

Dissecting the Code

- Identify the source of the data (FROM)
- Determine the columns obtained from each data source (COLUMNS)
- Evaluate any filters applied to the data population (WHERE)
- Assess the degree of data aggregation(GROUP BY)
- Evaluate the relationships between the tables (JOIN)

9

9

Polling Question

When was SQL initially created?

a. 1970s
b. 1990s
c. 2000s
d. None of the above

10

10

Understand how to write and document SQL Code

- SELECT column_name(s)
- FROM table_name
- WHERE condition
- GROUP BY column_name(s)
- HAVING condition
- ORDER BY column_name(s);

According to CodeAcademy:
We are living in a data-driven world, many businesses store information inside massive databases. Structured Query Language (aka SQL) helps data scientists, engineers, and non-technical(auditors) employees manage and work with all kinds of database info in their daily work.

11

11

Understand how to write and document SQL Code

IMPORTANT: This example does not have PRIMARY and FOREIGN key constraints, These are addresses in further detail in the breakout session for SQL coders

StudentID	StudentName	PreferredName	Address	City	PostalCode	Country	Tuition
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany	1,000
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico	20,000
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico	50,000
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK	1,500
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden	300
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany	5,000

Source: Adapted from the Student table, Northwind University database available in W3Schools.

12

12

Understand how to write and document SQL Code (SELECT/FROM)

- SELECT StudentID, StudentName, PreferredName, Address, City, PostalCode, Country, Tuition
- FROM Student;

13

13

Understand how to write and document SQL Code (SELECT/FROM)

StudentID	StudentName	PreferredName	Address	City	PostalCode	Country	Tuition
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany	1,000
2	Ana Trujillo Emparedados y helados	Ana Trujillo	Avda. de la Constitución 2222	México D.F.	05021	Mexico	20,000
3	Antonio Moreno Taquería	Antonio Moreno	Mataderos 2312	México D.F.	05023	Mexico	50,000
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK	1,500
5	Berglunds snabbköp	Christina Berglund	Berguvsvägen 8	Luleå	S-958 22	Sweden	300
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany	5,000

Source: Adapted from the Student table, Northwind University database available in W3Schools.

14

14

Understand how to write and document SQL Code (WHERE)

- SELECT StudentID, StudentName, PreferredName, Address, City, PostalCode, Country, Tuition
- FROM Student
- WHERE StudentID = '1';

This would limit your data to the two rows of tuition records for the student Alfreds Futterkiste.

StudentID	StudentName	PreferredName	Address	City	PostalCode	Country	Tuition
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany	1,000
1	Alfreds Futterkiste	Maria Anders	Obere Str. 57	Berlin	12209	Germany	5,000

15

15

Understand how to write and document SQL Code (GROUP BY)

- SELECT StudentID, StudentName, SUM(Tuition)
- FROM Student
- WHERE StudentID = '1'
- GROUP BY StudentID, StudentName;

The above SQL statement results in the following:

StudentID	StudentName	SUM of Tuition
1	Alfreds Futterkiste	6,000

16

16

Understand how to write and document SQL Code (HAVING)

- SELECT COUNT(STUDENTID), StudentName
- FROM Student
- WHERE StudentID = '1'
- GROUP BY StudentName
- HAVING COUNT (StudentID)>1;

This results in the following output:

COUNT of StudentID	StudentName
2	Alfreds Futterkiste

17

17

Understand how to write and document SQL Code (ORDER BY)

- SELECT StudentID, StudentName, PreferredName, Address, City, PostalCode, Country, Tuition
- FROM Student
- ORDER BY StudentName DESC;

18

18

Understand how to write and document SQL Code (ORDER BY)

StudentID	StudentName	PreferredName	Address	City	PostalCode	Country	Tuition
5	Berglunds snabbkop	Christina Berglund	Berguvsvagen 8	Lulea	S-958 22	Sweden	300
4	Around the Horn	Thomas Hardy	120 Hanover Sq.	London	WA1 1DP	UK	1500
3	Antonio Moreno Taqueria	Antonio Moreno	Mataderos 2312	Mexico D.F.	5023	Mexico	50000
2	na Trujillo Emparedados y helado	Ana Trujillo	vda. de la Constitution 222	Mexico D.F.	5021	Mexico	20000
1	Alfreds Futterkste	Maria Anders	Obere Str. 57	Berlin	12209	Germany	1000
1	Alfreds Futterkste	Maria Anders	Obere Str. 57	Berlin	12209	Germany	5000

19

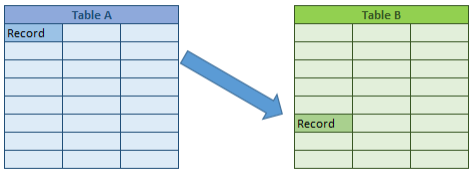
19

Understand how to write and document SQL Code (Table Relationships)

Table Relationships

A *one-to-one* (1:1)

Relationship means that each record in Table A relates to one, and only one, record in Table B, and each record in Table B relates to one, and only one, record in Table A.



