Basics of SQL and how to use it in Alma Analytics

Gem Stone-Logan
California Digital Library
gem.stone-logan@ucop.edu
Agenda

- General SQL Overview
- Explanation of basic SQL syntax
- Alma Analytics equivalents to SQL
- Where you can use SQL in Alma Analytics
- SQL example in column formula
- SQL example in filter
- Using SQL to join two subjects together
  - Possible bonus troubleshooting strategies
How to Pronounce SQL?

Two ways:

- Ess-cue-ell  /ˌɛsˌkju:ˈɛl/
- Sequel  /ˈsiːkwəl/

My opinion: it doesn't matter as long as people know what you're talking about.
What is SQL?

SQL is a query language that is mainly used when working with relational database management system (RDBMS) systems.

RDBMS Examples: MySQL, Microsoft SQL Server, Oracle Database, PostgreSQL, MySQL

While there is an SQL "standard", in practice every RDBMS uses a slightly different flavor.
Why care about SQL?

- Greater customization
- Better for automating
- General understanding of what's happening
Parts of an SQL Statement (when retrieving data)

SELECT

FROM

(optional) JOIN

(optional) WHERE

(optional) UNION

While every database has a slightly different way of doing things, the basics are often the same.
### Sample Bib Table

<table>
<thead>
<tr>
<th>title</th>
<th>mms_id</th>
<th>oclc</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Daughter of Doctor Moreau</td>
<td>9918605915306531</td>
<td>1281240579</td>
</tr>
<tr>
<td>The Kaiju Preservation Society</td>
<td>9917784458506531</td>
<td>1262693703</td>
</tr>
<tr>
<td>Nona the Ninth</td>
<td>9918538884306531</td>
<td>1304833436</td>
</tr>
<tr>
<td>Nettle &amp; Bone</td>
<td>9919069261806531</td>
<td>1268120733</td>
</tr>
<tr>
<td>Babel: or the necessity of violence: an arcane history of the Oxford translators' revolution</td>
<td>9919288262606531</td>
<td>1341991711</td>
</tr>
</tbody>
</table>

Databases are made up of tables. From Wikipedia: "A table is a collection of related data held in a table format within a database. It consists of columns and rows." This is a sample bib table which has title, mms_id, and oclc columns/fields and five rows of data.
Example SQL Statement (Pseudocode)

```
SELECT title
FROM bib
```

<table>
<thead>
<tr>
<th>title</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Daughter of Doctor Moreau</td>
</tr>
<tr>
<td>The Kaiju Preservation Society</td>
</tr>
<tr>
<td>Nona the Ninth</td>
</tr>
<tr>
<td>Nettle &amp; Bone</td>
</tr>
<tr>
<td>Babel: or the necessity of violence: an arcane history of the Oxford translators' revolution</td>
</tr>
</tbody>
</table>

This isn't real code, it's just to provide an idea of how SQL is usually structured. This would pull all titles contained within our fake bib table.
Here is a sample loan table which has loan_date, patron_id, mms_id and two rows of data.

<table>
<thead>
<tr>
<th>loan_date</th>
<th>patron_id</th>
<th>mms_id</th>
</tr>
</thead>
<tbody>
<tr>
<td>8/16/2023</td>
<td>1234</td>
<td>9917784458506531</td>
</tr>
<tr>
<td>7/30/2023</td>
<td>3456</td>
<td>9918538884306531</td>
</tr>
</tbody>
</table>
Example SQL Statement (Pseudocode)

```
SELECT bib.title,
    loan.loan_date
FROM bib
INNER JOIN loan
ON loan.mms_id = bib.mms_id
```

This isn't real code, it's just to provide an idea of how SQL is usually structured. Here I'm using an inner join which will return title and loan date for checked out items. We can do this because both tables contain an mms_id that we can use to join them together.

As an aside, this uses an explicit join. There's another way to do them that you may find in older textbooks and internet posts using just the FROM and WHERE clauses which is considered an implicit join. Newer database versions may not support the old method.
Here would be the results of our previous query.

<table>
<thead>
<tr>
<th>title</th>
<th>loan_date</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Kaiju Preservation Society</td>
<td>8/16/2023</td>
</tr>
<tr>
<td>Nona the Ninth</td>
<td>7/30/2023</td>
</tr>
</tbody>
</table>
Example SQL Statement (Pseudocode)

```
SELECT bib.title,  
loan.loan_date  
FROM bib  
LEFT JOIN loan  
ON loan.mms_id = bib.mms_id
```

This isn't real code, it's just to provide an idea of how SQL is usually structured. This is similar to the join we just looked at except it will return all titles, regardless of whether or not they're checked out, and if a loan date exists, it will also show that.
### Sample Results

<table>
<thead>
<tr>
<th>title</th>
<th>loan_date</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Daughter of Doctor Moreau</td>
<td></td>
</tr>
<tr>
<td>The Kaiju Preservation Society</td>
<td>8/16/2023</td>
</tr>
<tr>
<td>Nona the Ninth</td>
<td>7/30/2023</td>
</tr>
<tr>
<td>Nettle &amp; Bone</td>
<td></td>
</tr>
<tr>
<td>Babel: or the necessity of violence: an arcane history of the Oxford translators’ revolution</td>
<td></td>
</tr>
</tbody>
</table>

Here would be the results of our previous query.
Example SQL Statement (Pseudocode)

```sql
SELECT title, oclc
FROM bib
WHERE oclc = "1341991711"
```

<table>
<thead>
<tr>
<th>title</th>
<th>oclc</th>
</tr>
</thead>
<tbody>
<tr>
<td>Babel: or the necessity of violence: an arcane history of the Oxford translators' revolution</td>
<td>1341991711</td>
</tr>
</tbody>
</table>

This isn't real code, it's just to provide an idea of how SQL is usually structured. This would only pull titles and oclc numbers from the bib table if the title has this OCLC number.
**SQL: Does Case Matter?**

While keywords (such as SELECT, WHERE, COUNT, SUM) are often written in uppercase, they are not case sensitive.

