SQL Code Searches

**ADULT ITEMS ADDED DURING SPECIFIC TIME PERIOD**

This SQL code used in the Item Records SQL Search Find Tool will provide you with the number of Adult Items added during a given time. In this example all the numbers within the parenthesis are collection codes relating to our Adult material collections. I run this on a monthly basis and add the resulting numbers to my monthly report.

Select cir.itemrecordID

From CircItemRecords cir (nolock)

Inner Join ItemRecordDetails ird (nolock)

On ird.itemrecordid = cir.itemrecordid

where AssignedCollectionID in (26, 1, 28, 37, 4, 5, 29, 32, 3, 21, 17, 2, 18, 16, 20, 47, 60, 61)

and creationdate between '2019-03-01' and '2019-03-31 23:59:59'

**AUDIO ITEMS ADDED DURING SPECIFIC TIME PERIOD**

This SQL code used in the Item Records SQL Search Find Tool will provide you with the number of Audio Items added during a given time. In this example all the numbers within the parenthesis are collection codes relating to our Audio material collections. I run this on a monthly basis and add the resulting numbers to my monthly report.

Select cir.itemrecordID

From CircItemRecords cir (nolock)

Inner Join ItemRecordDetails ird (nolock)

On ird.itemrecordid = cir.itemrecordid

where AssignedCollectionID in (23, 51, 52, 27, 24, 25, 22, 56, 55, 54)

and creationdate between '2019-03-01' and '2019-03-31 23:59:59'

**DVD ITEMS ADDED DURING SPECIFIC TIME PERIOD**

This SQL code used in the Item Records SQL Search Find Tool will provide you with the number of DVD Items added during a given time. In this example all the numbers within the parenthesis are collection codes relating to our DVD material collection. I run this on a monthly basis and add the resulting numbers to my monthly report.

Select cir.itemrecordID

From CircItemRecords cir (nolock)

Inner Join ItemRecordDetails ird (nolock)

On ird.itemrecordid = cir.itemrecordid

where AssignedCollectionID in (30, 31)

and creationdate between '2019-03-01' and '2019-03-31 23:59:59'

**CHILDREN’S ITEMS ADDED DURING SPECIFIC TIME PERIOD**

This SQL code used in the Item Records SQL Search Find Tool will provide you with the number of juvenile items added during a given time. In this example all the numbers within the parenthesis are collection codes relating to our juvenile material collections. I run this on a monthly basis and add the resulting numbers to my monthly report.

Select cir.itemrecordID

From CircItemRecords cir (nolock)

Inner Join ItemRecordDetails ird (nolock)

On ird.itemrecordid = cir.itemrecordid

where AssignedCollectionID in (10, 46, 48, 7, 34, 33, 6, 45, 15, 8, 14, 9, 41, 11, 39, 13, 49, 12, 44, 59)

and creationdate between '2019-03-01' and '2019-03-31 23:59:59'

**HOW TO FIND “IN-PROCESSING” ITEMS THAT HAVE BEEN IN THIS STATUS FOR OVER A SPECIFIC TIME PERIOD**

This SQL search used in the Item Records SQL Search Find Tool will provide you with any items that have been in the circulation status of “in-processing” for over 3 days. We use this daily to make sure no items have been in processing for over this time frame as our turn-around time from receiving UPS boxes to being on the shelf to circulation is 3 days. If you have a different turn-around goal you can change the # after the – sign. In this example the #15 is the item status code of “in-processing” in our system. You would need to look up your “In-Processing” code.

select itemrecordid from circitemrecords (nolock)

where itemstatusid = 15

and itemstatusdate < DateAdd(dd,-3,GETDATE())

**FIND BIBLIOGRAPHICH RECORDS WITH NO ATTACHED ITEMS**

This SQL search used in the Bibliographic Records SQL Search Find Tool will provide you with any Bibliographic records with no attached items. I use this at least once a week to make sure I don’t have any bib records like this. There should never be a bib record without an item attached to it and if there is I need to investigate what is going on and correct the situation.

select bibliographicrecordID from BibliographicRecords br(nolock)

where bibliographicrecordID not in (select associatedbibrecordID from circitemrecords (nolock))

**FIND CHILDREN RECORDS MISSING THE AR FIELD**

This SQL search used in the Bibliographic Records SQL Search Find Tool will provide you with any Children’s bibliographic record that is missing the 526 AR field. In our system we want every children’s & YA item to have an AR number displayed in our PAC in the patron’s view. Normally this is done at the time new items come into our collection, but as this wasn’t always a priority I developed this SQL search to look for any childrens or YA records that were missing this field so that we could then add the AR field to these records. In this SQL search the numbers listed in the parenthesis are all of our children’s & YA collection codes. You would just need to replace these with your own collection codes.

select distinct br.BibliographicRecordID as RecordID

from polaris.polaris.CircItemRecords cir

join polaris.polaris.BibliographicRecords br on

cir.AssociatedBibRecordID = br.BibliographicRecordID

left join polaris.polaris.BibliographicTags bt on

bt.BibliographicRecordID = br.BibliographicRecordID and bt.TagNumber = 526

where bt.BibliographicTagID is null and cir.AssignedCollectionID in (10,46,48,7,51,34,33,6,45,52,15,14,9,41,24,11,39,13,49,12,44,25,18,16,20,47)

**FIND ANY PATRON RECORD WITH FEES ATTACHED TO AN ITEM NO LONGER IN OUR SYSTEM**

This SQL search used in Patron Records SQL Search Find Tool is used when we receive an item that has been returned, but has already been removed from our collection. We have a policy in place that says that once an item has been billed as lost, it is removed from our database at the end of 6 months. This was not always the case though and before this policy was implemented our policy was no matter how long an item had been in the lost status if a patron returned it in good condition the replacement cost would be waived.

We were told at the time of Polaris implementation that deleted items stayed on a patron’s account even after they were deleted from our system, but what we weren’t told was if an item had been deleted there was no way to find the patron’s account where this item would still be sitting as an outstanding bill. We did not want our database cluttered up with a bunch of “Lost” items, so we began to clean these up now that they would stay on patron’s accounts.

This began causing issues very quickly when we could not find the patron account for a removed lost item so that we could waive the replacement cost. I had to get with Polaris IT for help creating this particular SQL search and it works about 95% of the time, which we found acceptable.

The barcode listed between the 2 apostrophes is the only thing that you change when you are using this search.

select pa.patronid from PatronAcctDeletedItemRecords pad (nolock)

inner join patronaccount pa (nolock)

on pad.itemrecordid = pa.itemrecordid

where pad.barcode = '31834011316705'