

A Gentle Introduction to Sierra SQL

Phil Shirley

Cuyahoga Falls Library

With assistance from

Gem Stone-Logan

Mountain View Public Library

and

Ray Voelker

Public Library of Cincinnati and
Hamilton County



Table of Contents

- **Introduction**
- Chapter 1: SQL Basics
- Chapter 2: Collapsing multiple lines into one
- Chapter 3: Joining tables
- Chapter 4: A query within a query



Introduction

- Resources
- Getting Set Up
- Views (where your data is)
- How to do SQL queries in pgAdmin



What the Sierra Manual Says

"Innovative does not provide end-user support for creating SQL queries. Sierra Direct SQL Access users assume responsibility for the effects of custom SQL. Poorly-constructed queries can compromise system performance."



Official Resources

- Manual: Sierra Direct SQL Access
 - Getting Started
 - Sierra DNA
 - Query Examples
 - Troubleshooting
- Sierra DNA: <http://techdocs.iii.com/sierradna/>
- Innovative Developer Network: <https://developer.iii.com/>
- PostgreSQL
 - <https://www.postgresql.org/docs/10/static/queries.html>



IUG Resources

- IUG Forum – Sierra SQL category
- IUG Clearinghouse
- Past conference presentations

All at <http://innovativeusers.org/>



Resources from other Customers

- Jeremy Goldstein
<https://github.com/jmgold/SQL-Queries>
- Joe Montibello
<https://github.com/joemontibello/iii-sql-queries>
- Public Library of Cincinnati & Hamilton County
<https://github.com/plch/sierra-sql/wiki>
- UNC-Libraries
<https://github.com/UNC-Libraries/III-Sierra-SQL/wiki>



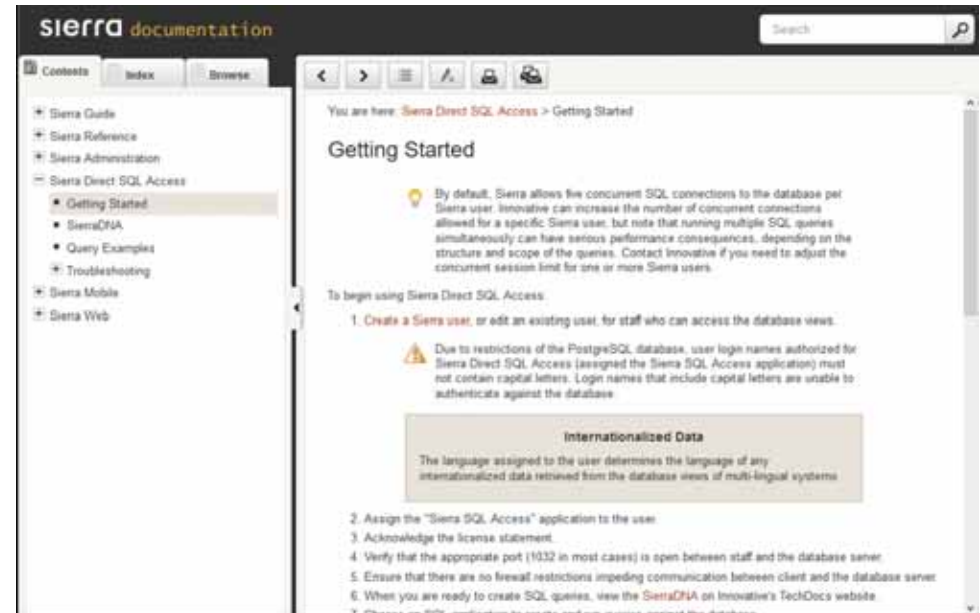
Other Resources

- Stanford free online mini-courses in SQL
<https://lagunita.stanford.edu/courses/DB/2014/SelfPaced/about>
- WiseOwl Training Videos
<https://www.wiseowl.co.uk/sql/videos/sql-selecting-data/>



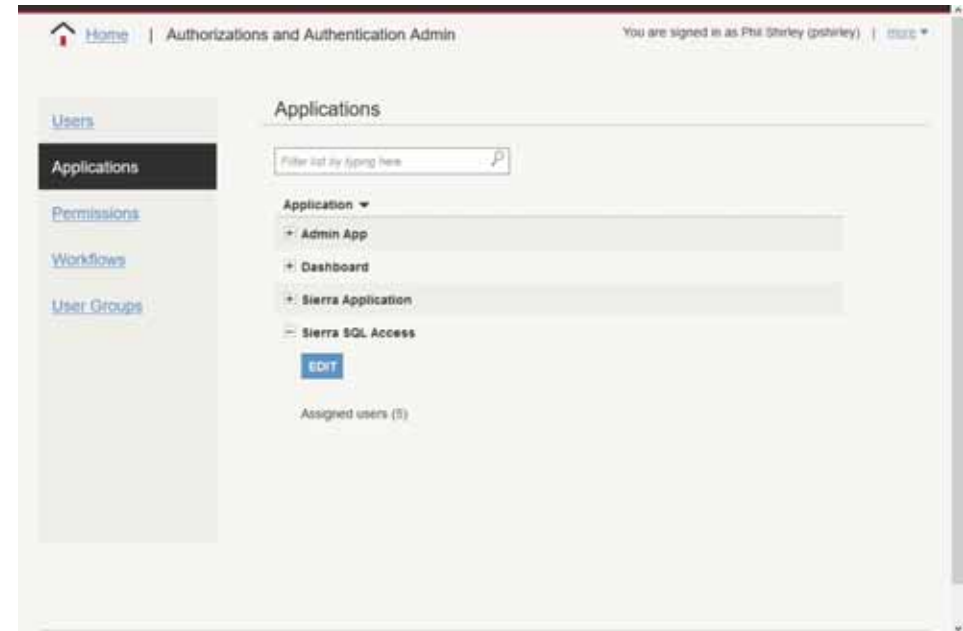
Getting Set Up

- Sierra Documentation
Sierra Direct SQL Access > Getting Started



Getting Set Up

1. Set up user



Getting Set Up

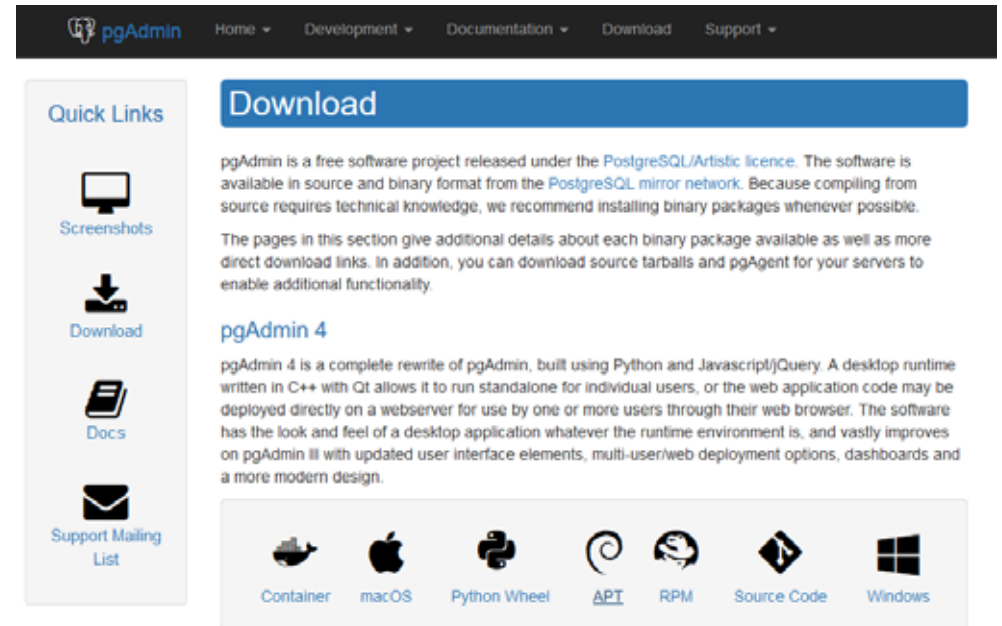
1. Set up user
2. Ensure firewall access to port 1032