However, Alma data **is** case sensitive so filtering for "Berkeley" will not retrieve results for "berkeley".
Alma Analytics and SQL

<table>
<thead>
<tr>
<th>Alma Analytics</th>
<th>SQL Equivalent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Columns to retrieve</td>
<td>SELECT</td>
</tr>
<tr>
<td>Subject</td>
<td>FROM</td>
</tr>
<tr>
<td></td>
<td>JOIN</td>
</tr>
<tr>
<td>Filters</td>
<td>WHERE</td>
</tr>
<tr>
<td>Combine subject results</td>
<td>UNION</td>
</tr>
</tbody>
</table>

Understanding the SQL equivalents can be useful when brainstorming how to get the results you want. For example, if you’re trying to change how your columns look, you’re probably looking for something that can be done in the SELECT part of an SQL statement. Whereas if you want to change what data you retrieve, you want to look for things that can be done in a WHERE clause. JOINs do not appear to have a clear equivalent in regular Alma Analytics.
Understanding the SQL equivalents can be useful when brainstorming how to get the results you want. For example, if you're trying to change how your columns look, you're probably looking for something that can be done in the SELECT part of an SQL statement. Whereas if you want to change what data you retrieve, you want to look for things that can be done in a WHERE clause.
Combine subjects in the SQL UNION equivalent. There doesn't seem to be an Analytics equivalent to JOIN.
SQL with Alma Analytics

Alma Analytics is built on Oracle Analytics Server (OAS) and is related to OBIEE (Oracle Business Intelligence Enterprise Edition)

Sometimes you can find ideas searching specifically for these two products. This page, says Alma Analytics is "built on Oracle Analytics Server". However, this page refers to Alma Analytics as Oracle Business Intelligence.
Where can you use SQL in Alma Analytics?

- Regular analysis/reports
  - Columns
  - Filters
  - Other random places
- Create Analysis from Simple Logical SQL
Using SQL in Column Formulas: Case(If)

Challenge:
Electronic Collections can have a "Public Name" and a "Public Name (override)". I want to retrieve one column that contains the version that will appear to the user.

Solution: Case (If)
Using SQL in Column Formulas: Case(If)

- In Criteria click the gear next to the column you wish to modify
- Click the insert Function icon and select a function from a menu to be given the basic SQL syntax
Using SQL in Column Formulas: Case(If)

How to figure out a new formula?

- The Insert Function window provides some basic information
- Google "Oracle Analytics Server" and your function
- Check your favorite Alma Analytics Resources

Some of my favorite resources are included at the end of this presentation.
Selected: Case (If)
Syntax: CASE WHEN request_condition1 THEN expr1 ELSE expr2 END
Where: exprs is any valid expression.

Example

Description: This form of the Case statement evaluates each WHEN condition and if satisfied, assigns the value in the corresponding THEN expression. If none of the WHEN conditions are satisfied, it assigns the default value specified in the ELSE expression. If no ELSE expression is specified, the system will automatically add an ELSE NULL.
<table>
<thead>
<tr>
<th>Expression</th>
<th>Example</th>
<th>Description</th>
<th>Syntax</th>
</tr>
</thead>
<tbody>
<tr>
<td>CASE (if)</td>
<td>CASE</td>
<td>Evaluates each WHEN condition and if satisfied, assigns the value in the corresponding THEN expression.</td>
<td>CASE WHEN request_condition1 THEN expr1 ELSE expr2 END</td>
</tr>
<tr>
<td></td>
<td>WHEN score-par &lt; 0 THEN 'Under Par'</td>
<td>If none of the WHEN conditions are satisfied, it assigns the default value specified in the ELSE expression. If no ELSE expression is specified, the system automatically adds an ELSE NULL.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>WHEN score-par = 0 THEN 'Par'</td>
<td><strong>Note</strong>: See Best Practices for using CASE statements in Analyses and Visualizations.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>WHEN score-par = 1 THEN 'Bogey'</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>WHEN score-par = 2 THEN 'Double Bogey'</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ELSE 'Triple Bogey or Worse'</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>END</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
### Basic formula

```
CASE WHEN [Insert Alma Column SQL] = ['Result you want replaced'] THEN ['New result'] ELSE [Repeat Insert Alma Column SQL] END
```

### Example with LC Classification Code

```
CASE
  WHEN "LC Classifications"."Classification Code" = 'Unknown'
  THEN 'Unclassed or mis-coded'
  ELSE "LC Classifications"."Classification Code"
END
```

https://wiki.harvard.edu/confluence/display/LibraryStaffDoc/Advanced+Formulas#AdvancedFormulas-ModifyingaPortionofaColumnwithaCaseStatement(columnformula)
Using SQL in Column Formulas: Case(If)