20

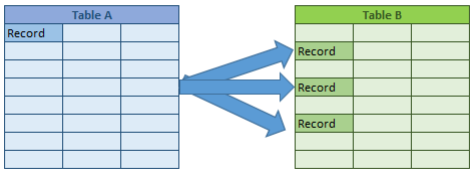
20

Understand how to write and document SQL Code (Table Relationships)

Table Relationships

A *one-to-many* (1:N)

Relationship means that each record instance in Table A, there exists zero, one, or many instances in Table B.



21

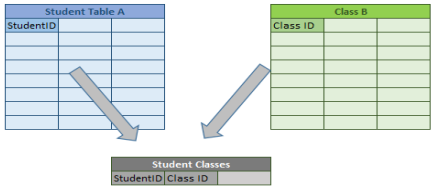
21

Understand how to write and document SQL Code (Table Relationships)

Table Relationships

Many-to-many (N:N) relationship.

Relationship in which multiple instances on Table A exist in with multiple instances in Table B. This constitutes a *many-to-many* (N:N) relationship.

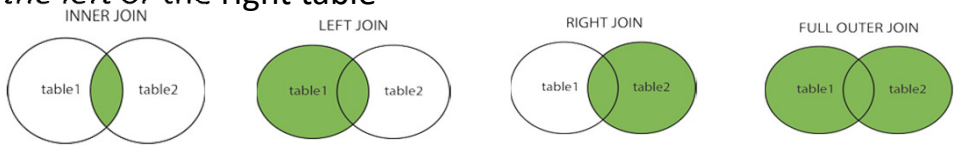


22

22

Understand how to write and document SQL Code

- **Inner Join:** returns records that are matching in both tables
- **Left (Outer) Join:** Returns all records from the left table and only the matching from the right table.
- **Right (Outer) Join:** Returns all records from the right table, and the matched records from the left table.
- **Full (Outer) Join:** Returns all records when there is a match in either the left or the right table



23

23

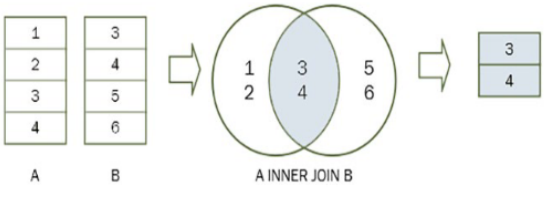
Understand how to write and document SQL

- **Inner Join:**

SQL INNER JOIN (sometimes called SIMPLE JOIN)

The Inner Join only returns data that appears in both data sets. With the above example of student and course tables, this join would return only students who have enrolled in courses, thus have a record in the Student table and one or more records in the Course table.

```
SELECT <select_list>
FROM Table_A A
INNER JOIN Table_B B
ON A.Key = B.Key
```



24

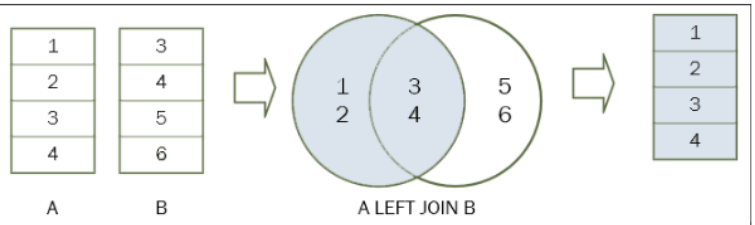
24

Understand how to write and document SQL Code

SQL LEFT OUTER JOIN (sometimes called LEFT JOIN)

The Left Outer Join returns all data in table A and only the matching records from table B. In our example, this Join would return all students in the student table and matching course information for the students in the student table. In other words, the results would list all students, but if a student was not assigned to any courses the course-related fields in the output would be blank.

```
SELECT
<select_list>
FROM Table_A A
LEFT JOIN
Table_B B
ON A.Key =
B.Key
```



25

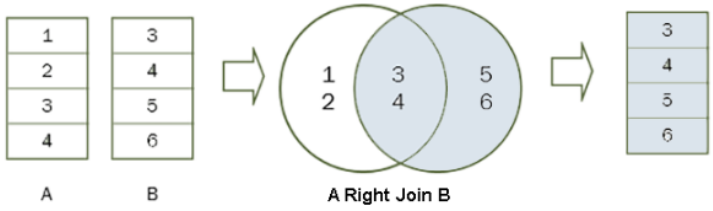
25

Understand how to write and document SQL Code

SQL RIGHT OUTER JOIN (sometimes called RIGHT JOIN)