Getting Set Up

1. Set up user
2. Ensure firewall access to port 1032
3. Install pgAdmin

<https://www.pgadmin.org/download/>

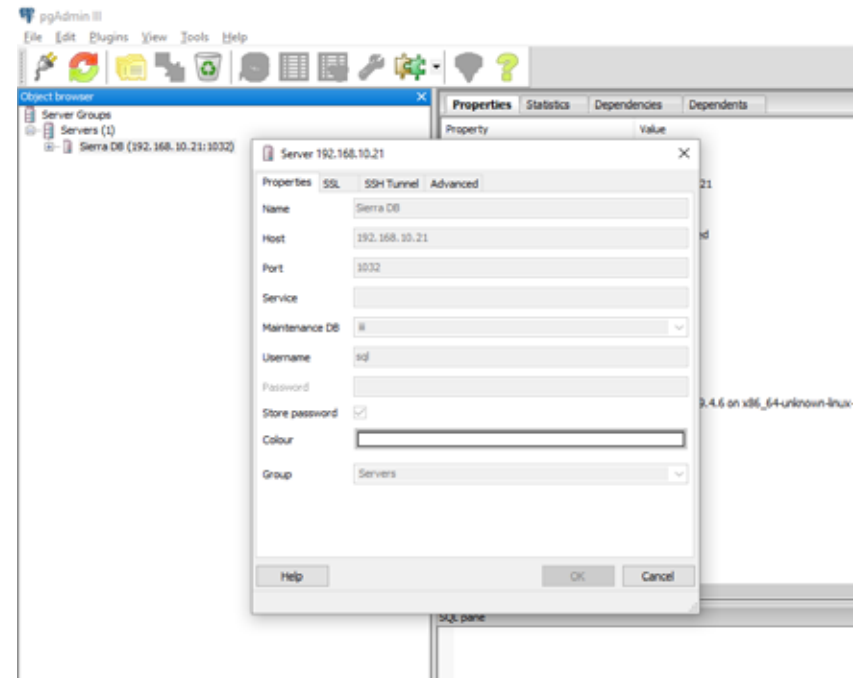


The screenshot shows the pgAdmin website's 'Download' page. At the top, there is a navigation bar with links for Home, Development, Documentation, Download, and Support. Below the navigation bar, there is a 'Quick Links' sidebar with icons for Screenshots, Download, Docs, and Support Mailing List. The main content area features a blue 'Download' header, followed by a paragraph explaining that pgAdmin is a free software project released under the PostgreSQL Artistic license. It then describes the availability of source and binary formats and recommends installing binary packages. Below this, there is a section for 'pgAdmin 4' which describes it as a complete rewrite of pgAdmin, built using Python and JavaScript/Query. At the bottom of the page, there is a row of icons representing different deployment options: Container, macOS, Python Wheel, APT, RPM, Source Code, and Windows.



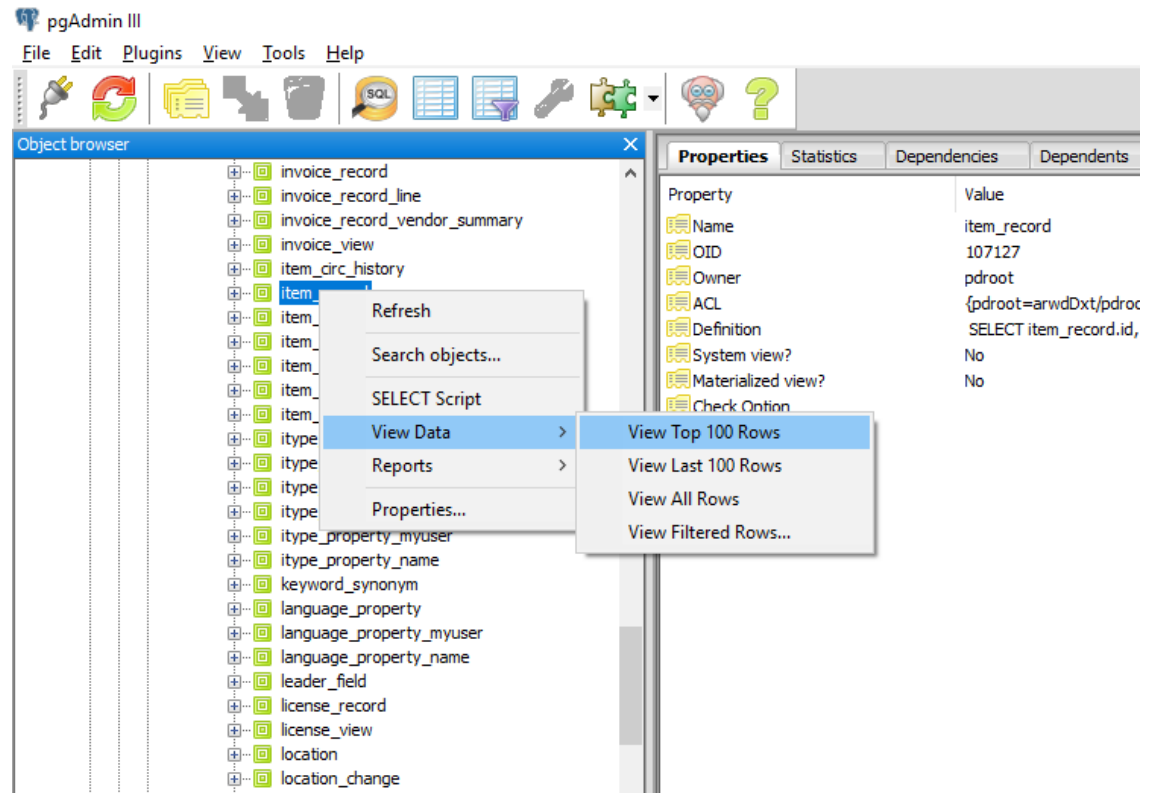
Getting Set Up

1. Set up user
2. Ensure firewall access to port 1032
3. Install pgAdmin
4. Configure pgAdmin
 - Name: Whatever you want
 - Host: Sierra database server



Ways to Learn about a View

- Right click > View Data > View Top 100 Rows

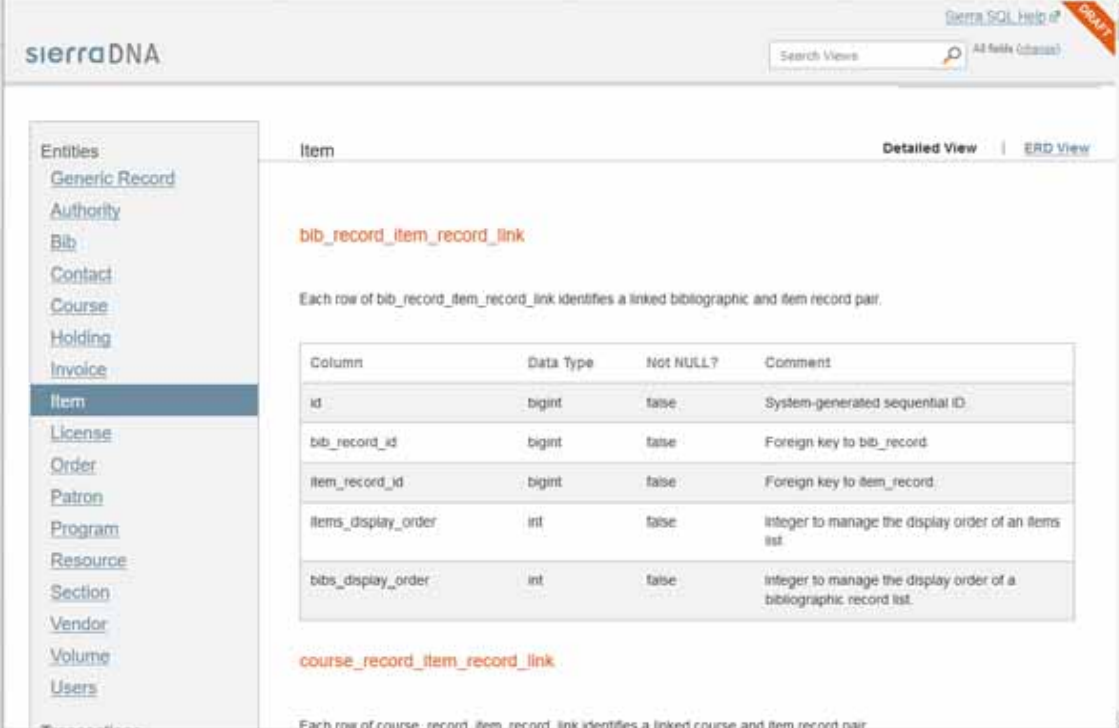


	id bigint	record_id bigint	icode1 integer	icode2 character(1)	itype_code_num smallint	location_code character varying(5)	agency_code_num smallint	item_status_code character varying(3)	is_inherit_loc boolean	price numer
1	3941100652988071	3941100652988071	0	-	230	9cinc	0	⌘	FALSE	0.0000
2	10133550135207265	10133550135207265	0	-	230	re2pl	0	⌘	FALSE	0.0000
3	6474375437855531	6474375437855531	0	-	230	pc2pl	0	⌘	FALSE	0.0000
4	1126350881433238	1126350881433238	0	-	230	tllpl	0	⌘	FALSE	0.0000
5	30962698412117321	30962698412117321	0	-	231	9cler	0	⌘	FALSE	0.0000
6	4504050600725359	4504050600725359	0	-	230	9midp	0	⌘	FALSE	0.0000
7	450972566211	450972566211	0	-	10	non	0	-	FALSE	24.9500
8	1407825856742974	1407825856742974	0	-	230	9lane	0	⌘	FALSE	0.0000
9	3941100652814480	3941100652814480	0	-	231	9cinc	0	⌘	FALSE	0.0000
10	4785525577381147	4785525577381147	0	-	231	9wcpl	0	⌘	FALSE	0.0000
11	450972871833	450972871833	0	-	10	fic	0	-	FALSE	24.9500
12	450972674256	450972674256	0	-	10	jnon	0	-	FALSE	12.3500
13	3378150696429195	3378150696429195	0	-	230	9ascp	0	⌘	FALSE	0.0000
14	1689300833192927	1689300833192927	0	-	231	9ment	0	⌘	FALSE	0.0000
15	450973024698	450973024698	0	-	10	jgrap	0	-	FALSE	26.6500
16	3941100653275535	3941100653275535	0	-	230	9cinc	0	⌘	FALSE	0.0000
17	281925950139744	281925950139744	0	-	230	we4pl	0	⌘	FALSE	0.0000
18	450973133240	450973133240	0	-	42	crock	0	-	FALSE	13.9800
19	281925950139744	281925950139744	0	-	231	we4pl	0	⌘	FALSE	0.0000