The official names of the columns:

- "Electronic Collection"."Electronic Collection Public Name"
- "Electronic Collection"."Electronic Collection Public Name (override)"

I find it easiest to add both columns I want data from initially to the report and then copy their complete columns names prior to working on the formula. We find the names by clicking on the gear icon and then choosing Edit Formula.
Using SQL in Column Formulas: Case(If)

If it exists, we want the "Public Name (override)" to appear. Otherwise, we want "Public Name"
Using SQL in Column Formulas: Case(If)

CASE WHEN "Electronic Collection"."Electronic Collection Public Name (override)" IS NOT NULL THEN "Electronic Collection"."Electronic Collection Public Name (override)"
ELSE "Electronic Collection"."Electronic Collection Public Name"
END

Note that while we won't get an error if we try to say "Electronic Collection"."Electronic Collection Public Name (override)"='', it won't give us the results we want. This is because a blank/null can't be equal to anything. Here's a good discussion of NULL (specifically for SQL Server for the same concepts apply):
https://simplesqltutorials.com/7-mistakes-with-null/
Using SQL in Filters: New Titles

Challenge:
Titles added in the last 90 days
<table>
<thead>
<tr>
<th>Bibliographic Details</th>
<th>Title</th>
<th>Creation Date</th>
<th>Title Measures</th>
<th>Num of Physical Items (Active)</th>
</tr>
</thead>
</table>

**Filters**

- Institution Code is equal to / is in 01UCS_UCM
- Suppressed From Discovery is equal to / is in No
- Num of Physical Items (Active) is greater than 0
- Creation Date is greater than or equal to 06/01/2023 12:00:00 AM

Use Date - New Titles
Using SQL in Filters: New Titles

Using SQL to create a relative date:

- Find a formula close to what we want

  ```
  TIMESTAMPADD(SQL_TSI_DAY, -7, CURRENT_DATE)
  ```

- Edit our existing creation date filter and convert it to SQL

https://knowledge.exlibrisgroup.com/Alma/Product_Documentation/010Alma_Online_Help_(English)/080Analytics/050Common__Analytics_Procedures#Relative_Dates
Using SQL in Filters: New Titles

TIMESTAMPADD: Add a specific amount of time to a date

SQL_TSI_DAY: Specifying that we want to add days (vs hours or some other time unit)

-90: The negative indicates we want to subtract

CURRENT_DATE: Today's date

https://docs.oracle.com/middleware/1221/bieee/BIVUG/GUID-1A697795-7D1E-4296-961A-1002FDBD4F47.htm#BILUG667
Using SQL in Filters: New Titles

Converting an existing filter to SQL

Edit a filter and click the **Convert this filter to SQL** option.
Using SQL in Filters: New Titles

Advanced SQL Filter

This page allows you to enter a custom where clause using SQL syntax.

"Bibliographic Details"."Creation Date" >= TIMESTAMPADD(SQL_TSI_DAY, -90, CURRENT_DATE)
Using SQL in Alma: Additional Functions

Sometimes you run across a useful sounding function which you can't find in the Analytics function list. There are two functions you may be able to use to mimic the behavior:

EVALUATE and EVALUATE_AGGR

https://docs.oracle.com/middleware/12211/biee/BIVUG/GUID-7035040C-BB40-4392-920A-9A435593F659.htm#BILUG683

Evalute and Evaluate_aggr pass through your command directly to the underlying database.
Using SQL in Alma: Additional Functions

**EVALUATE** and **EVALUATE_AGGR** are similar but **EVALUATE_AGGR** is used with aggregate functions.

The [Harvard Analytics Wiki](https://wiki.harvard.edu/confluence/display/LibraryStaffDoc/Advanced+Formulas#AdvancedFormulas-EVALUATEandEVALUATE_AGGRDatabaseFunctions) has some specific examples of how these functions can be used.
SQL in Alma Analytics: Simple Logical SQL

Create Analysis from Simple Logical SQL is an option when you want to do something that Analytics doesn't easily allow. It is particularly useful when trying to join two subjects that aren't designed to join.
SQL in Alma Analytics: Simple Logical SQL

Downsides

- Complicated
- Hard to maintain
- No Ex Libris Support
- Sometimes things break for no reason

Mitigation

- Document *everything* you do
SQL in Alma Analytics: JOIN Scenario

Challenge:

● One campus wants to see the summary holdings both at their campus and our storage facilities
UCSD's holdings are easy to pull. It's adding a column for NRLF and SRLF where things get more complicated.

<table>
<thead>
<tr>
<th>Library Code</th>
<th>Summary Holding (UCSD)</th>
<th>Title (Complete)</th>
<th>Summary Holding (NRLF)</th>
<th>Summary Holding (SRLF)</th>
</tr>
</thead>
</table>
SQL in Alma Analytics: JOIN Scenario

One Solution:

- Create a report with the campus information and add the storage facilities as a LEFT JOIN
SQL in Alma Analytics: JOIN Scenario

To use a JOIN you need a column that is in both data sets.

- In this case, we can use Network Id.
- If this was all from one IZ, we could potentially use MMS Id.

