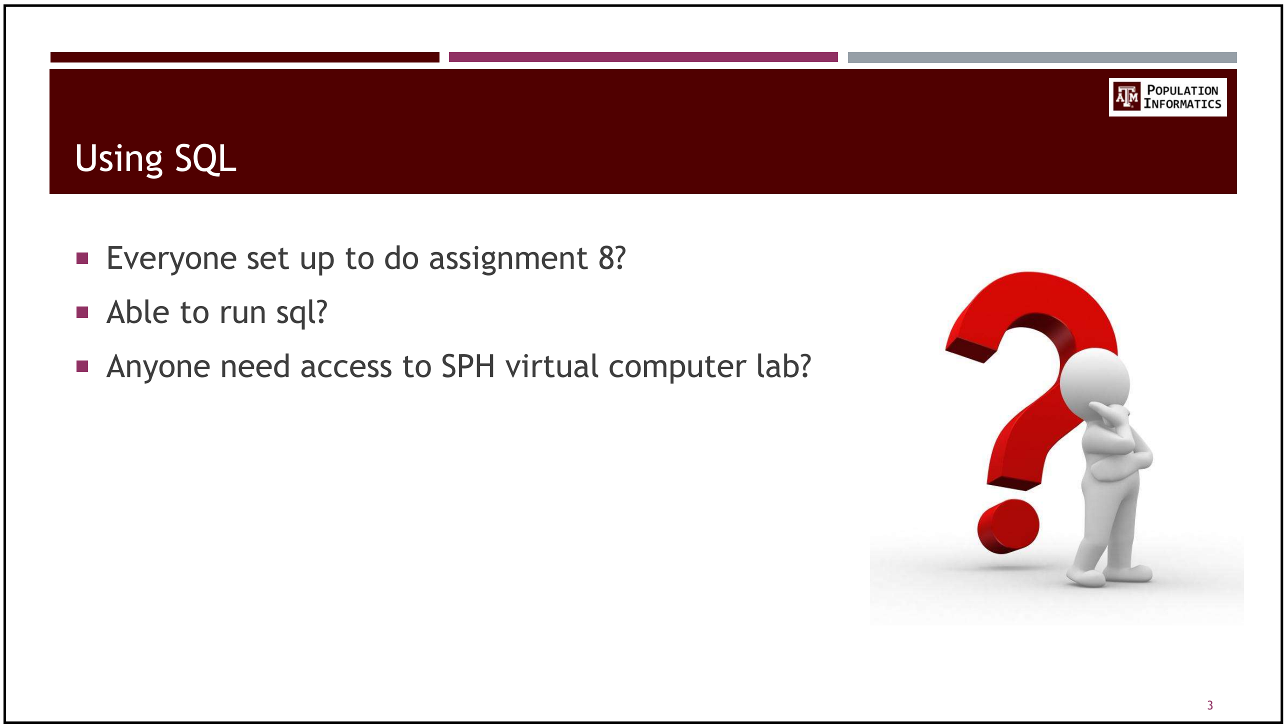
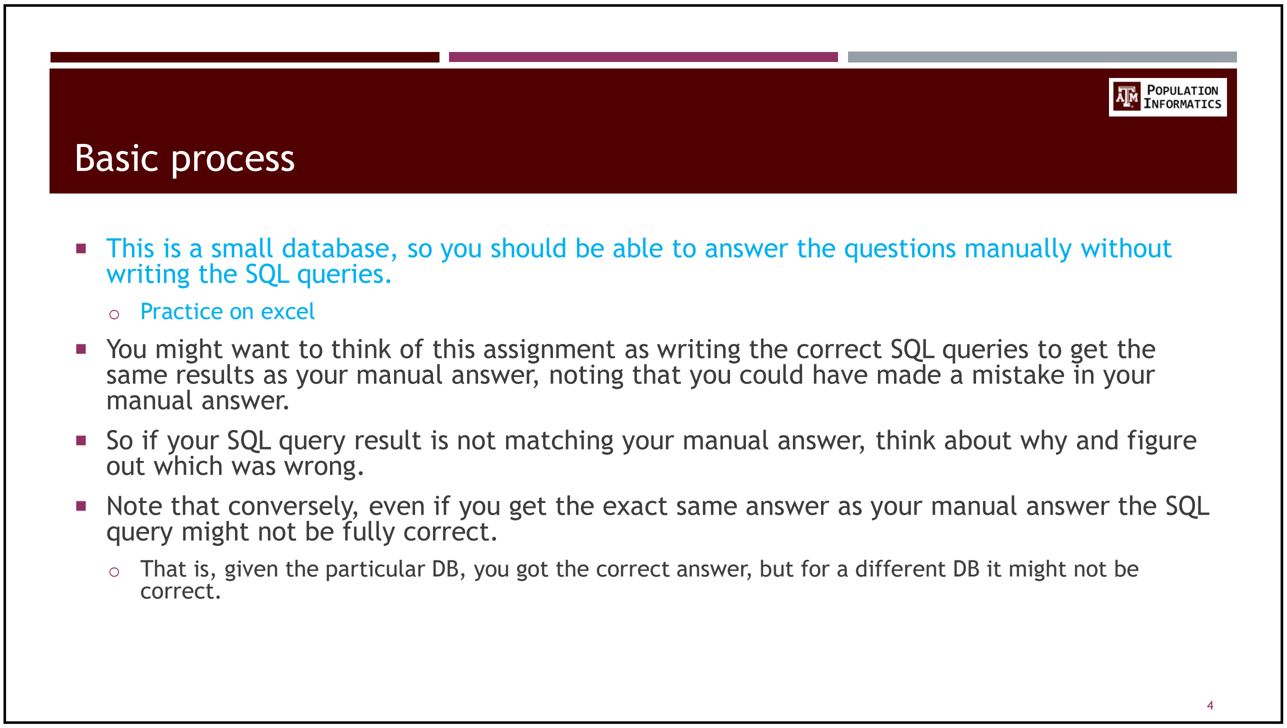


Using SQL

- Everyone set up to do assignment 8?
- Able to run sql?
- Anyone need access to SPH virtual computer lab?



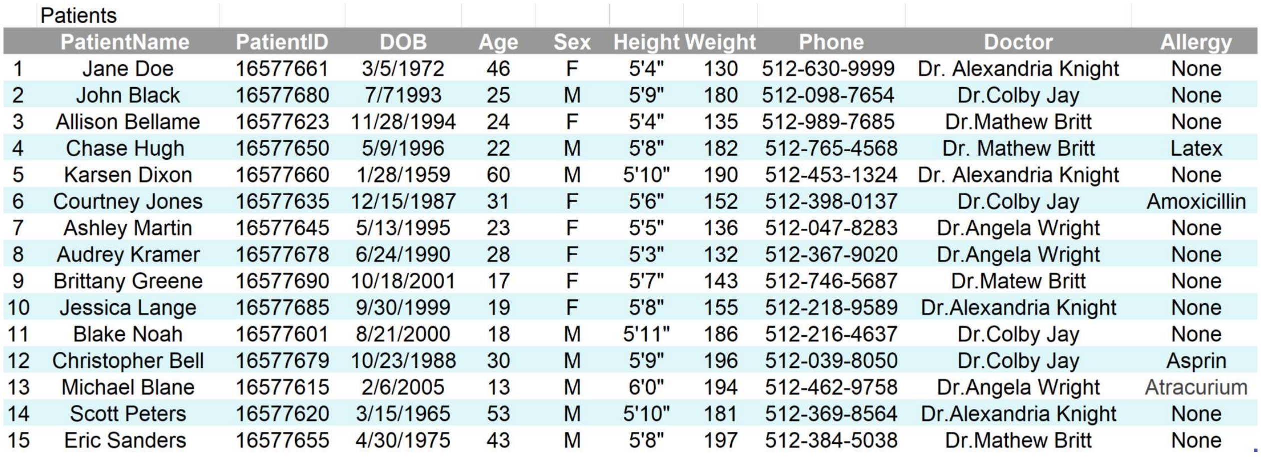
Basic process

- This is a small database, so you should be able to answer the questions manually without writing the SQL queries.
  - Practice on excel
- You might want to think of this assignment as writing the correct SQL queries to get the same results as your manual answer, noting that you could have made a mistake in your manual answer.
- So if your SQL query result is not matching your manual answer, think about why and figure out which was wrong.
- Note that conversely, even if you get the exact same answer as your manual answer the SQL query might not be fully correct.
  - That is, given the particular DB, you got the correct answer, but for a different DB it might not be correct.