Ways to Learn about a View

- Sierra DNA
- (Documentation Navigator)
- <https://techdocs.iii.com/sierradna/>

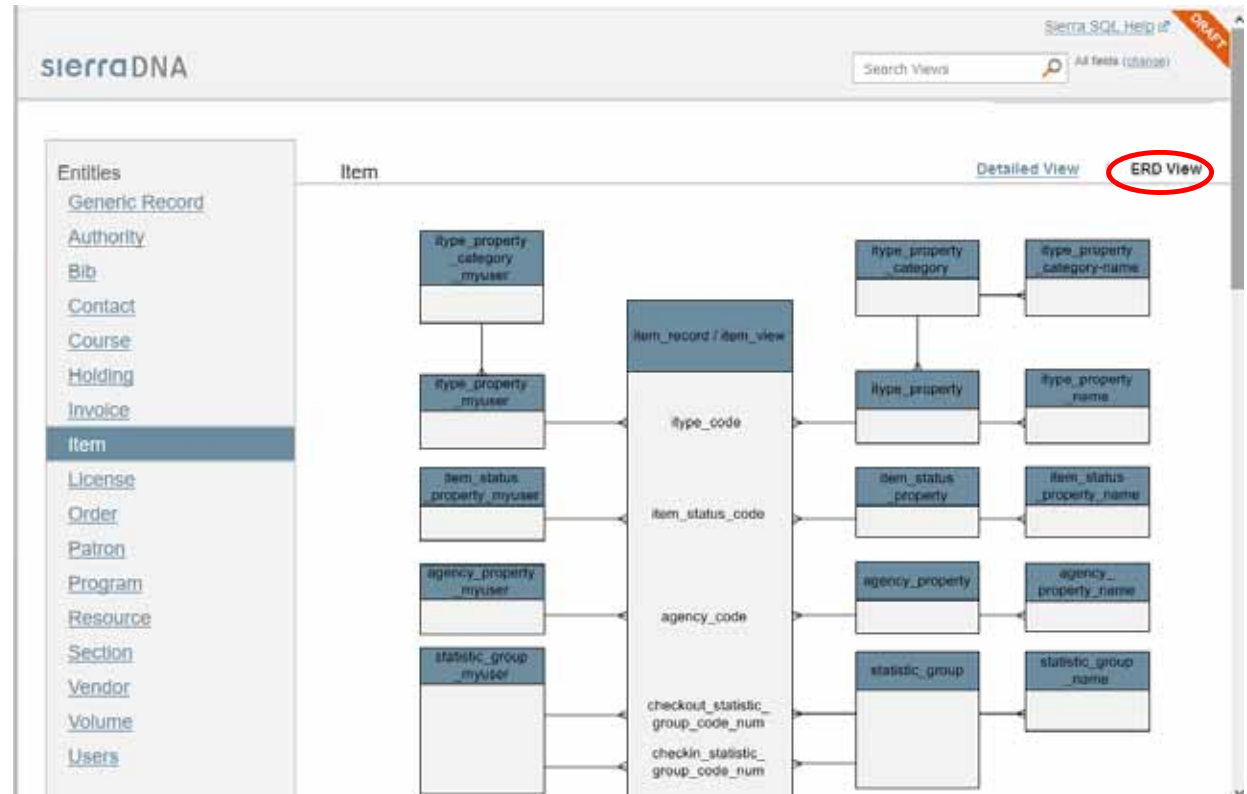


The screenshot shows the SierraDNA documentation interface. On the left is a navigation menu with the following items: Entities, Generic Record, Authority, Bib, Contact, Course, Holding, Invoice, Item (highlighted), License, Order, Patron, Program, Resource, Section, Vendor, Volume, and Users. The main content area is titled 'Item' and shows the 'Detailed View' of the 'bib_record_item_record_link' view. It includes a table with the following columns: Column, Data Type, Not NULL?, and Comment.

Column	Data Type	Not NULL?	Comment
id	bigint	false	System-generated sequential ID
bib_record_id	bigint	false	Foreign key to bib_record
item_record_id	bigint	false	Foreign key to item_record
items_display_order	int	false	Integer to manage the display order of an items list
bibs_display_order	int	false	Integer to manage the display order of a bibliographic record list

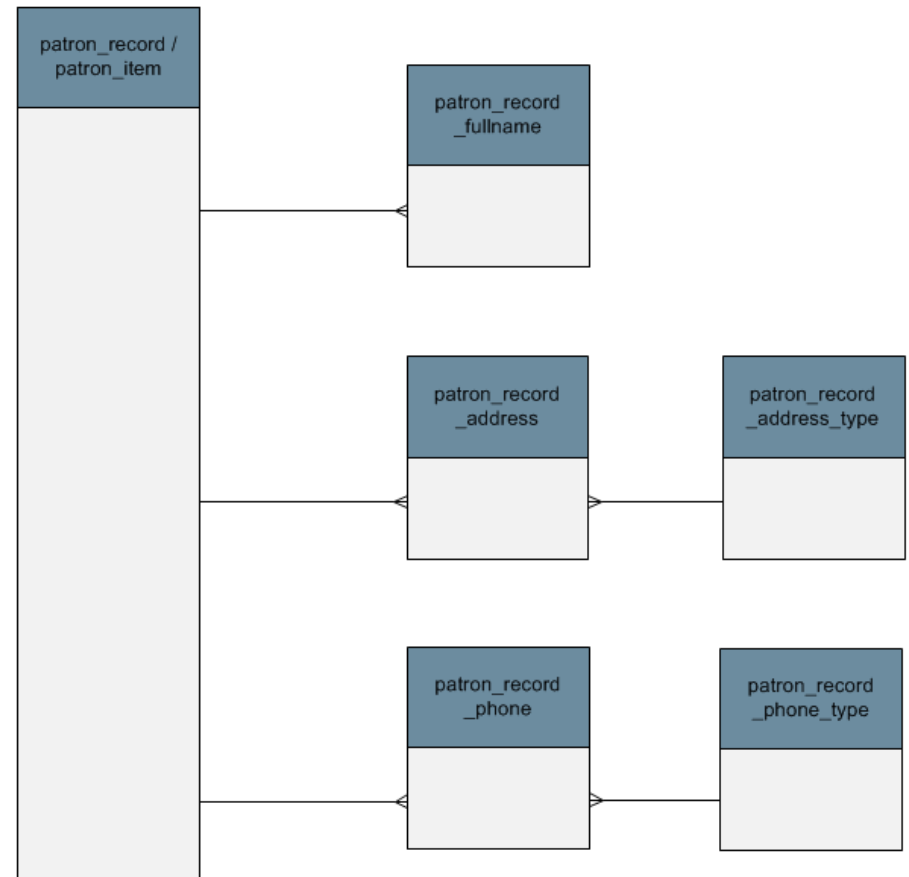


Don't Miss the ERD View



Sierra DNA: Some Views

- patron_record
- patron_view
- patron_record_address
- patron_address_type
- patron_record_fullname
- patron_record_phone
- patron_record_phone_type



item_record vs. item_view

- item_view also includes metadata like the item record number and the record creation date.
- item_view also contains barcode.
- item_record contains is_available_at_library.

- item_view is less efficient



Sierra DNA

item_view

Each row of item_view includes metadata and data for one item record. The contents include identification and circulation information, as well as data that determines how the system handles the record.

Column	Data Type	Not NULL?	Comment
id	bigint	false	System-generated sequential ID.
record_type_code	char	false	Record type code, i.e., 'I'.
record_num	int	false	Record number.
barcode	varchar	false	The item's barcode.
icode1	int	false	The library determines the name and purpose of this code and the code's definition.
icode2	char	false	The library determines the name and purpose of this code and the code's definition.
itype_code_num	int2	false	The type of item. (The library defines item types.)
location_code	varchar	false	The branch location code for the item.