At least, it doesn't seem I can prompt when I join two subjects. However, I could be wrong about this.
SQL in Alma Analytics: Types of Joins

LEFT JOIN

Includes all of UCSD's holdings but only includes NRLF if UCSD also has holdings.

<table>
<thead>
<tr>
<th>Library Code</th>
<th>Summary Holding (UCSD)</th>
<th>Title (Complete)</th>
<th>Summary Holding (NRLF)</th>
</tr>
</thead>
</table>

An INNER JOIN would include too little information because it would only retrieve results where both UCSD and NRLF had holdings.

https://dataschool.com/how-to-teach-people-sql/sql-join-types-explained-visually/
SQL in Alma Analytics: JOIN Scenario

1. Get the report as close as possible to what you want.
2. Click the Advanced tab and look at the SQL in the SQL Issued section.
Selected Columns

Location | Holding Details | Bibliographic Details
---------|-----------------|------------------------
  Library Code | Location Code | Permanent Call Number | Summary Holding | Title (Complete) | MMS Id | OCL

Filters

1. Location Code is equal to / is in 'WongAvery'
2. Location Code is equal to / is in 'jml'
3. Holding Lifecycle is equal to / is in 'Active'
4. Bibliographic Level is equal to / is in 's'
5. Institution Code is equal to / is in '01UCS_SD1'
SELECT
  0 s_0,
  "Physical Items"."Bibliographic Details"."MMS Id" s_1,
  "Physical Items"."Bibliographic Details"."Network Id" s_2,
  "Physical Items"."Bibliographic Details"."OCLC Control Number (035a)" s_3,
  "Physical Items"."Bibliographic Details"."Title (Complete)" s_4,
  "Physical Items"."Holding Details"."Permanent Call Number" s_5,
  "Physical Items"."Holding Details"."Summary Holding" s_6,
  "Physical Items"."Location"."Library Code" s_7,
  "Physical Items"."Location"."Location Code" s_8
FROM "Physical Items"
WHERE
  "Physical Items"."Location"."Location Code" = '88888888'
SQL in Alma Analytics: Join Scenario

3. Clean up the SQL (personal preference)
   a. In the SELECT portion, replace the meaningless column names with actual names
   b. Remove the column I don't want
   c. Remove the remaining saw aliases (like saw_0)
   d. Delete the extra stuff at the end
   e. If needed, rearrange the columns
SELECT
    "Physical Items"."Location"."Library Code",
    "Physical Items"."Location"."Location Code",
    "Physical Items"."Holding Details"."Permanent Call Number",
    "Physical Items"."Holding Details"."Summary Holding",
    "Physical Items"."Bibliographic Details"."Title (Complete)",
    "Physical Items"."Bibliographic Details"."MMS Id",
    "Physical Items"."Bibliographic Details"."OCLC Control Number (035a)",
    "Physical Items"."Bibliographic Details"."Network Id"
FROM "Physical Items"
WHERE
    (("Location"."Library Code" = 'WongAvery')
    AND ("Location"."Location Code" = 'jrnl')
    AND ("Holding Details"."Holding Lifecycle" = 'Active')
    AND ("Bibliographic Details"."Bibliographic Level" = 's')
    AND (DESCRIPTOR_IDOF("Physical Items"."Institution"."Institution Code") = 6535))
SQL in Alma Analytics: JOIN Scenario

4. Test the new SQL
   a. In Analytics, click Create
   b. Click Analysis
   c. Click Create Analysis from Simple Logical SQL
   d. Copy in SQL to test
   e. Click OK
Select Subject Area

Search
- API Usage
- Analytics Objects
- Analytics Usage Tracking
- Benchmark
- Borrowing Requests (Resource Sharing)
- Course Reserves
- DARA
- Digital Inventory
- Digital Usage
- Create Analysis from Simple Logical SQL

Create analysis by entering simple logical SQL to create Analytics Server.
Enter a simple SQL statement to create an Analysis:

```
(()
  ("Location","Library Code" = 'WongAvery')
AND ("Location","Location Code" = 'jnl')
AND ("Holding Details","Holding Lifecycle" = 'Active')
AND ("Bibliographic Details","Bibliographic Level" = 's')
AND (DESCRIPTION_IDOF("Physical Items","Institution","Institution Code") = 6535))
```
SQL in Alma Analytics: JOIN Scenario

5. Reformat the existing SQL to be a subquery and test again

This is necessary because Alma Analytics will not let you join existing subjects together. You have to convince Analytics you have an entirely new subject.
SELECT ucsd."Library Code",
   ucsd."Location Code",
   ucsd."Permanent Call Number",
   ucsd."Summary Holding",
   ucsd."Title (Complete)",
   ucsd."MMS Id",
   ucsd."OCLC Control Number (035a)",
   ucsd."Network Id"
FROM
   (SELECT
      "Physical Items"."Location"."Library Code",
      "Physical Items"."Location"."Location Code",
      "Physical Items"."Holding Details"."Permanent Call Number",
      "Physical Items"."Holding Details"."Summary Holding",
      "Physical Items"."Bibliographic Details"."Title (Complete)",
      "Physical Items"."Bibliographic Details"."MMS Id",
      "Physical Items"."Bibliographic Details"."OCLC Control Number (035a)",
      "Physical Items"."Bibliographic Details"."Network Id"
   FROM "Physical Items"
   WHERE
      (("Location"."Library Code" = 'WongAvery')
      AND ("Location"."Location Code" = 'jrnl')
      AND ("Holding Details"."Holding Lifecycle" = 'Active')
      AND ("Bibliographic Details"."Bibliographic Level" = 's')
      AND (DESCRIPTOR_IDOF("Physical Items"."Institution"."Institution Code") = 6535))
   ) ucsd

Original report created as a subquery
SQL in Alma Analytics: JOIN Scenario

6. Now we can create our NRLF query
   a. If we want, we can create this first in Analytics, copy the SQL, and then clean it up.