Try it: Using the handout excel sheet from last week

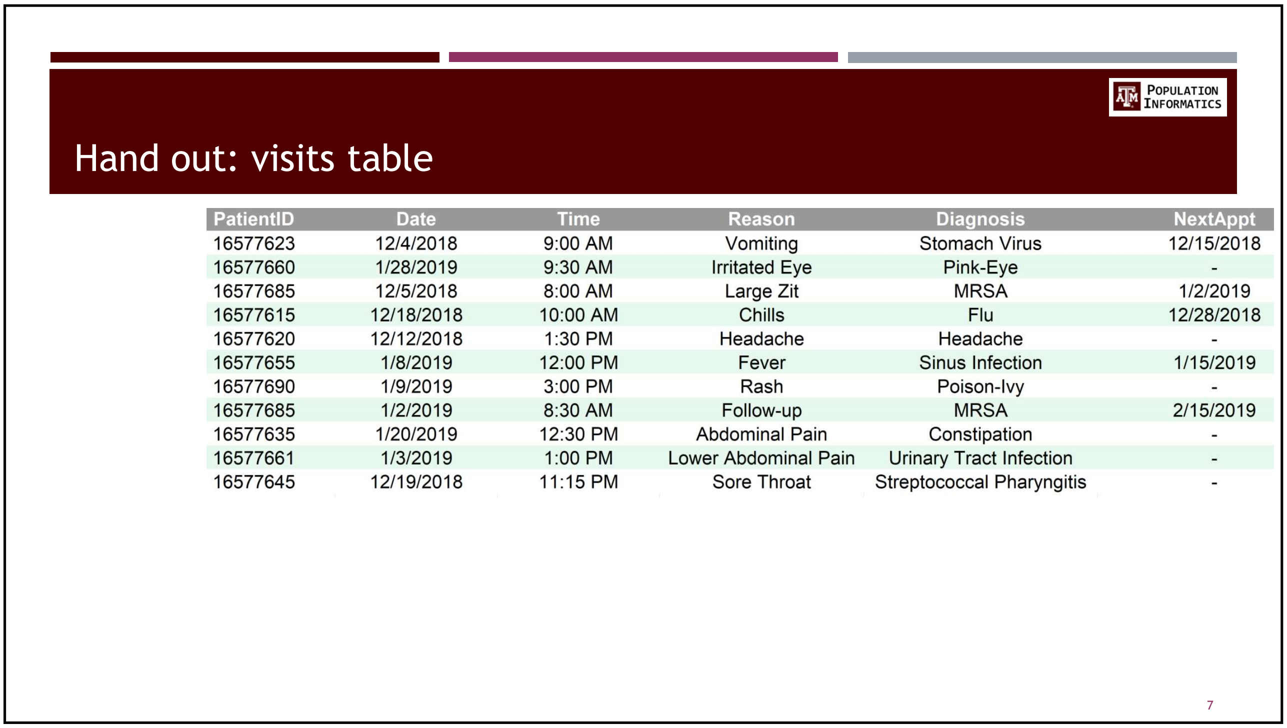
- Fill out the gforms
  - https://forms.gle/UGRxo4XZsGuBwtnW9
- How many patients visited for vomiting?
- Who (what are their names) visited for vomiting?
- Which tables did you use to get this information?
- How did you combine the tables? That is, what columns did you use to combine the tables?

Handout Patients Table



Patients

	PatientName	PatientID	DOB	Age	Sex	Height	Weight	Phone	Doctor	Allergy
1	Jane Doe	16577661	3/5/1972	46	F	5'4"	130	512-630-9999	Dr. Alexandria Knight	None
2	John Black	16577680	7/71993	25	M	5'9"	180	512-098-7654	Dr.Colby Jay	None
3	Allison Bellame	16577623	11/28/1994	24	F	5'4"	135	512-989-7685	Dr.Mathew Britt	None
4	Chase Hugh	16577650	5/9/1996	22	M	5'8"	182	512-765-4568	Dr. Mathew Britt	Latex
5	Karsen Dixon	16577660	1/28/1959	60	M	5'10"	190	512-453-1324	Dr. Alexandria Knight	None
6	Courtney Jones	16577635	12/15/1987	31	F	5'6"	152	512-398-0137	Dr.Colby Jay	Amoxicillin
7	Ashley Martin	16577645	5/13/1995	23	F	5'5"	136	512-047-8283	Dr.Angela Wright	None
8	Audrey Kramer	16577678	6/24/1990	28	F	5'3"	132	512-367-9020	Dr.Angela Wright	None
9	Brittany Greene	16577690	10/18/2001	17	F	5'7"	143	512-746-5687	Dr.Matew Britt	None
10	Jessica Lange	16577685	9/30/1999	19	F	5'8"	155	512-218-9589	Dr.Alexandria Knight	None
11	Blake Noah	16577601	8/21/2000	18	M	5'11"	186	512-216-4637	Dr.Colby Jay	None
12	Christopher Bell	16577679	10/23/1988	30	M	5'9"	196	512-039-8050	Dr.Colby Jay	Asprin
13	Michael Blane	16577615	2/6/2005	13	M	6'0"	194	512-462-9758	Dr.Angela Wright	Atracurium
14	Scott Peters	16577620	3/15/1965	53	M	5'10"	181	512-369-8564	Dr.Alexandria Knight	None
15	Eric Sanders	16577655	4/30/1975	43	M	5'8"	197	512-384-5038	Dr.Mathew Britt	None