Top Rows

Edit Data - Sierra DB (192.168.10.21:1032) - iii - sierra_view.item_view

File Edit View Tools Help

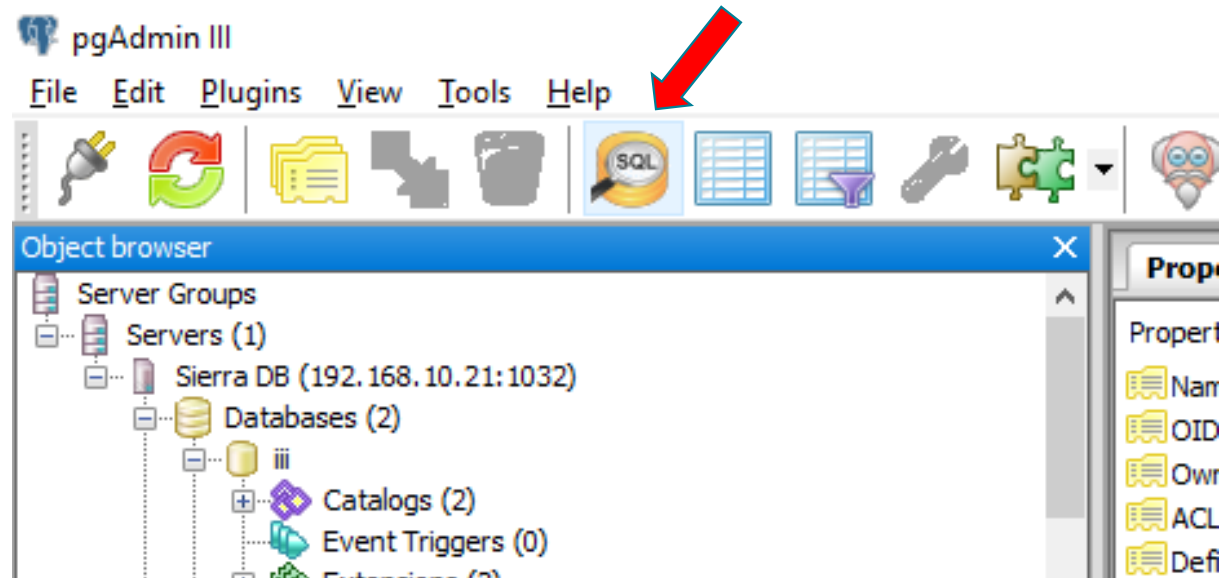
	id bigint	record_type_code character(1)	record_num integer	barcode character varying(1000)	icode1 integer	icode2 character(1)	itype_code_num smallint	locatio charac
1	450972975603	i	1409523	ITMPL004814752	0	-	42	crock
2	450973195937	i	1629857	ITMPL006179675	0	-	10	bio
3	450972914947	i	1348867	ITMPL004814828	0	-	43	jfdbk
4	450972567054	i	1000974	ITMPL000650100	0	u	10	mys
5	450972567230	i	1001150	ITMPL000930098	0	-	10	non
6	450972567962	i	1001882	ITMPL090129718	0	-	10	ov
7	450973001906	i	1435826	ITMPL004225801	0	-	10	non
8	450973028076	i	1461996	ITMPL004480786	0	-	31	dvde
9	450972566373	i	1000293	ITMPL000071794	0	-	10	non
10	450972566866	i	1000786	ITMPL090012575	0	z	10	rflhb
11	450972566325	i	1000245	ITMPL000053420	0	-	10	non
12	450973109849	i	1543769	ITMPL005310941	0	-	10	ov
13	450972567088	i	1001008	ITMPL001491165	0	-	10	jfic
14	450972567238	i	1001158	ITMPL000922939	0	-	10	jnon
15	450973188537	i	1622457	ITMPL006105720	0	-	10	jnon
16	450973059310	i	1493230	ITMPL004785903	0	-	31	dvdt
17	450973160585	i	1544505	ITMPL005831771	0	-	10	ific



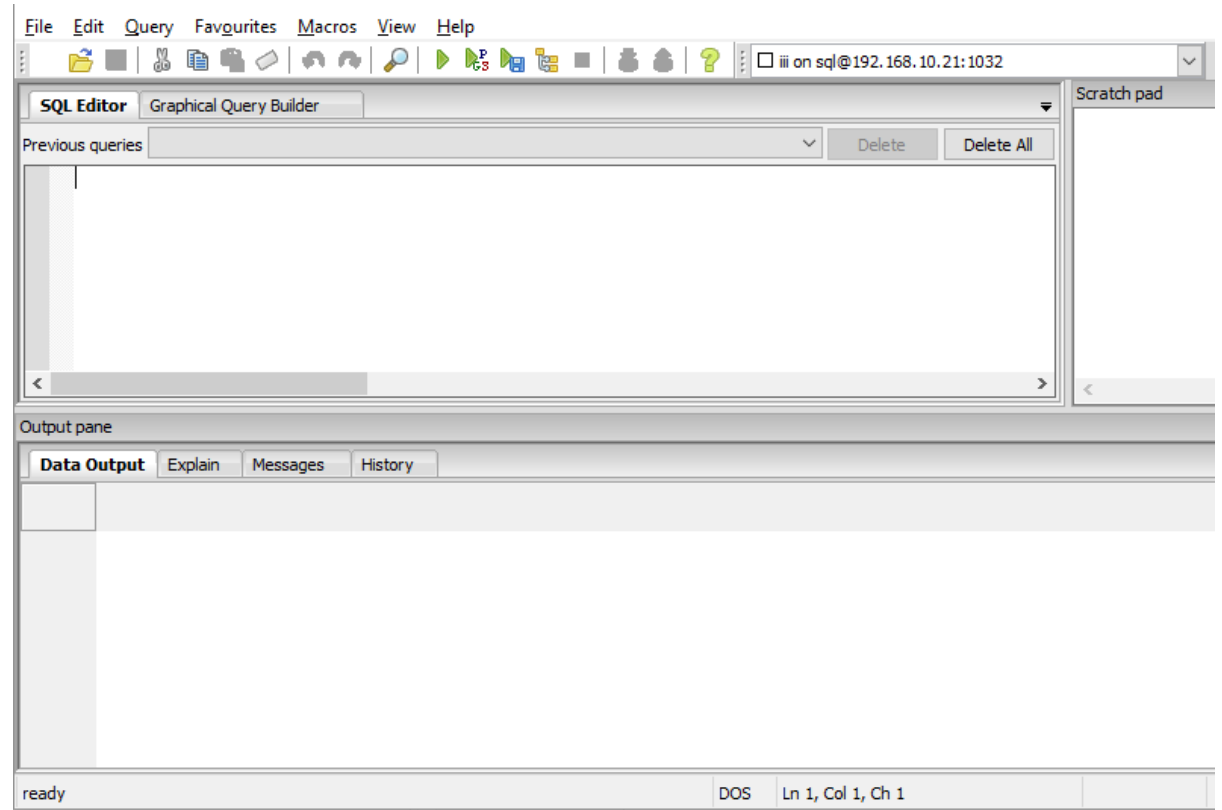
Let's do some SQL!



How to do SQL Queries in pgAdmin



pgAdmin SQL Window



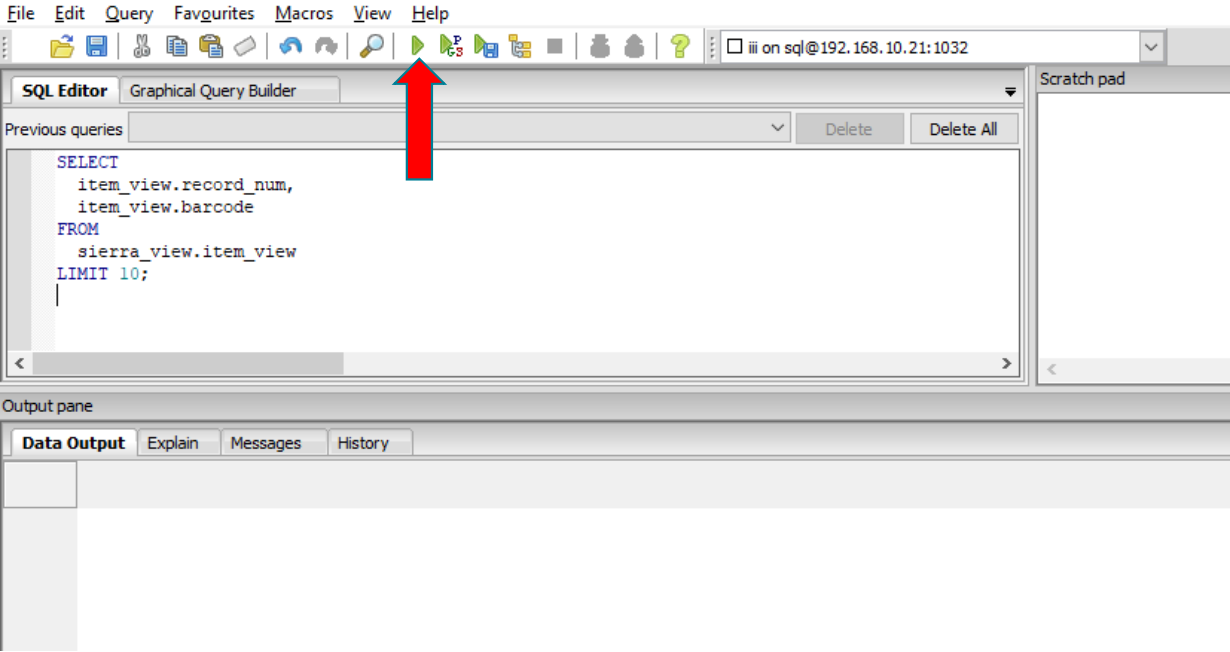
Type your query

A screenshot of a SQL Editor window. The window title is 'SQL Editor' and it has a menu bar with 'File', 'Edit', 'Query', 'Favourites', 'Macros', 'View', and 'Help'. Below the menu bar is a toolbar with various icons. The main text area contains the following SQL query:

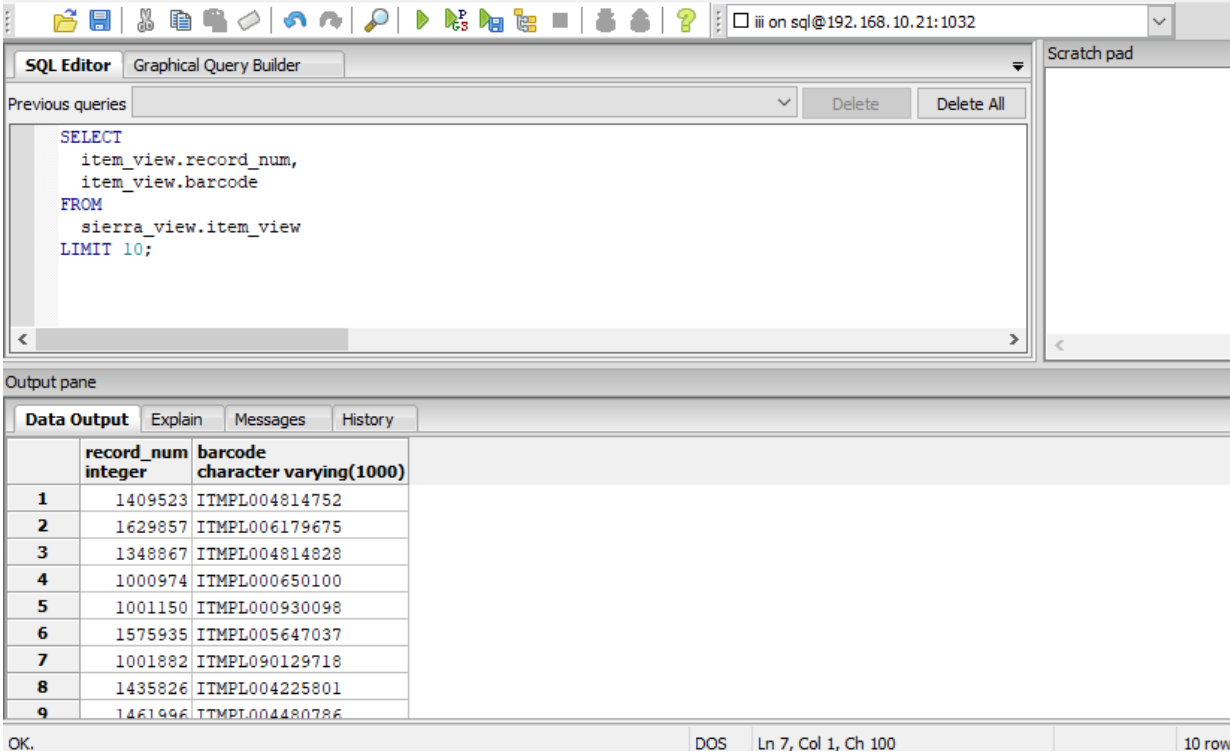
```
SELECT
  item_view.record_num,
  item_view.barcode
FROM
  sierra_view.item_view
LIMIT 10;
```

The query is highlighted in blue. Below the text area is a scroll bar. To the right of the text area is a 'Scratch pad' tab. Below the text area is an 'Output pane' with tabs for 'Data Output', 'Explain', 'Messages', and 'History'. The 'Data Output' tab is selected and is currently empty.

Click the green triangle
(or press F5)



Results appear below



The screenshot shows a SQL Editor window with the following SQL query:

```
SELECT
  item_view.record_num,
  item_view.barcode
FROM
  sierra_view.item_view
LIMIT 10;
```

The Output pane displays the results in a table with the following data:

	record_num integer	barcode character varying(1000)
1	1409523	ITMPL004814752
2	1629857	ITMPL006179675
3	1348867	ITMPL004814828
4	1000974	ITMPL000650100
5	1001150	ITMPL000930098
6	1575935	ITMPL005647037
7	1001882	ITMPL090129718
8	1435826	ITMPL004225801
9	1461996	ITMPL004480786

A red arrow points to the first row of the table. The status bar at the bottom indicates 'OK.', 'DOS', 'Ln 7, Col 1, Ch 100', and '10 row'.



Tips on Using pgAdmin

- Queries: Save, Open
- F5 to execute query
- Alt-break to stop a query
- Saving results: File – Export
 - Exports as CSV
 - But name the file with the extension .txt so you can import it into Excel nicely



Table of Contents

- Introduction
- **Chapter 1: SQL Basics**
- Chapter 2: Collapsing multiple lines into one
- Chapter 3: Joining tables
- Chapter 4: A query within a query



Chapter 1. SQL Basics

- SELECT
- FROM
- LIMIT
- AS
- ORDER BY
- WHERE
- id2reckey()



A Basic SQL Query

```
SELECT
  item_view.record_num,
  item_view.barcode
FROM
  sierra_view.item_view
LIMIT 10;
```



Another way to write it

```
select item_view.record_num, item_view.barcode from sierra_view.item_view limit 10;
```



Yet another way to write it

```
SELECT
  item_view.record_num,
  item_view.barcode

FROM
  sierra_view.item_view

LIMIT 10;
```



Results

Output pane

	record_num integer	barcode character varying(1000)
1	1409523	ITMPL004814752
2	1629857	ITMPL006179675
3	1348867	ITMPL004814828
4	1000974	ITMPL000650100
5	1001150	ITMPL000930098
6	1001882	ITMPL090129718
7	1435826	ITMPL004225801
8	1461996	ITMPL004480786
9	1000293	ITMPL000071794
10	1000786	ITMPL090012575



Provide Aliases for Column Headings with AS

```
SELECT
  item_view.record_num AS "Record number",
  item_view.barcode AS "Barcode"
FROM
  sierra_view.item_view
LIMIT 10;
```



Results

Output pane

	Data Output	Explain	Messages	History
	Record number integer	Barcode character varying(1000)		
1	1409523	ITMPL004814752		
2	1629857	ITMPL006179675		
3	1348867	ITMPL004814828		
4	1000974	ITMPL000650100		
5	1001150	ITMPL000930098		
6	1001882	ITMPL090129718		
7	1435826	ITMPL004225801		
8	1461996	ITMPL004480786		
9	1000293	ITMPL000071794		
10	1000786	ITMPL090012575		



Add a Table Alias

```
SELECT
  i.record_num AS "Record number",
  i.barcode AS "Barcode"
FROM
  sierra_view.item_view i
LIMIT 10;
```



ORDER BY

SELECT

b.id AS "Record number",

b.title AS "Title"

FROM

sierra_view.bib_view b

ORDER BY

b.id

LIMIT 10;



Output pane

Data Output Explain Messages History		
	Record number bigint	Title character varying(1000)
1	420907795030	Literary market place : LMP.
2	420907795040	The magic school bus, inside the human body
3	420907795047	Oh my baby bear!
4	420907795064	A book of love poetry
5	420907795121	The foxman
6	420907795125	The art of landscape and seascape painting
7	420907795149	Crooning : a collection
8	420907795170	Sliver : a novel
9	420907795172	The firm
10	420907795178	From the kingdom of memory : reminiscences



WHERE

SELECT

```
o.id AS "Record number",  
o.order_status_code AS "Status Code"
```

FROM

```
sierra_view.order_record o
```

WHERE

```
o.order_status_code = 'o'
```

LIMIT 10;



Output pane

Data Output

Explain

Messages

	Record number bigint	Status Code character(1)
1	476742617843	o
2	476742617849	o
3	476742617859	o
4	476742612351	o
5	476742577407	o
6	476742612430	o
7	476742611546	o
8	476742617874	o
9	476742616301	o
10	476742612504	o



WHERE: = or <>

- WHERE o.order_status_code = 'o';
- WHERE o.volume_count = 1;
- WHERE o.order_status_code <> 'o';
- WHERE o.volume_count <> 1;

- != is the same as <>



WHERE: BETWEEN

- WHERE o.volume_count BETWEEN 1 AND 5;
- WHERE b.title BETWEEN 'A' AND 'B';

- WHERE UPPER(b.title) BETWEEN 'A' AND 'B';
- WHERE LOWER(b.title) BETWEEN 'a' AND 'b';



WHERE: IN

- WHERE o.material_type_code IN ('a', 'b', 'g');
- WHERE o.volume_count IN (1,2,7);



WHERE: LIKE or NOT LIKE

_ (underscore) exactly one character

% zero or more characters (anything, or nothing)

- WHERE LOWER(b.title) LIKE '%librar%';
- WHERE b.title LIKE ' _';



WHERE Summary

- = or <>
- BETWEEN
- IN
- LIKE or NOT LIKE



id2reckey

```
SELECT
  id2reckey(o.id) AS "Record number",
  o.order_status_code AS "Status Code"
FROM
  sierra_view.order_record o
WHERE
  o.order_status_code = 'o'
LIMIT 10;
```



Output pane

Data Output

Explain

Messages

His

	Record number character varying	Status Code character(1)
1	o1247987	o
2	o1247993	o
3	o1248003	o
4	o1242495	o
5	o1207551	o
6	o1242574	o
7	o1241690	o
8	o1248018	o
9	o1246445	o
10	o1242648	o



So far we know

- SELECT
- FROM
- LIMIT
- AS
- ORDER BY
- WHERE
- id2reckey()



Table of Contents

- Introduction
- Chapter 1: SQL Basics
- **Chapter 2: Collapsing multiple lines into one**
- Chapter 3: Joining tables
- Chapter 4: A query within a query



Chapter 2: Collapsing multiple lines into one

- Aggregate functions
 - MAX()
 - COUNT()
- GROUP BY
- HAVING



What is the maximum number of checkouts any picture book has?