This is necessary because Alma Analytics will not let you join existing subjects together. You have to convince Analytics you have an entirely new subject.
We need the Network Id because that's how we're going to join this with UCSD's data.
SELECT 0 s_0,
    "Physical Items"."Bibliographic Details"."Network Id" s_1,
    "Physical Items"."Holding Details"."Summary Holding" s_2
FROM "Physical Items"
WHERE (("Location"."Library Code" = 'NRLF') AND ("Holding Details"."Holding Lifecycle" = 'Active') AND ("Bibliographic Details"."Bibliographic Level" = 's') AND (DESCRIPTION_IDOF("Physical Items"."Institution"."Institution Code") = 6532))
ORDER BY 3 ASC NULLS FIRST, 2 ASC NULLS FIRST
FETCH FIRST 10000001 ROWS ONLY

Sample SQL for pulling data from nrlf.
SELECT
  "Physical Items"."Bibliographic Details"."Network Id",
  "Physical Items"."Holding Details"."Summary Holding"
FROM "Physical Items"
WHERE
  ("Location"."Library Code" = 'NRLF')
AND ("Holding Details"."Holding Lifecycle" = 'Active')
AND ("Bibliographic Details"."Bibliographic Level" = 's')
AND (DESCRIPTOR_IDOF("Physical Items"."Institution"."Institution Code") = 6532)
SQL in Alma Analytics: JOIN Scenario

7. Add the NRLF query to the UCSD query
   a. Make sure to give the NRLF subquery an alias
   b. Explain what the common column is between the two tables
   c. Add the NRLF Summary holdings in the top SELECT clause

This is necessary because Alma Analytics will not let you join existing subjects together. You have to convince Analytics you have an entirely new subject.
SELECT ucsd."Library Code",
    ucsd."Location Code",
    ucsd."Permanent Call Number",
    ucsd."Summary Holding",
    ucsd."Title (Complete)",
    ucsd."MMS Id",
    ucsd."OCLC Control Number (035a)",
    ucsd."Network Id",
    nrlf."Summary Holding"
FROM
    (SELECT
        "Physical Items"."Location"."Library Code",
        "Physical Items"."Location"."Location Code",
        "Physical Items"."Holding Details"."Permanent Call Number",
        "Physical Items"."Holding Details"."Summary Holding",
        "Physical Items"."Bibliographic Details"."Title (Complete)",
        "Physical Items"."Bibliographic Details"."MMS Id",
        "Physical Items"."Bibliographic Details"."OCLC Control Number (035a)",
        "Physical Items"."Network Id"
    FROM "Physical Items"
    WHERE
        ("Location"."Library Code" = 'WongAvery')
        AND ("Location"."Location Code" = 'jrnl')
        AND ("Holding Details"."Holding Lifecycle" = 'Active')
        AND ("Bibliographic Details"."Bibliographic Level" = 's')
        AND (DESCRIPTOR_IDOF("Physical Items"."Institution"."Institution Code") = 6535))
    ucsd
LEFT JOIN
    (SELECT
        "Physical Items"."Bibliographic Details"."Network Id",
        "Physical Items"."Holding Details"."Summary Holding"
    FROM "Physical Items"
    WHERE
        ("Location"."Library Code" = 'NRLF')
        AND ("Holding Details"."Holding Lifecycle" = 'Active')
        AND ("Bibliographic Details"."Bibliographic Level" = 's')
        AND (DESCRIPTOR_IDOF("Physical Items"."Institution"."Institution Code") = 6532))
    nrlf ON nrlf."Network Id" = ucsd."Network Id"

UCSD and NRLF combined with a Left Join (recommend grabbing the text from the slide and pasting into something like notepad to make it bigger font)
SQL in Alma Analytics: JOIN Scenario

8. The same process for SRLF
   a. Our base table is UCSD so join SRLF with UCSD, not with NRLF

This is necessary because Alma Analytics will not let you join existing subjects together. You have to convince Analytics you have an entirely new subject.
SELECT 0 s_0,  
  "Physical Items"."Bibliographic Details"."Network Id" s_1,  
  "Physical Items"."Holding Details"."Summary Holding" s_2  
FROM "Physical Items"  
WHERE  
  ("Location"."Library Code" = 'SRLF') AND ("Holding Details"."Holding Lifecycle" = 'Active') AND ("Bibliographic Details"."Bibliographic Level" = 's') AND (DESCRIPTOR_IDOF("Physical Items"."Institution"."Institution Code") = 6533))  
ORDER BY 3 ASC NULLS FIRST, 2 ASC NULLS FIRST  
FETCH FIRST 10000001 ROWS ONLY
SELECT "Physical Items"."Bibliographic Details"."Network Id",
       "Physical Items"."Holding Details"."Summary Holding"
FROM "Physical Items"
WHERE
  (("Location"."Library Code" = 'SRLF')
AND ("Holding Details"."Holding Lifecycle" = 'Active')
AND ("Bibliographic Details"."Bibliographic Level" = 's')
AND (_DESCRIPTOR_IDOF("Physical Items"."Institution"."Institution Code") = 6533))
NOTE: We will get an error if we try to run this.
UCSD, NRLF, and SRLF altogether (recommend grabbing the text from the slide and pasting into something like notepad to make it bigger font)
SQL in Alma Analytics: JOIN Scenario

What if we have a typo?