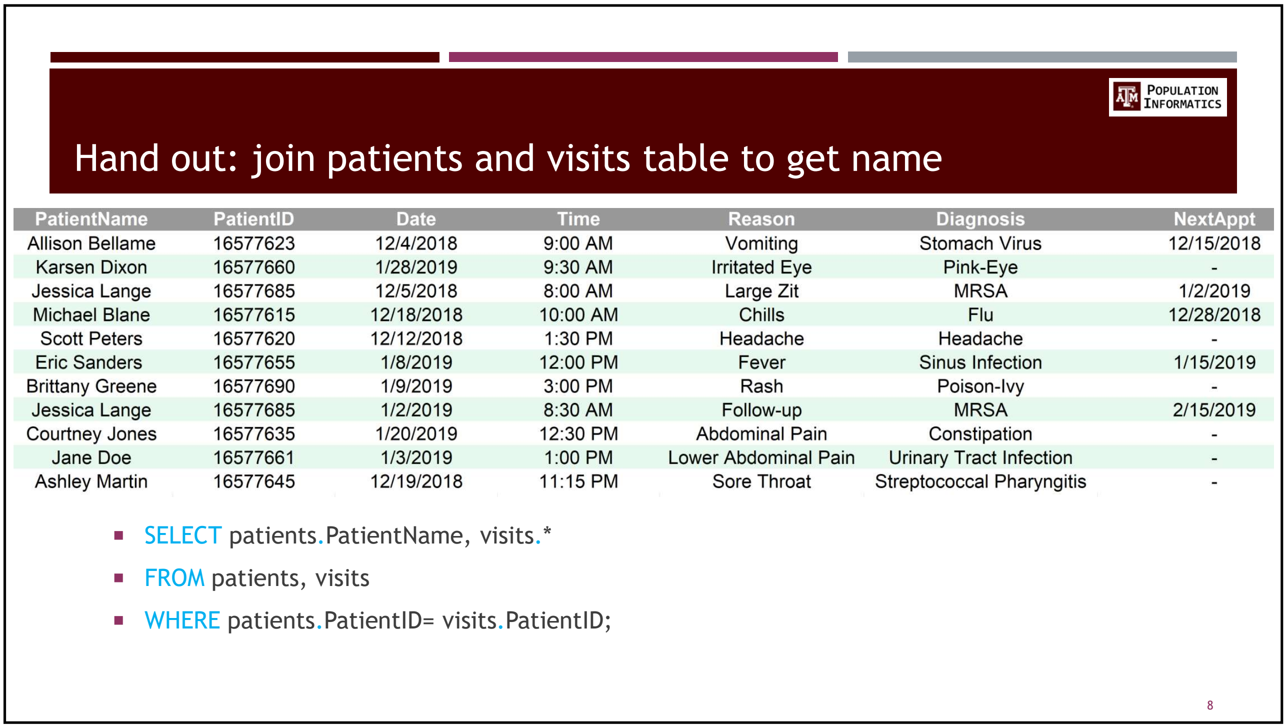
POPULATION INFORMATICS

Hand out: visits table

PatientID	Date	Time	Reason	Diagnosis	NextAppt
16577623	12/4/2018	9:00 AM	Vomiting	Stomach Virus	12/15/2018
16577660	1/28/2019	9:30 AM	Irritated Eye	Pink-Eye	-
16577685	12/5/2018	8:00 AM	Large Zit	MRSA	1/2/2019
16577615	12/18/2018	10:00 AM	Chills	Flu	12/28/2018
16577620	12/12/2018	1:30 PM	Headache	Headache	-
16577655	1/8/2019	12:00 PM	Fever	Sinus Infection	1/15/2019
16577690	1/9/2019	3:00 PM	Rash	Poison-Ivy	-
16577685	1/2/2019	8:30 AM	Follow-up	MRSA	2/15/2019
16577635	1/20/2019	12:30 PM	Abdominal Pain	Constipation	-
16577661	1/3/2019	1:00 PM	Lower Abdominal Pain	Urinary Tract Infection	-
16577645	12/19/2018	11:15 PM	Sore Throat	Streptococcal Pharyngitis	-

7

7



POPULATION INFORMATICS

Hand out: join patients and visits table to get name

PatientName	PatientID	Date	Time	Reason	Diagnosis	NextAppt
Allison Bellame	16577623	12/4/2018	9:00 AM	Vomiting	Stomach Virus	12/15/2018
Karsen Dixon	16577660	1/28/2019	9:30 AM	Irritated Eye	Pink-Eye	-
Jessica Lange	16577685	12/5/2018	8:00 AM	Large Zit	MRSA	1/2/2019
Michael Blane	16577615	12/18/2018	10:00 AM	Chills	Flu	12/28/2018
Scott Peters	16577620	12/12/2018	1:30 PM	Headache	Headache	-
Eric Sanders	16577655	1/8/2019	12:00 PM	Fever	Sinus Infection	1/15/2019
Brittany Greene	16577690	1/9/2019	3:00 PM	Rash	Poison-Ivy	-
Jessica Lange	16577685	1/2/2019	8:30 AM	Follow-up	MRSA	2/15/2019
Courtney Jones	16577635	1/20/2019	12:30 PM	Abdominal Pain	Constipation	-
Jane Doe	16577661	1/3/2019	1:00 PM	Lower Abdominal Pain	Urinary Tract Infection	-
Ashley Martin	16577645	12/19/2018	11:15 PM	Sore Throat	Streptococcal Pharyngitis	-

- SELECT patients.PatientName, visits.*
- FROM patients, visits
- WHERE patients.PatientID= visits.PatientID;

8

8

Hand out: join patients and visits table to get name

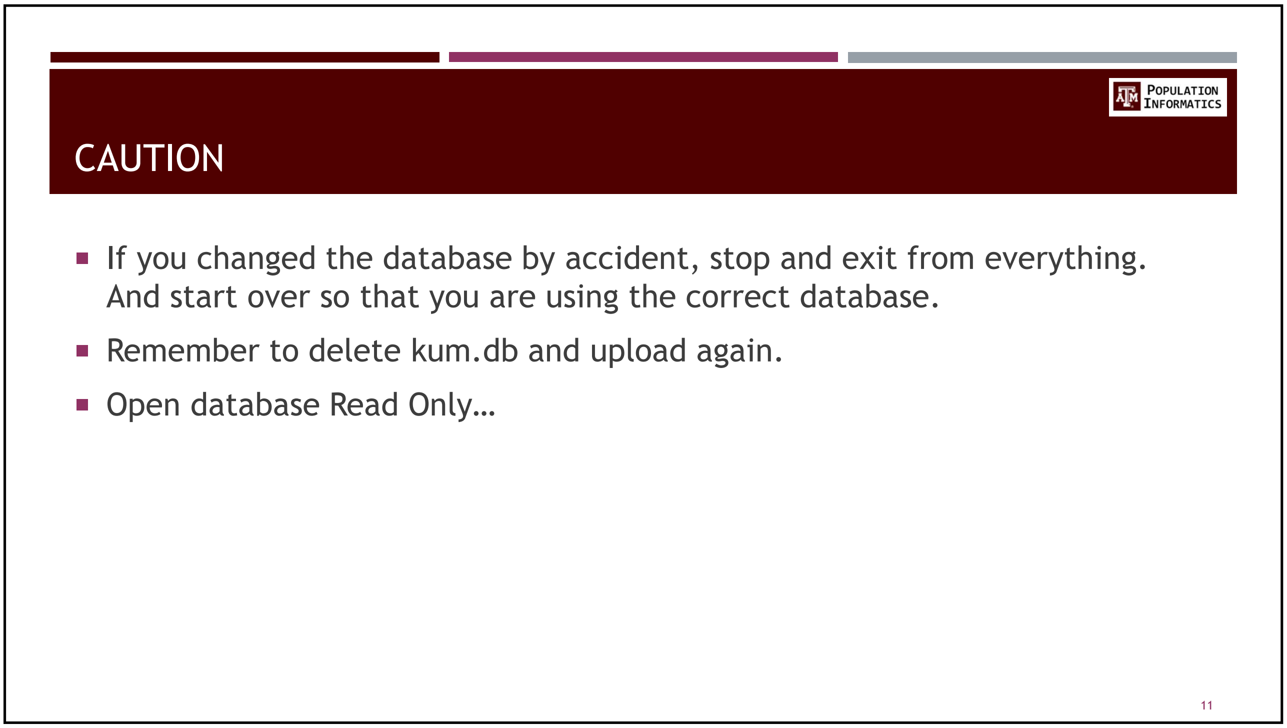
PatientName	PatientID	Date	Time	Reason	Diagnosis	NextAppt
Allison Bellame	16577623	12/4/2018	9:00 AM	Vomiting	Stomach Virus	12/15/2018

- SELECT patients.PatientName, visits.*
- FROM patients, visits
- WHERE patients.PatientID= visits.PatientID and Reason=“Vomiting”

9

Using SQL software and loading kum db?