Aggregate functions: MAX

```
SELECT
  MAX(i.checkout_total)
FROM
  sierra_view.item_view i
WHERE
  i.location_code = 'jpic';
```



Output pane

	Data Output	Explain	Messages
	max integer		
1	219		



How many items are there in the location 'jpic'?



Aggregate functions: COUNT(*)

```
SELECT
  COUNT(*)
FROM
  sierra_view.item_view i
WHERE
  i.location_code = 'jpic';
```



Output pane

	Data Output	Explain	Messages
	count bigint		
1	8765		



What's the count of items in 'jpic' for each status code?



Let's add GROUP BY

```
SELECT
  COUNT(*),
  i.item_status_code
FROM
  sierra_view.item_view i
WHERE
  i.location_code = 'jpic'
GROUP BY
  i.item_status_code;
```



Output pane

Data Output Explain Messages

	count bigint	item_status_code character varying(3)
1	19	u
2	9	w
3	1	!
4	2	e
5	15	@
6	1	(
7	8615	-
8	63	n
9	2	p
10	13	\$
11	25	m



Only show the statuses where the count is less than 20



Add HAVING

```
SELECT
  COUNT(i.id),
  i.item_status_code
FROM
  sierra_view.item_view i
WHERE
  i.location_code = 'jpic'
GROUP BY
  i.item_status_code
HAVING
  COUNT(i.id) < 20;
```



Output pane

Data Output Explain Messages H

	count bigint	item_status_code character varying(3)
1	19	u
2	9	w
3	1	!
4	2	e
5	15	@
6	1	(
7	2	p
8	13	\$



So far:

- SELECT
- FROM
- LIMIT
- AS
- ORDER BY
- WHERE
- MAX
- COUNT
- GROUP BY
- HAVING



Table of Contents

- Introduction
- Chapter 1: SQL Basics
- Chapter 2: Collapsing multiple lines into one
- **Chapter 3: Joining tables**
- Chapter 4: A query within a query



Chapter 3: Joining tables

- The SQL
- How to join two views
- Two kinds of JOIN
- How to deal with varfield_view



The SQL (snippet)

FROM

something s

JOIN

other_thing o

ON

s.field = o.other_field



How to join two views?

- One table has to know the other table's ID number.

OR

- A third table has to know the ID numbers of both tables.



How to join two views? We'll look at:

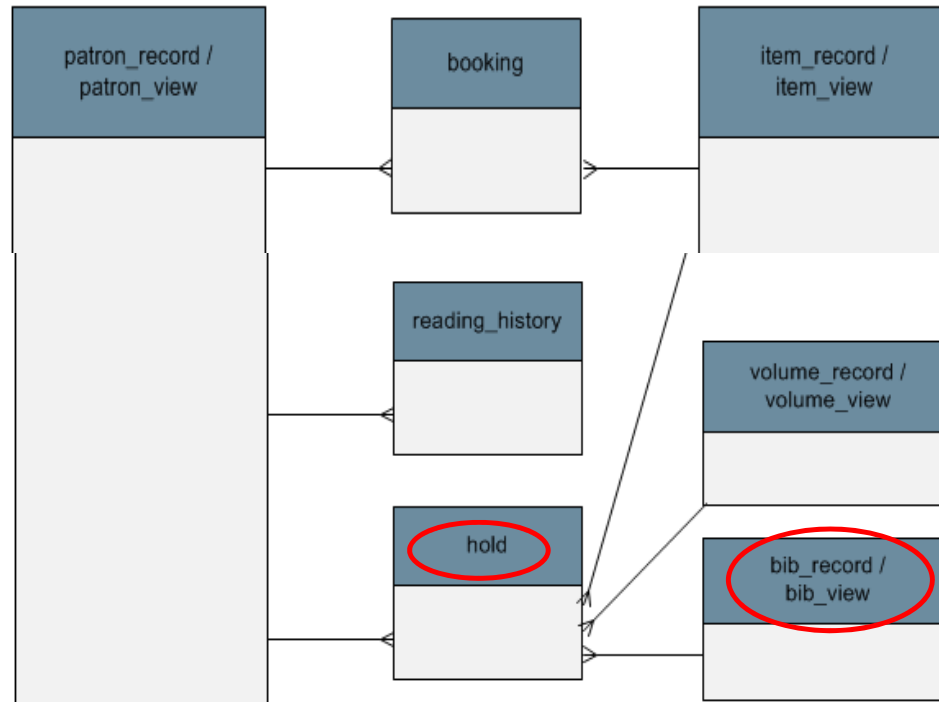
- Sierra DNA (web site)
 - Detailed view
 - ERD view
- Top rows (in pgAdmin)



Bibs with (bib-level) holds



A direct link: bib_view and hold



Sierra DNA for hold

hold

Each row of hold describes a bibliographic, item, or volume hold.

Column	Data Type	Not NULL?	Comment
id	bigint	false	System-generated sequential ID.
patron_record_id	bigint	false	Foreign key to patron_record.
<u>record_id</u>	bigint	false	<u>Foreign key to record.</u>
placed_on	timestamp	false	Date the hold was placed



Top rows for bib_view and hold

Edit Data - Sierra DB (192.168.10.21:1032) - iii - sierra_view.bib_view

File Edit View Tools Help

100 rows

	<u>id</u> bigint	record_type_code character(1)	record_num integer	language characte
1	420908065471	b	1270463	eng
2	420908172785	b	1377777	eng
3	420907795149	b	1000141	eng
4	420907795678	b	1000670	eng

Edit Data - Sierra DB (192.168.10.21:1032) - iii - sierra_view.hold

File Edit View Tools Help

9 rows

	<u>id</u> bigint	patron_record_id bigint	<u>record_id</u> bigint	pla tin
1	483160	481037351623	3941100649441521	20:
2	481974	481037468270	420908493552	20:
3	486438	9852105222256298	450973194005	20:
4	487101	3096705790300726	450973164331	20:

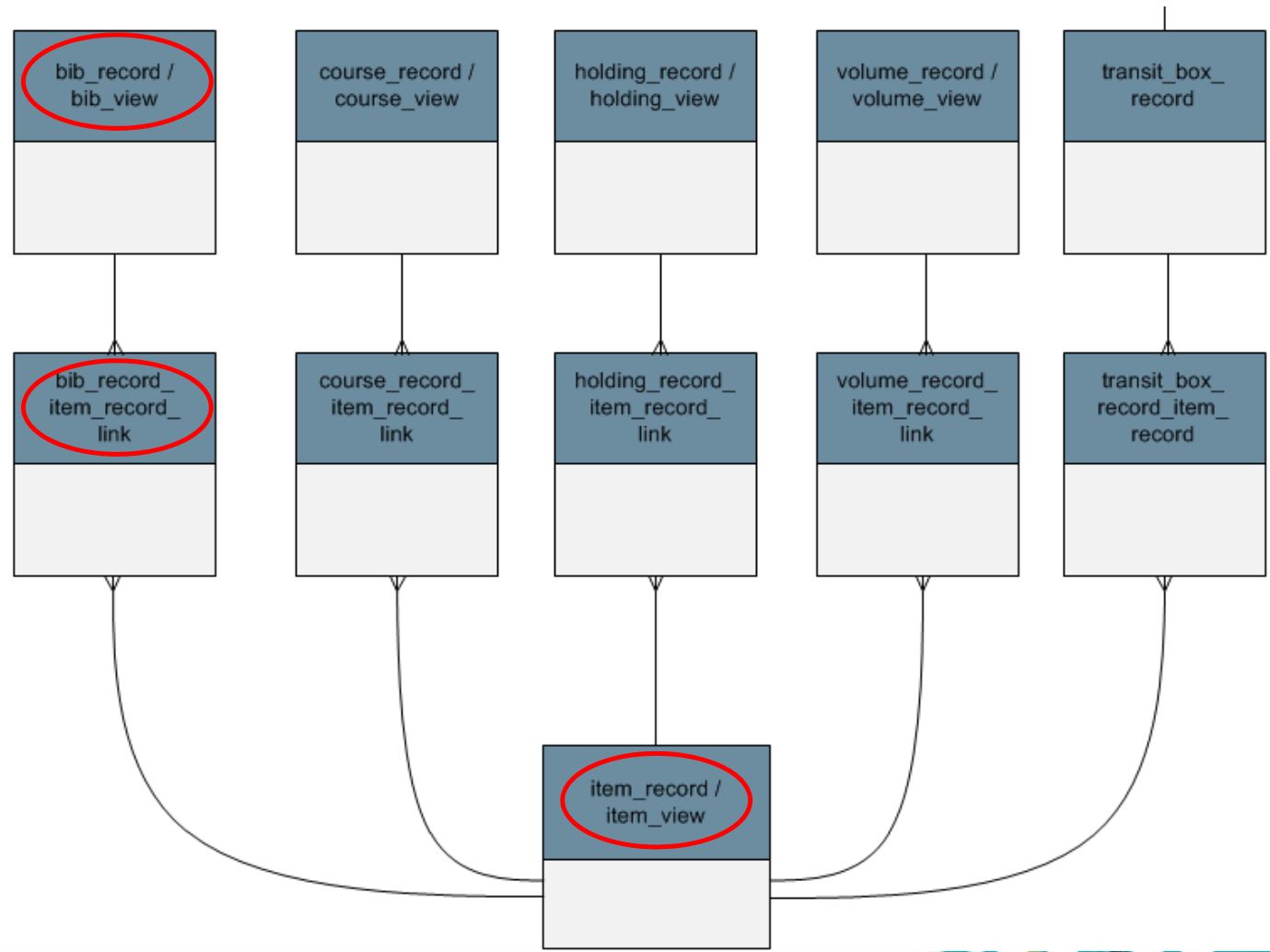


Using JOIN with bib_view and hold

```
SELECT
  b.record_num,
  h.id
FROM
  sierra_view.bib_view b
JOIN
  sierra_view.hold h
ON h.record_id = b.id;
```



Sometimes there's a "link" view



Sierra DNA

bib_record_item_record_link

Each row of bib_record_item_record_link identifies a linked bibliographic and item record pair.