- When cutting and pasting it's really easy to mess up commas

This is necessary because Alma Analytics will not let you join existing subjects together. You have to convince Analytics you have an entirely new subject.

(SELECT "Physical Items"."Location"."Library Code", "Physical Items"."Location"."Location Code", "Physical Items"."Holding Details"."Permanent Call Number", "Physical Items"."Holding Details"."Summary Holding", "Physical Items"."Bibliographic Details"."Title (Complete)", "Physical Items"."Bibliographic Details"."MMS Id", "Physical Items"."Bibliographic Details"."OCLC Control Number (035a)", "Physical Items"."Bibliographic Details"."Network Id" FROM "Physical Items" WHERE ("Location"."Library Code" = 'WongAvery') AND ("Location"."Location Code" = 'jrnl') AND ("Holding Details"."Holding Lifecycle" = 'Active') AND ("Bibliographic Details"."Bibliographic Level" = 's') AND (DESCRIPTOR_IDOF("Physical Items"."Institution"."Institution Code") = 6535)) ucsd

LEFT JOIN (SELECT "Physical Items"."Bibliographic Details"."Network Id", "Physical Items"."Holding Details"."Summary Holding" FROM "Physical Items" WHERE ("Location"."Library Code" = 'NRLF') AND ("Holding Details"."Holding Lifecycle" = 'Active') AND ("Bibliographic Details"."Bibliographic Level" = 's') AND (DESCRIPTOR_IDOF("Physical Items"."Institution"."Institution Code") = 6532)) nrlf ON nrlf."Network Id" = ucsd."Network Id"

LEFT JOIN (SELECT "Physical Items"."Bibliographic Details"."Network Id", "Physical Items"."Holding Details"."Summary Holding" FROM "Physical Items" WHERE ("Location"."Library Code" = 'SRLF') AND ("Holding Details"."Holding Lifecycle" = 'Active') AND ("Bibliographic Details"."Bibliographic Level" = 's') AND (DESCRIPTOR_IDOF("Physical Items"."Institution"."Institution Code") = 6533)) srlf ON srlf."Network Id" = ucsd."Network Id"

UCSD, NRLF, and SRLF altogether with the typo fixed.

NOTE: as of 8/28/2023 this is not working for unknown reasons but it did work just a couple of weeks ago (i.e. this version does not have a typo in it).

(recommend grabbing the text from the slide and pasting into something like notepad to make it bigger font)
SQL in Alma Analytics: JOIN Scenario

Sometimes you just can't get what you want


Please have your service administrator review this error. (HY000)
SQL in Alma Analytics: JOIN Scenario

Each query is working individually so if we order by ucsd."MMS Id", would could combine the two reports in Excel.

This may not be the ideal solution but it's an option.
SELECT ucsd."Library Code", ucsd."Location Code", ucsd."Permanent Call Number", ucsd."Summary Holding", ucsd."Title (Complete)", ucsd."MMS ID", ucsd."OCLC Control Number (035a)", ucsd."Network Id", srlf."Summary Holding" FROM (SELECT "Physical Items"."Location"."Library Code", "Physical Items"."Location"."Location Code", "Physical Items"."Holding Details"."Permanent Call Number", "Physical Items"."Holding Details"."Summary Holding", "Physical Items"."Bibliographic Details"."Title (Complete)", "Physical Items"."Bibliographic Details"."MMS Id", "Physical Items"."Bibliographic Details"."OCLC Control Number (035a)", "Physical Items"."Bibliographic Details"."Network Id" FROM "Physical Items" WHERE (("Location"."Library Code" = 'WongAvery') AND ("Location"."Location Code" = 'jrnl') AND ("Holding Details"."Holding Lifecycle" = 'Active') AND ("Bibliographic Details"."Bibliographic Level" = 's') AND (DESCRIPTOR_IDOF("Physical Items"."Institution"."Institution Code") = 6535)) ucsd LEFT JOIN (SELECT "Physical Items"."Bibliographic Details"."Network Id", "Physical Items"."Holding Details"."Summary Holding" FROM "Physical Items" WHERE (("Location"."Library Code" = 'SRLF') AND ("Holding Details"."Holding Lifecycle" = 'Active') AND ("Bibliographic Details"."Bibliographic Level" = 's') AND (DESCRIPTOR_IDOF("Physical Items"."Institution"."Institution Code") = 6533)) srlf ON srlf."Network Id"=ucsd."Network Id" ORDER BY ucsd."MMS Id"