10

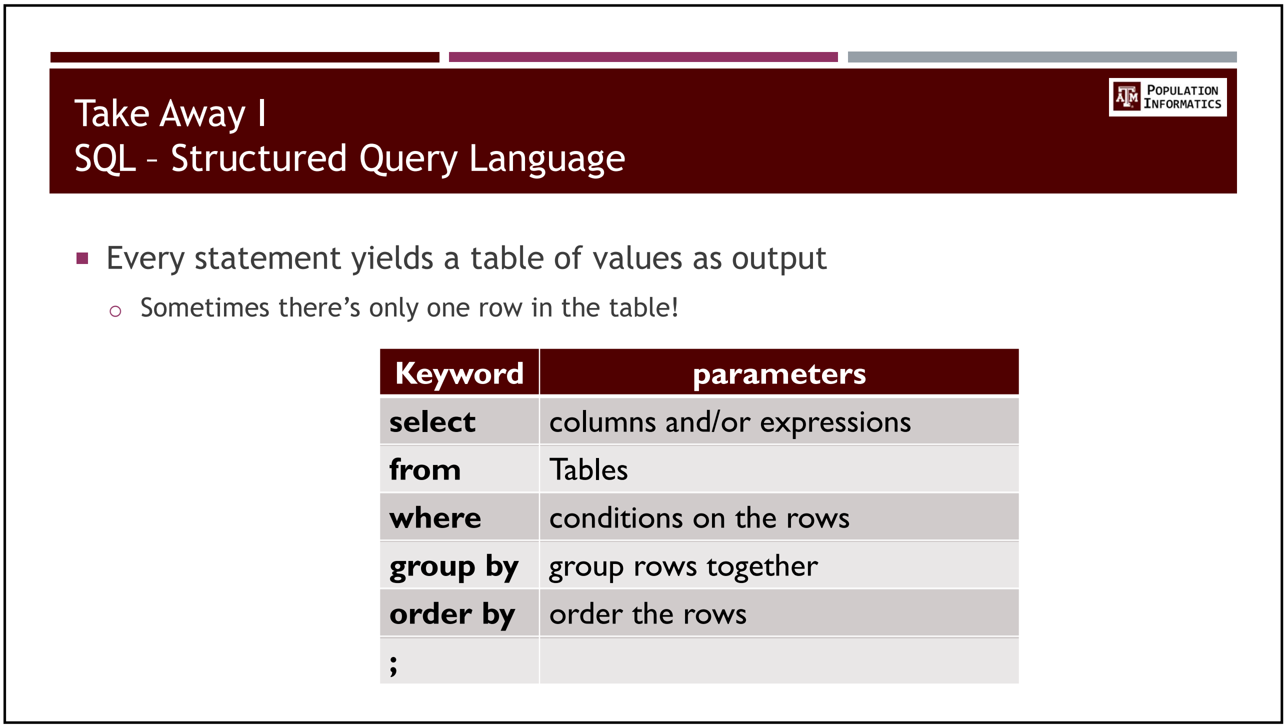


CAUTION

- If you changed the database by accident, stop and exit from everything. And start over so that you are using the correct database.
- Remember to delete kum.db and upload again.
- Open database Read Only…

11

11

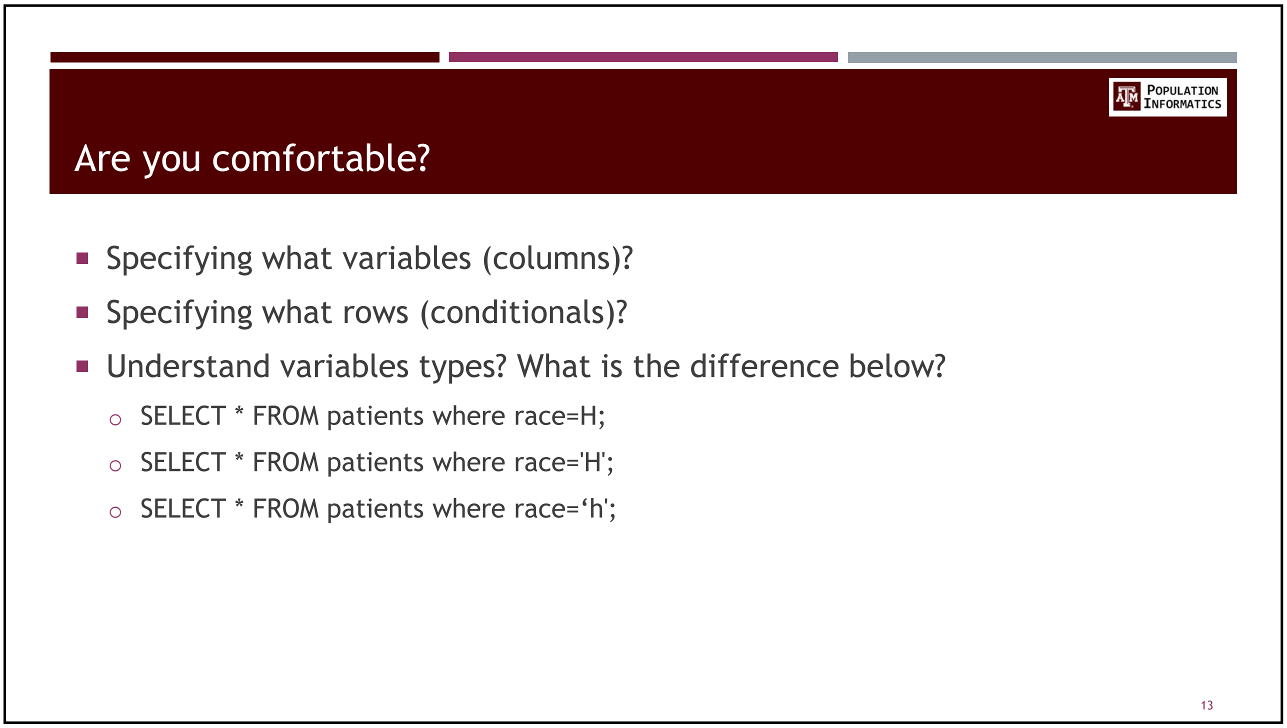


Take Away I
SQL – Structured Query Language

- Every statement yields a table of values as output
  - Sometimes there's only one row in the table!

Keyword	parameters
select	columns and/or expressions
from	Tables
where	conditions on the rows
group by	group rows together
order by	order the rows
;	