Column	Data Type	Not NULL?	Comment
id	bigint	false	System-generated sequential ID.
<u>bib_record_id</u>	bigint	false	<u>Foreign key to bib_record.</u>
<u>item_record_id</u>	bigint	false	<u>Foreign key to item_record.</u>
items_display_order	int	false	Integer to manage the display order of an items list.
bibs_display_order	int	false	Integer to manage the display order of a bibliographic record list.



Edit Data - Sierra DB (192.168.10.21:1032) - iii - sierra_view.bib_record_item_record_link

File Edit View Tools Help

100 rows

	id bigint	bib_record_id bigint	item_record_id bigint	items_display_order integer	bibs_display_order integer
1	110484	420908192612	450973016239	0	0
2	168664	420908268026	450973099199	0	0
3	326793	420908532670	450973209082		0
4	82	420908348172	450973129978	0	0
5	84	420908348197	450973128732	0	0
6	339059	30962668348043236	30962698412814308	0	0
7	246963	420908372871	450973159892		0
8	234485	5629920442032659	5629950506803731	0	0
9	246970	420908467025	450973159899		0
10	274415	420908487713	450973176058		0
11	326797	420908536452	450973209086		0
12	234490	1126320818405145	1126350883176217	0	0
13	314605	420908533437	450973201587		0



JOIN bib, link, and item

```
SELECT *  
FROM  
  sierra_view.bib_record b  
JOIN  
  sierra_view.bib_record_item_record_link L  
ON b.id = L.bib_record_id  
JOIN  
  sierra_view.item_record i  
ON i.id = L.item_record_id;
```



When to use a different kinds of JOIN

- JOIN
- LEFT JOIN



An earlier example

```
SELECT
    b.record_num,
    b.title,
    h.id
FROM sierra_view.bib_view b
JOIN sierra_view.hold h
ON h.record_id = b.id
ORDER BY b.title;
```



Results

Output pane

	record_num integer	title character varying(1000)	id bigint
1	1569295	"C" is for corpse : a Kinsey Millhone mystery	484848
2	1703624	"D" is for deadbeat	457766
3	1764291	12 Rules for life : an antidote to chaos	482335
4	1764291	12 Rules for life : an antidote to chaos	481754
5	1765746	12 Strong (Rental)	470220
6	1765746	12 Strong (Rental)	480175
7	1765745	12 Strong (Rental)	480174
8	1765746	12 Strong (Rental)	475560
9	1765745	12 Strong (Rental)	468942
10	1765745	12 Strong (Rental)	467844
11	1765745	12 Strong (Rental)	468580
12	1765746	12 Strong (Rental)	469224
13	1765746	12 Strong (Rental)	474459
14	1765745	12 Strong (Rental)	467879

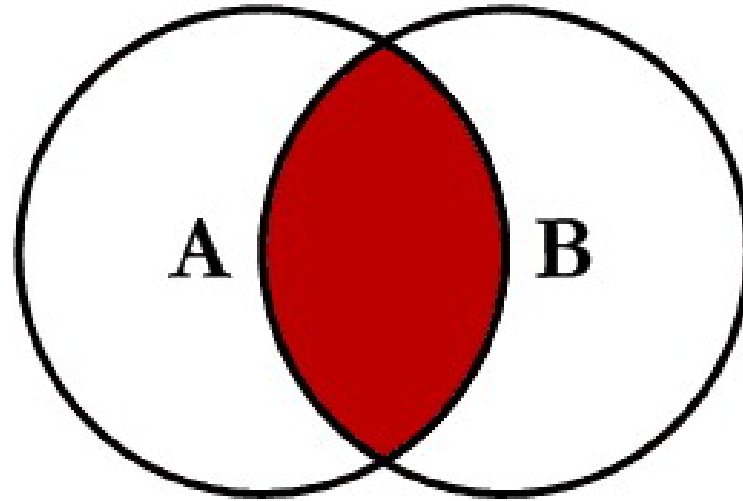
OK.



This doesn't include bibs with zero bib-level holds



JOIN (INNER JOIN):
Bibs with title-level holds

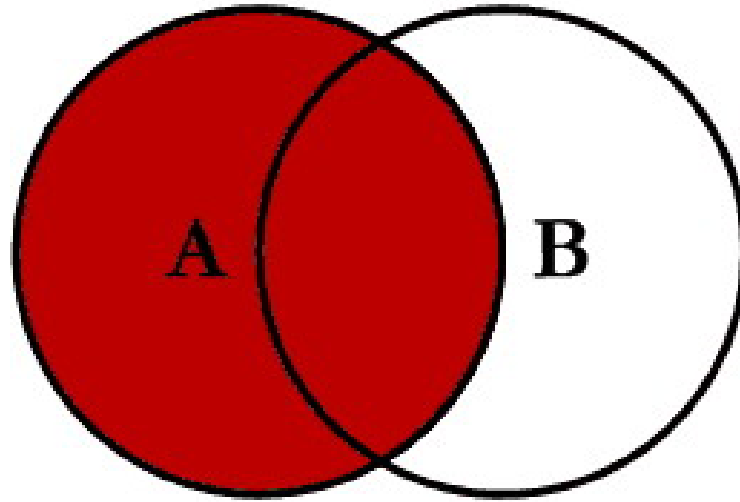


bib_view

hold



LEFT JOIN (LEFT OUTER JOIN):
All bibs, some of which have title-level holds



bib_view

hold



Include bibs with no bib-level holds

```
SELECT
    b.record_num,
    b.title,
    h.id
FROM sierra_view.bib_view b
LEFT JOIN sierra_view.hold h
ON h.record_id = b.id;
```



Problem child: varfield_view



varfield_view: Each line has

- The content of a variable-length field
- The ID of the record it's from
- The kind of record it's from (b, i, p, etc.)
- The field tag
- The MARC tag and indicators



	id bigint	record_id bigint	record_type_code character(1)	record_num integer	varfield_type_code character(1)	marc_tag character varying(3)	marc_ind1 character(1)	marc_ind2 character(1)	occ_num integer	field_content character varying(20001)
74	5994477	481037404117	p	1066965	x				2	Fri Jul 11 2010: Bill \$18.9
75	5994479	481037404117	p	1066965	x				3	Fri Jul 02 2010: Bill \$18.9
76	485202	420907795149	b	1000141	t	245	1	0	0	aCrooning : ba collection
77	485203	420907795149	b	1000141	r	300			0	a287 p.
78	485204	420907795149	b	1000141	c	092			0	a814 Dun
79	485205	420907795149	b	1000141	p	260			0	aNew York : bSimon and Sch
80	485206	420907795149	b	1000141	a	100	1		0	aDunne, John Gregory, d19:
81	485207	420907795149	b	1000141	n	590			0	aDUNNE 363121 \$19.95
82	485208	420907795149	b	1000141	l	010			0	a90009404
83	485209	420907795149	b	1000141	y	005			0	19900105082902.2
84	485210	420907795149	b	1000141	y	040			1	aDLC cDLC
85	485211	420907795149	b	1000141	i	020			0	z0671672362
86	222097	450972975603	i	1409523	b				0	ITMPL004814752
87	222098	450972975603	i	1409523	p				0	CD ROCK
88	222099	450972975603	i	1409523	l				0	\$11.98 1CD 1/08
89	12399802	450973195937	i	1629857	b				0	ITMPL006179675
90	152297	450972914947	i	1348867	b				0	ITMPL004814828
91	152298	450972914947	i	1348867	p				0	J CD BOOK
92	152299	450972914947	i	1348867	l				0	\$90.00 17CD 7/05
93	152300	450972914947	i	1348867	m				0	17 cds.
94	152301	450972914947	i	1348867	x				0	RTI CLEANED ALL 17 DISCS
95	176	450972567054	i	1000974	b				0	ITMPL000650100
96	177	450972567054	i	1000974	a	947			0	aITMPL000650100 bMN cMYST

Get MARC field 245 for bib records

```
SELECT
  b.record_num,
  v.field_content
FROM
  sierra_view.bib_view b
JOIN sierra_view.varfield_view v
ON v.record_num = b.record_num
WHERE
  v.marc_tag = '245';
```