UCSD and SRLF with an Order By Clause (recommend grabbing the text from the slide and pasting into something like notepad to make it bigger font)
SELECT ucsd."Library Code", ucsd."Location Code", ucsd."Permanent Call Number", ucsd."Summary Holding", ucsd."Title (Complete)", ucsd."MMS Id", ucsd."OCLC Control Number (035a)", ucsd."Network Id", nrlf."Summary Holding" FROM (SELECT "Physical Items"."Location"."Library Code", "Physical Items"."Location"."Location Code", "Physical Items"."Holding Details"."Permanent Call Number", "Physical Items"."Holding Details"."Summary Holding", "Physical Items"."Bibliographic Details"."Title (Complete)", "Physical Items"."Bibliographic Details"."MMS Id", "Physical Items"."Bibliographic Details"."OCLC Control Number (035a)", "Physical Items"."Bibliographic Details"."Network Id" FROM "Physical Items" WHERE ("Location"."Library Code" = 'WongAvery') AND ("Location"."Location Code" = 'jrnl') AND ("Holding Details"."Holding Lifecycle" = 'Active') AND ("Bibliographic Details"."Bibliographic Level" = 's') AND (DESCRIPTOR_IDOF("Physical Items"."Institution"."Institution Code") = 6535)) ucsd LEFT JOIN (SELECT "Physical Items"."Bibliographic Details"."Network Id", "Physical Items"."Holding Details"."Summary Holding" FROM "Physical Items" WHERE ("Location"."Library Code" = 'NRLF') AND ("Holding Details"."Holding Lifecycle" = 'Active') AND ("Bibliographic Details"."Bibliographic Level" = 's') AND (DESCRIPTOR_IDOF("Physical Items"."Institution"."Institution Code") = 6532)) nrlf ON nrlf."Network Id" = ucsd."Network Id" ORDER BY ucsd."MMS Id"

UCSD and NRLF with an Order By Clause (recommend grabbing the text from the slide and pasting into something like notepad to make it bigger font)
Additional Resources (Alma)

- Video: Alma Analytics: Become an Export (highly recommended)
- Ex Libris: Alma Analytics SQL Filter Examples
- Ex Libris: Presentations and Documents - Analytics
- Harvard Wiki: Advanced Formulas
- Ex Libris Developers: Using JOIN in Analytics to combine data from two subject areas
Additional Resources (Generic)

- SQL Operators: The Complete Guide
- SQL Server NULL: Are you making these 7 mistakes?
Questions?

Contact:
gem.stone-logan@ucop.edu
Bonus Slides

For the sake of time, I had to cut a little bit. After this are some extra slides.
Using SQL in Filters

Challenge:
Count how many bibs have science fiction in their subject
Using SQL in Filters: Bib Count

First attempt:

- Use the Num of Titles (Active) Measure
- Filter on subject contains science fiction
Using SQL in Filters: Bib Count

This seems too low.

Data is case sensitive.
Using SQL in Filters: Bib Count

One way to convert the subject to all lowercase:

- Click to edit the filter
- Next to Column click the fx icon
- Then enter the formula similar to how we did when doing the columns
<table>
<thead>
<tr>
<th>Functions</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Locate</td>
<td></td>
</tr>
<tr>
<td>LocateN</td>
<td></td>
</tr>
<tr>
<td><strong>Lower</strong></td>
<td></td>
</tr>
<tr>
<td>Octet_Length</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Selected: Lower

Syntax: LOWER(expr)

Where: `expr` is any expression that evaluates to a character string.

Example

Description: Converts a character string to lowercase.
Using SQL in Filters: Bib Count

Looking at the filter SQL

- Edit the filter like normal
- Check next to **Convert this filter to SQL**
- Click **OK**

`LOWER("Bibliographic Details"."Subjects") LIKE '%science fiction%'`
SQL in Alma Analytics: Union Scenario

Challenge:

- Titles subject only updates monthly
- I have new MMS Ids and I don't know if they're for physical or electronic items
- The built-in union ability doesn't allow for prompting

At least, it doesn't seem I can prompt when I join two subjects. However, I could be wrong about this.
SQL in Alma Analytics: Union Scenario

1. Get the report as close as possible to what you want.
2. Click the Advanced tab and look at the SQL in the SQL Issued section
SELECT
    saw_0,
    saw_1,
    saw_2,
    saw_3
FROM ((SELECT
    "Institution"."Institution Name" saw_0,
    "Bibliographic Details"."MMS Id" saw_1,
    "Bibliographic Details"."Network Id" saw_2,
    "Portfolio"."Portfolio Id" saw_3
FROM "E-Inventory"
WHERE
"Bibliographic Details"."MMS Id" = '991055712629706532'
) UNION (SELECT
    "Institution"."Institution Name" saw_0,
    "Bibliographic Details"."MMS Id" saw_1,
    "Bibliographic Details"."Network Id" saw_2,
    "Physical Item Details"."Physical Item Id" saw_3
FROM "Physical Items"
WHERE
"Bibliographic Details"."MMS Id" = '991055712629706532'
)) t1 ORDER BY saw_0, saw_1, saw_2, saw_3
SQL in Alma Analytics: Union Scenario