12

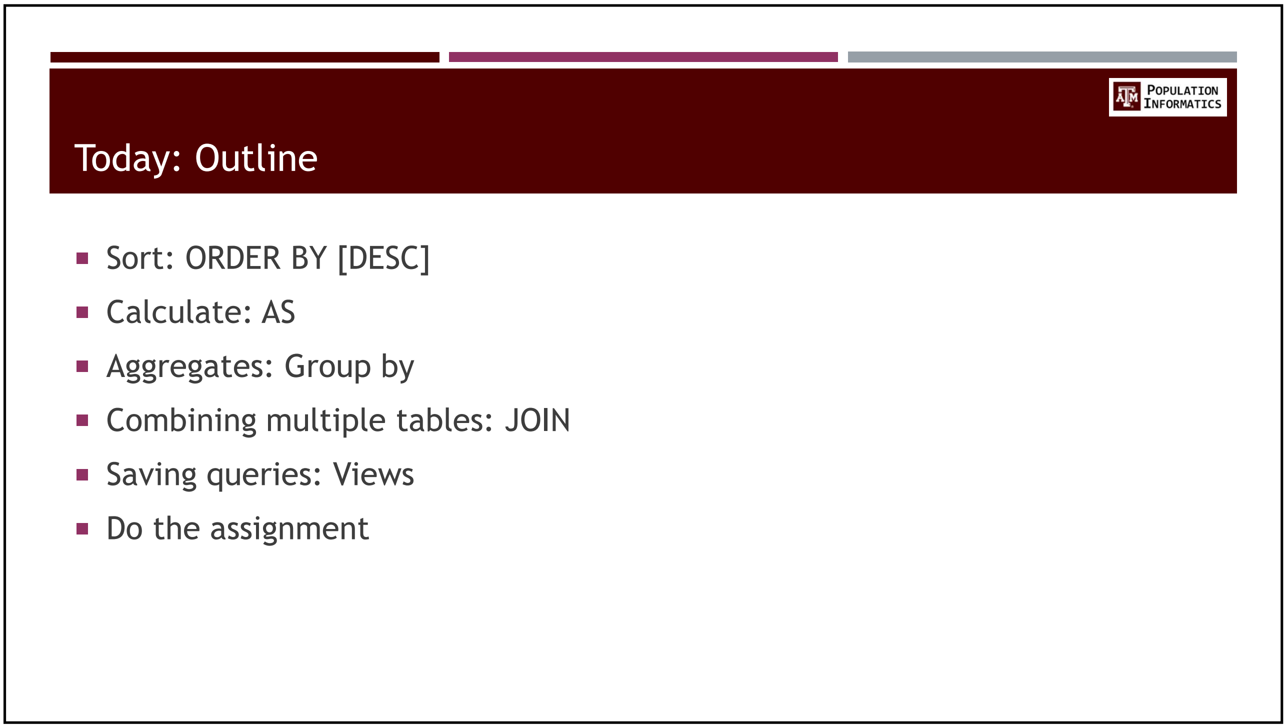


Are you comfortable?

- Specifying what variables (columns)?
- Specifying what rows (conditionals)?
- Understand variables types? What is the difference below?
  - SELECT * FROM patients where race=H;
  - SELECT * FROM patients where race='H';
  - SELECT * FROM patients where race=‘h';

13

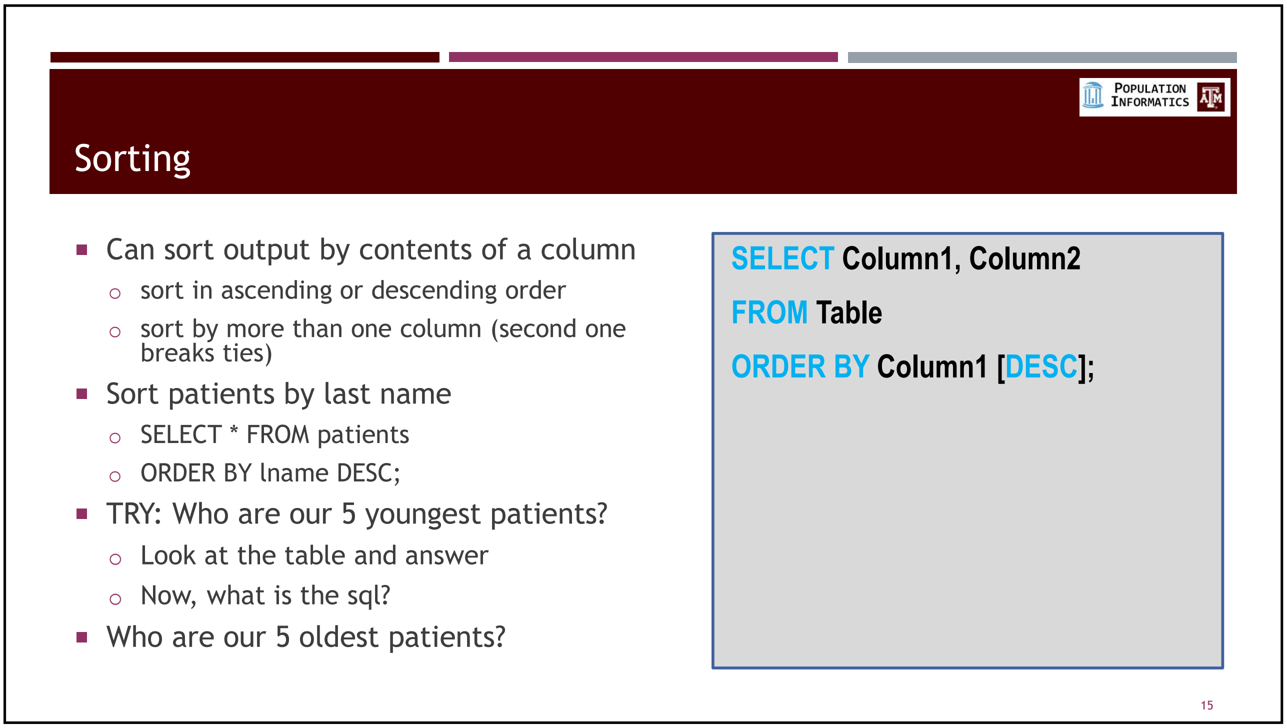
13



Today: Outline

- Sort: ORDER BY [DESC]
- Calculate: AS
- Aggregates: Group by
- Combining multiple tables: JOIN
- Saving queries: Views
- Do the assignment