The Right Outer Join returns all data in table B and the matching records from table A. This Join would return all courses from the course table and matching student information for the courses in the courses table. In other words, the results would list all courses, but if there were not students in a course the student-related fields in the output would be blank.

```
SELECT
<select_list>
FROM Table_A A
RIGHT JOIN
Table_B B
ON A.Key =
B.Key
```



26

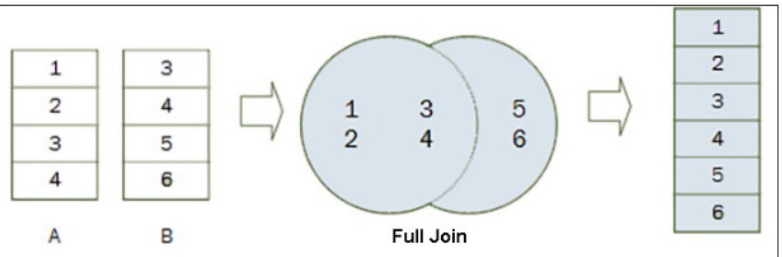
26

Understand how to write and document SQL Code

SQL FULL OUTER JOIN (sometimes called FULL JOIN)

The Full Outer Join returns all data in both table A and table B. This Join would return all students from the student table and courses from the course table. It is important to understand that relationships (one-to-one, one-to-many, many-to-many) differ from Joins.

```
SELECT
<select_list>
FROM Table_A A
FULL OUTER
JOIN Table_B B
ON A.Key =
B.Key
```



27

27

Polling Question

You have to an SQL programmer to be able to read the language?

a. True

b. False

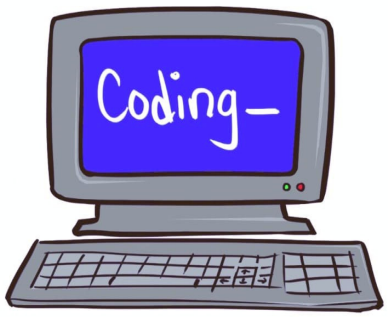
28

28

Break Out Sessions

Learning how to read SQL Code

Developing SQL Programming Skills



29

29

Polling Question

What is the Structured Query Language used for?

a. A Database Management System
b. It is a decentralized and unstructured language, that can only be used in one database management system
c. It is a standardized programming language that is used to manage relationship databases and perform various operations on the data in them
d. None of the above

30

30

Resources

Test your skills
Introduction to SQL PPT (Pt 1)
Introduction to SQL PPT (Pt 2)

31

31

Resources

1. Getting Started with SQL for Data Analysis Kick Starter(To be published by ACUA)
2. Aggregate Functions
https://docs.microsoft.com/en-us/sql/t-sql/functions/aggregate-functions-transact-sql?view=sql-server-ver15
3. SQL: Joins
https://www.techonthenet.com/sql/joins.php
4. SQL Tutorial
https://www.w3schools.com/sql/
5. Structured Query Language
https://www.computerworld.com/article/2595492/structured-query-language.html
6. Visual Representation of SQL Joins. The Code Project
https://www.codeproject.com/Articles/33052/Visual-Representation-of-SQL-Joins
7. Taylor, A. G. (2019). SQL for Dummies (9th Edition). ISBN 978-1-119-52707-7
8. Forta, B. (2019). Sams Teach Yourself SQL in 10 minutes (5th Edition); ISBN-13: 978-0-13-518279-6; ISBN-10: 0-13-518279-4
9. Stephens, R., Plew, R. and Jones, A.D. (2013). Sams Teach Yourself SQL in 24 hours (5th Edition); ISBN-13: 978-0-372-33541-9; ISBN-10: 0-672-33541-7

32

32

ACUA Data Analytics Team

Data Analytics Strategic Chair
Tiffany Yordan
tyordan@princeton.edu

Data Analytics Faculty Chair
Joselyn De La Cruz- Rameau
Joselyn.rameau@utsa.edu

Data Analytics Kick Starter Chair
Christine Heise
Christine.Heise@usnh.edu

33

33

Thank You

Kate Head, CPA, CFE, CISA, CIG	**Joselyn De La Cruz-Rameau, Ed. D./ET**
University of South Florida Office of Internal Audit (813) 974-3737 Khead@usf.edu	University of Texas-San Antonio Audit and Consulting (210) 458-4237 joselyn.rameau@utsa.edu

34

34