Output pane

Data Output Explain Messages History

	record_num integer	field_content character varying(20001)
1	1000113	aThe foxman / cby Gary Paulsen.
2	1000162	aSliver : ba novel / cIra Levin.
3	1756201	aNo room for small dreams : bcourage, imagination, and the making of modern Israel / cShimon Peres.
4	1000573	aRealms of fantasy / cMalcolm Edwards & Robert Holdstock.
5	1000682	aThe river bank : band other stories from the Wind in the willows / cKenneth Grahame ; illustrated by Inga
6	1623482	aClassical harp h[electronic resource] / cPatricia Spero.
7	1000410	aThe Christmas album h[sound recording].
8	1767558	aSoul / cJoey W. Hill.
9	1309072	aThe costume party / cstory and pictures by Victoria Chess.
10	1000519	aWashington County, Ohio, to 1980 : ba collection of topical & family sketches / cWashington County Historic



Two fields? Two joins

JOIN

```
sierra_view.varfield_view v1 ON (b.record_num = v1.record_num AND  
v1.marc_tag = '245')
```

JOIN

```
sierra_view.varfield_view v2 ON (b.record_num = v2.record_num AND  
v2.marc_tag = '260')
```



SELECT

b.record_num AS "Bib#",
v1.field_content AS "245",
v2.field_content AS "260"

FROM

sierra_view.bib_view b

JOIN

sierra_view.varfield_view v1 on (b.record_num = v1.record_num AND v1.marc_tag =
'245')

JOIN

sierra_view.varfield_view v2 on (b.record_num = v2.record_num AND v2.marc_tag =
'260');



Output pane

	Data Output	Explain	Messages	History
	Bib# integer	245 character varying(20001)	260 character varying(20001)	
1	1000113	aThe foxman / cby Gary Paulsen.	aNew York : bPuffin Books, c1990.	
2	1000162	aSliver : ba novel / cIra Levin.	aNew York : bBantam Books, c1991.	
3	1000573	aRealms of fantasy / cMalcolm Edwards & Robert Holdo	aGarden City, N.Y. : bDoubleday, c1983.	
4	1000682	aThe river bank : band other stories from the Wind i	aCambridge, Massachusetts : bCandlewick Press, cc1996.	
5	1623482	aClassical harp h[electronic resource] / cPatricia S	aSuffolk, Eng. ; aBoulder, Colo. : bNew World Music, c2	
6	1000410	aThe Christmas album h[sound recording].	a[S.I.] : bPhilips Digital Classics, cp1990.	
7	1309072	aThe costume party / cstory and pictures by Victoria	aLa Jolla, CA : bKane/Miller, c2005.	
8	1000519	aWashington County, Ohio, to 1980 : ba collection of	a[Marietta] : bThe Society, cc1980.	
9	1309664	aPeyton Place / cby Grace Metalious ; with a new int	aBoston : bNortheastern University Press, c1999.	
10	1559864	aThe lion and the bird / cMarianne Dubuc ; translate	aNew York : bEnchanted Lion Books, c2014.	
11	1003075	aFamous kings & queens of England & Scotland / cby J	aNew York : bArco Pub. Co., cc1977.	
12	1004207	aHow to write a play / cby Raymond Hull.	aCincinnati, Ohio : bWriter's Digest Books, cc1983.	
13	1007964	aMotor auto engines and electrical systems / ceditor	aNew York : bMotor, cc1977.	
14	1003974	aPassage to Pontefract / cJean Plaidy	aNew York : bPutnam c1982	



Repeated field? Put all instances on one line

```
SELECT
```

```
  b.record_num as "Bib#",
```

```
  STRING_AGG(v.field_content, ';' order by occ_num) as "500"
```

```
FROM
```

```
  sierra_view.bib_view b
```

```
JOIN
```

```
  sierra_view.varfield_view v on (b.record_num = v.record_num and v.marc_tag =  
'500')
```

```
GROUP BY 1;
```



Output pane

Data Output			
	Explain	Messages	History
	Bib# integer	500 text	
1	1000022	a"The directory of the American book publishing industry with industry yellow pages."; ab1000	
2	1000056	aOriginally published under title: The Penguin book of love poetry.; aIncludes indexes.	
3	1000113	a"First published in the United States by Nelson in 1977"--T.p. verso.	
4	1000117	aIncludes index.	
5	1000217	aIncludes index.	
6	1000219	a"Parts of this book were previously published as Painting with light / by Betty Lou Schlemm ;	
7	1000226	aIncludes index.	
8	1000319	a"Part no. 7819"--Cover.	
9	1000393	aIncludes index.	
10	1000399	aCompact disc. Analog recording. -- Music from the play based on T.S. Eliot's Old Possum's book	
11	1000401	aCompact disc.	
12	1000410	aCompact disc. -- Includes pamphlet (8 p.).	
13	1000419	aOn spine: First United Methodist Church history.	
14	1000441	aCover title.; aOn spine: The history of Dyke College, 1848-1973.	



So far:

- SELECT
- FROM
- LIMIT
- AS
- ORDER BY
- WHERE
- MAX
- COUNT
- GROUP BY
- HAVING
- JOIN ON
- LEFT JOIN ON
- JOIN ON (_ and _)
- STRING_AGG



Table of Contents

- Introduction
- Chapter 1: SQL Basics
- Chapter 2: Collapsing multiple lines into one
- Chapter 3: Joining tables
- **Chapter 4: A query within a query**



Items Added Last Year that Cost More than Average

- Query 1: Average cost of items added last year
- Query 2: Items added last year that cost more than that
- Then... combine these into one query!



What was the average price of items added last year?

```
SELECT
  AVG(v.price)
FROM
  sierra_view.item_view v
WHERE
  v.record_creation_date_gmt BETWEEN '1/1/2017' AND '12/31/2017';
```



Output pane

	Data Output	Explain	Messages	Hi
	avg numeric			
1	20.8908830010493179			



Items added last year with prices greater than that

```
SELECT
  id2reckey(i.id),
  i.barcode
FROM
  sierra_view.item_view i
WHERE
  i.price > 20
AND
  i.record_creation_date_gmt BETWEEN '1/1/2017' AND '12/31/2017':
```



Can we combine them?

```
SELECT id2reckey(i.id), i.barcode  
FROM sierra_view.item_view i  
WHERE i.price > (  
  
)  
AND i.record_creation_date_gmt BETWEEN '1/1/2017' AND '12/31/2017';
```



Yes! (This is called a subquery.)

```
SELECT id2reckey(i1.id), i1.barcode
FROM sierra_view.item_view i1
WHERE i1.price > (
    SELECT AVG(i2.price)
    FROM sierra_view.item_view i2
    WHERE i2.record_creation_date_gmt BETWEEN '1/1/2017' AND '12/31/2017'
)
AND i1.record_creation_date_gmt BETWEEN '1/1/2017' AND '12/31/2017';
```



Output pane

	id2reckey character varying	barcode character varying(1000)
1	i1629857	ITMPL006179675
2	i1630541	itmpl006186894
3	i1628200	ITMPL006164511
4	i1628219	ITMPL006164826
5	i1641418	ITMPL006299762
6	i1642299	ITMPL006306773
7	i1631892	ITMPL006202170
8	i1632702	ITMPL006207096
9	i1643774	ITMPL006319545
10	i1637300	ITMPL006254445
11	i1649281	ITMPL006378491
12	i1635306	ITMPL006239099
13	i1646239	ITMPL006338289
14	i1628239	ITMPL006165310



Can we include a column that gives the average number?



```
SELECT id2reckey(i1.id), i1.barcode, (  
    SELECT AVG(i3.price)  
    FROM sierra_view.item_view i3  
    WHERE i3.record_creation_date_gmt BETWEEN '1/1/2017' AND '12/31/2017'  
)  
FROM sierra_view.item_view i1  
WHERE i1.price > (  
    SELECT AVG(i2.price)  
    FROM sierra_view.item_view i2  
    WHERE i2.record_creation_date_gmt BETWEEN '1/1/2017' AND '12/31/2017'  
)  
AND i1.record_creation_date_gmt BETWEEN '1/1/2017' AND '12/31/2017'
```



Output pane

Data Output Explain Messages History

	id2reckey character varying	barcode character varying(1000)	round numeric
1	i1629857	ITMPL006179675	21
2	i1630541	itmpl006186894	21
3	i1628200	ITMPL006164511	21
4	i1628219	ITMPL006164826	21
5	i1641418	ITMPL006299762	21
6	i1642299	ITMPL006306773	21
7	i1631892	ITMPL006202170	21
8	i1632702	ITMPL006207096	21
9	i1643774	ITMPL006319545	21
10	i1637300	ITMPL006254445	21
11	i1649281	ITMPL006378491	21
12	i1635306	ITMPL006239099	21
13	i1646239	ITMPL006338289	21
14	i1628239	ITMPL006165310	21

OK



What we learned:

- SELECT
- FROM
- LIMIT
- AS
- ORDER BY
- WHERE
- MAX
- COUNT
- GROUP BY
- HAVING
- JOIN ON
- LEFT JOIN ON
- JOIN ON (_ and _)
- STRING_AGG
- Subqueries



Questions?

pshirley@fallslibrary.org