14



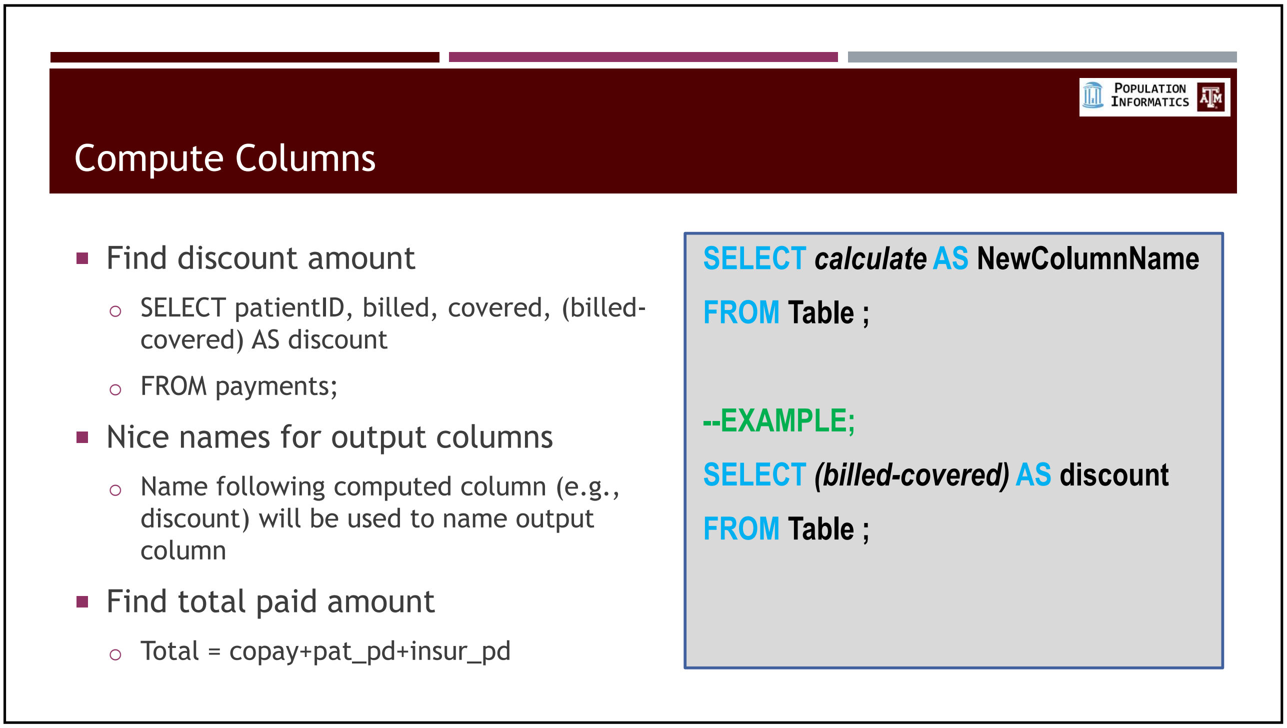
Sorting

- Can sort output by contents of a column
  - sort in ascending or descending order
  - sort by more than one column (second one breaks ties)
- Sort patients by last name
  - SELECT * FROM patients
  - ORDER BY lname DESC;
- TRY: Who are our 5 youngest patients?
  - Look at the table and answer
  - Now, what is the sql?
- Who are our 5 oldest patients?

```
SELECT Column1, Column2
FROM Table
ORDER BY Column1 [DESC];
```

15

15



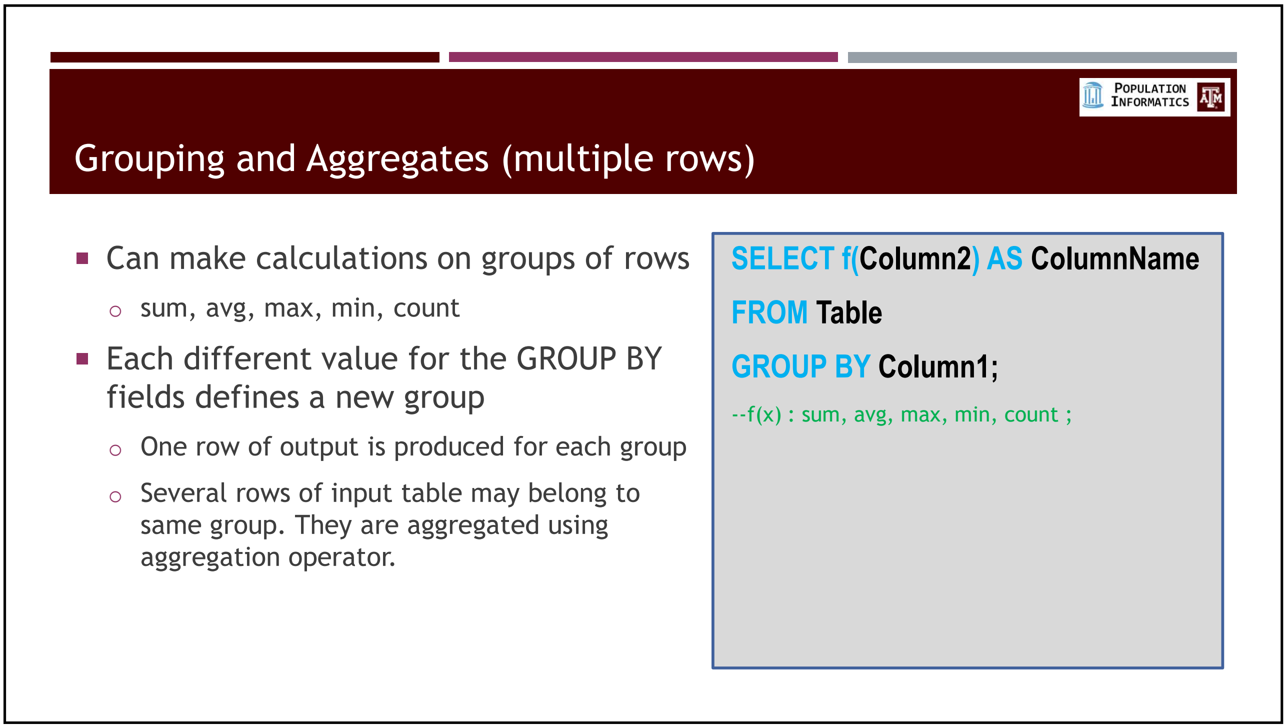
Compute Columns

- Find discount amount
  - SELECT patientID, billed, covered, (billed-covered) AS discount
  - FROM payments;
- Nice names for output columns
  - Name following computed column (e.g., discount) will be used to name output column
- Find total paid amount
  - Total = copay+pat_pd+insur_pd

```
SELECT calculate AS NewColumnName
FROM Table ;

--EXAMPLE;
SELECT (billed-covered) AS discount
FROM Table ;
```

16

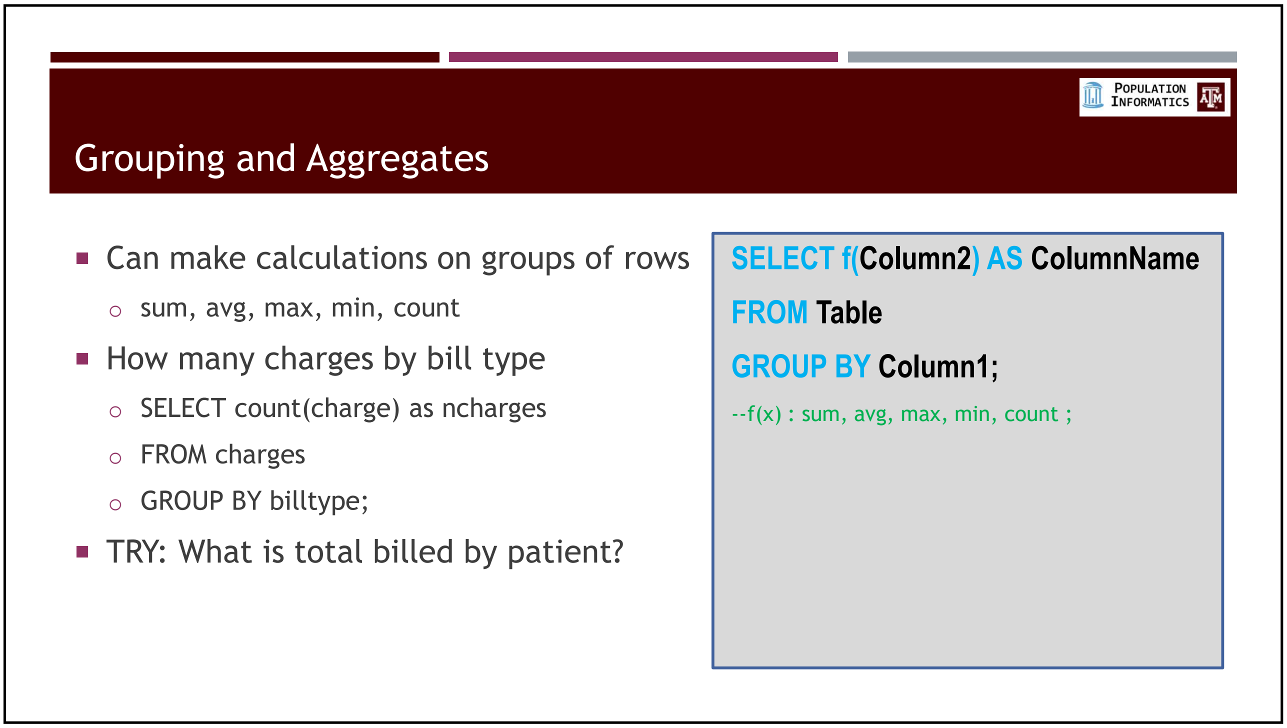


Grouping and Aggregates (multiple rows)