3. Clean up the SQL (personal preference)
   a. In the SELECT portion, replace the meaningless column names with actual names
   b. Remove the column I don't want
   c. Remove the remaining saw aliases (like saw_0) but keep column alias at the end (t1)
   d. Delete the extra stuff at the end
SELECT
    "Institution Name",
    "MMS Id",
    "Network Id"
FROM ((SELECT
    "Institution"."Institution Name",
    "Bibliographic Details"."MMS Id",
    "Bibliographic Details"."Network Id",
    "Portfolio"."Portfolio Id"
FROM "E-Inventory"
WHERE
    "Bibliographic Details"."MMS Id" = '991055712629706532'
) UNION (SELECT
    "Institution"."Institution Name",
    "Bibliographic Details"."MMS Id",
    "Bibliographic Details"."Network Id",
    "Physical Item Details"."Physical Item Id"
FROM "Physical Items"
WHERE
    "Bibliographic Details"."MMS Id" = '991055712629706532'
)) t1
SQL in Alma Analytics: Union Scenario

4. Test the new SQL
   a. In Analytics, click **Create**
   b. Click **Analysis**
   c. Click **Create Analysis from Simple Logical SQL**
   d. Copy in SQL to test
   e. Click **OK**
4. Replace the MMS Ids with a Variable Prompt
   a. Figure out what you want to call the variable (for example, prompt_mms_id)
   b. Enclose it in @{} (for example @{prompt_mms_id})
   c. Substitute the MMS Ids for your new prompt.
SELECT
    "Institution Name",
    "MMS Id",
    "Network Id"
FROM ((SELECT
    "Institution"."Institution Name",
    "Bibliographic Details"."MMS Id",
    "Bibliographic Details"."Network Id",
    "Portfolio"."Portfolio Id"
FROM "E-Inventory"
WHERE
"Bibliographic Details"."MMS Id" = ' @prompt_mms_id'
) UNION (SELECT
    "Institution"."Institution Name",
    "Bibliographic Details"."MMS Id",
    "Bibliographic Details"."Network Id",
    "Physical Item Details"."Physical Item Id"
FROM "Physical Items"
WHERE
"Bibliographic Details"."MMS Id" = ' @prompt_mms_id'
)) t1
SQL in Alma Analytics: Union Scenario

5. Enter the new SQL and add a Variable Prompt
   a. Follow the same steps as #4
   b. Click the Prompts link
   c. Click the Plus sign and choose Variable Prompt
SQL in Alma Analytics: Union Scenario

5. Enter the new SQL and add a Variable Prompt (cont)
   d. Next to Presentation Variable type your variable name (i.e. prompt_mms_id) don't include the @ or curly braces
   e. Give it a Label
   f. Click OK and Save the report
SQL in Alma Analytics: Union Scenario

5. Test the report by clicking **Open**
6. Try entering multiple MMS Ids
   a. What does the SQL look like?
   b. What does this look like in a regular report?
   c. Compare the differences:

Test with MMS Id 991055712629706532 (physical)
and 991035455165104701 (electronic)
Our SQL:

"MMS Id" = '991055712629706532;991035455165104701'

Regular SQL:

"MMS Id" IN ('991055712629706532',
'991035455165104701')

Test with MMS Id 991055712629706532 (physical)
and 991035455165104701 (electronic)
SQL in Alma Analytics: Union Scenario

7. Modify our SQL, run it, and look at the results again.

Our SQL:

"MMS Id" IN
('991055712629706532','991035455165104701')

This time, try entering the MMS Ids as:
991055712629706532,991035455165104701
SQL in Alma Analytics: Union Scenario

8. Modify our SQL, run it, and look at the results again.

I have a suspicion it's going to work this time so when I'm making the prompt I edit the "display" to provide some instruction for the user how to properly enter the MMS Ids
This time, try entering the MMS Ids as:
'991055712629706532' ,'991035455165104701'
SELECT
  "Institution Name",
  "MMS Id",
  "Network Id"
FROM ((SELECT
      "Institution"."Institution Name",
      "Bibliographic Details"."MMS Id",
      "Bibliographic Details"."Network Id",
      "Portfolio"."Portfolio Id"
    FROM "E-Inventory"
    WHERE
    "Bibliographic Details"."MMS Id" IN (@{prompt_mms_id})) UNION (SELECT
      "Institution"."Institution Name",
      "Bibliographic Details"."MMS Id",
      "Bibliographic Details"."Network Id",
      "Physical Item Details"."Physical Item Id"
    FROM "Physical Items"
    WHERE
    "Bibliographic Details"."MMS Id" IN (@{prompt_mms_id})) t1
SQL in Alma Analytics: Types of Joins

**FULL OUTER JOIN**
Pulls everything from both locations.

<table>
<thead>
<tr>
<th>Library Code</th>
<th>Summary Holding (UCSD)</th>
<th>Title (Complete)</th>
<th>Summary Holding (NRLF)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>winter/spring 2001/02</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>źesz. 1(1928)-3(1936)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>źesz. 1(1951)</td>
</tr>
</tbody>
</table>

A FULL OUTER JOIN would include too much information because we don't care about NRLF holdings if UCSD doesn't have them. Note: In other systems you can often say "OUTER JOIN" rather than "FULL OUTER JOIN"

https://dataschool.com/how-to-teach-people-sql/sql-join-types-explained-visually/
SQL in Alma Analytics: Types of Joins

INNER JOIN

Only includes holdings if both locations have it.

An INNER JOIN would include too little information because it would only retrieve results where both UCSD and NRLF had holdings.

https://dataschool.com/how-to-teach-people-sql/sql-join-types-explained-visually/