- Can make calculations on groups of rows
  - sum, avg, max, min, count
- Each different value for the GROUP BY fields defines a new group
  - One row of output is produced for each group
  - Several rows of input table may belong to same group. They are aggregated using aggregation operator.

```
SELECT f(Column2) AS ColumnName
FROM Table
GROUP BY Column1;
--f(x) : sum, avg, max, min, count ;
```

17



Grouping and Aggregates

- Can make calculations on groups of rows
  - sum, avg, max, min, count
- How many charges by bill type
  - SELECT count(charge) as ncharges
  - FROM charges
  - GROUP BY billtype;
- TRY: What is total billed by patient?

```
SELECT f(Column2) AS ColumnName
FROM Table
GROUP BY Column1;
--f(x) : sum, avg, max, min, count ;
```

18

POPULATION INFORMATICS

Table Operations

- Aggregate columns: col1 op col2 AS col3

col1	col2
a	d
b	e
c	f

→

col1	col2	col3
a	d	a+d
b	e	b+e
c	f	c+f

- Aggregate rows: Group BY

A
B
C

→

D

Where D=function(A,B,C)
Examples of function are
Sum(A,B,C) Avg(A,B,C) Max(A,B,C) Min(A,B,C) Count(A,B,C)

19

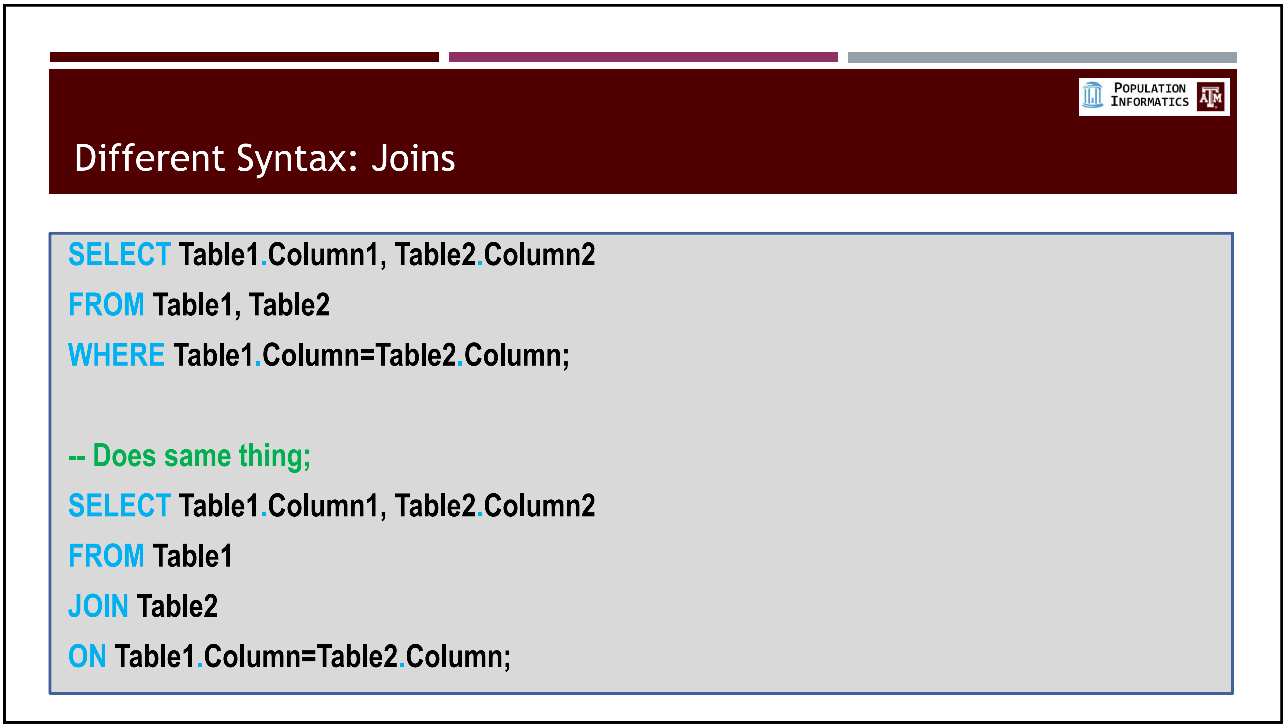
POPULATION INFORMATICS

Joins

- Combine rows from one table with rows from another
- Usually join on some common column
  - Don't combine rows unless their value in the common column is the same
  - WHERE clause says the common column must be same in each table
- Produce a list of bills for all patients with their name
  - SELECT patients.fname, patients.lname, patients.patientID, bill
  - from patients, payments
  - where patients.patientID=payments.patientID;

```
SELECT Table1.Column1, Table2.Column2
FROM Table1, Table2
WHERE Table1.Column=Table2.Column;
```

20

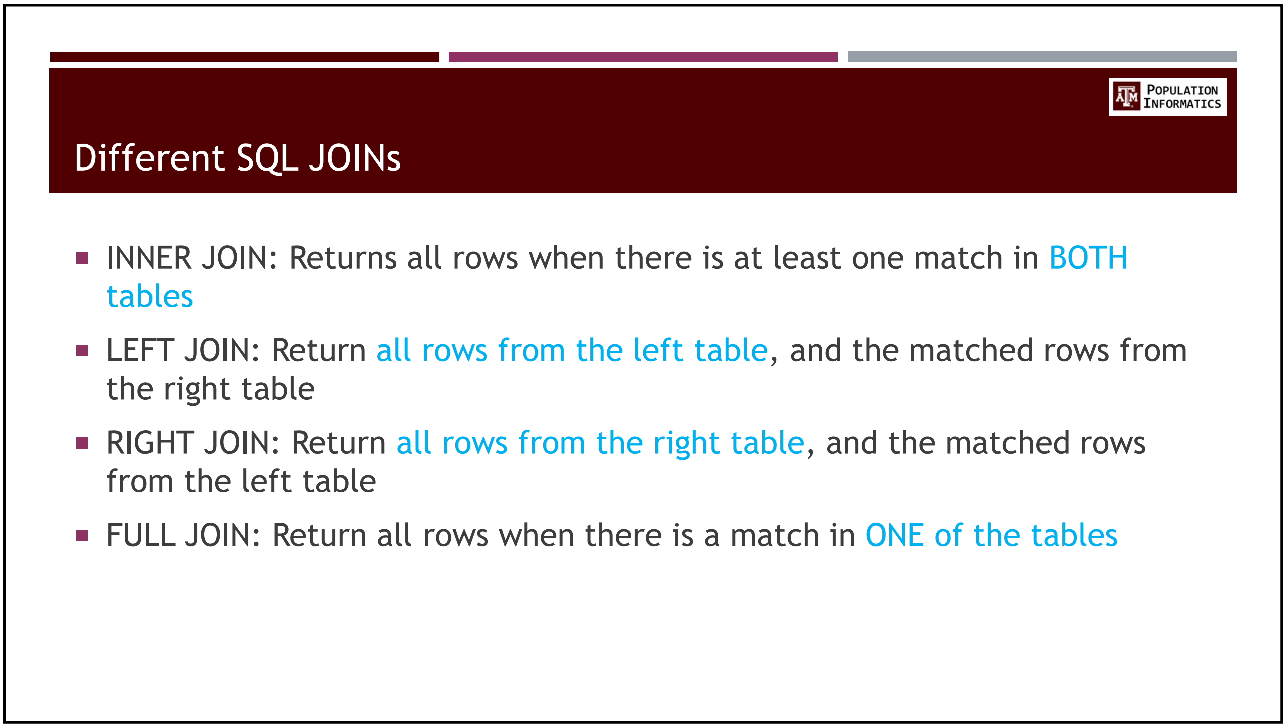


Different Syntax: Joins

```
SELECT Table1.Column1, Table2.Column2
FROM Table1, Table2
WHERE Table1.Column=Table2.Column;

-- Does same thing;
SELECT Table1.Column1, Table2.Column2
FROM Table1
JOIN Table2
ON Table1.Column=Table2.Column;
```

21



Different SQL JOINs

- INNER JOIN: Returns all rows when there is at least one match in BOTH tables
- LEFT JOIN: Return all rows from the left table, and the matched rows from the right table
- RIGHT JOIN: Return all rows from the right table, and the matched rows from the left table
- FULL JOIN: Return all rows when there is a match in ONE of the tables

22

Views: Permanent Queries

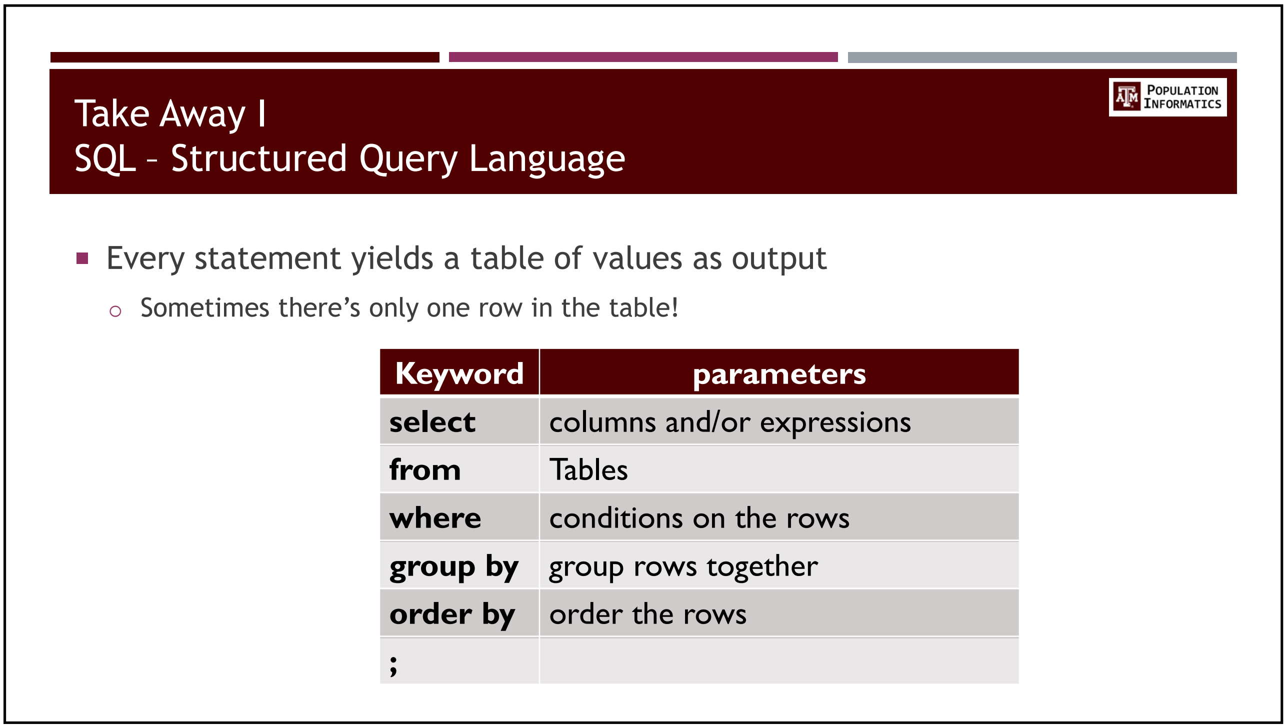
- Looks and feels like a table
- Saved queries
- Virtual table: not a real table in the DB
- Can treat it like a real table, as if it exists

23

Create View

```
CREATE VIEW panel AS
SELECT
    providers.fname as dr_first,    providers.lname as dr_last,
    patients.fname, patients.lname
FROM providers, patients
WHERE providers.providerID=patients.primary_dr;
```

24



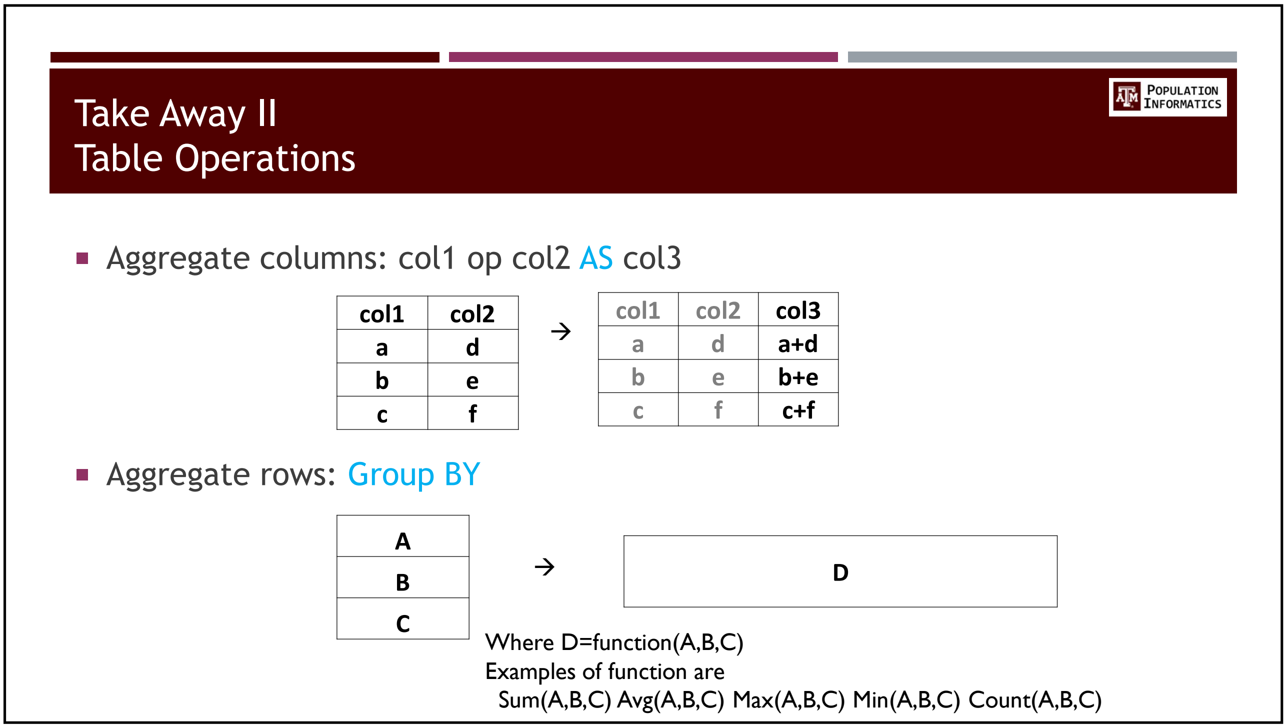
Take Away I
SQL – Structured Query Language

POPULATION INFORMATICS

- Every statement yields a table of values as output
  - Sometimes there's only one row in the table!

Keyword	parameters
select	columns and/or expressions
from	Tables
where	conditions on the rows
group by	group rows together
order by	order the rows
;	

25



Take Away II
Table Operations

POPULATION INFORMATICS

- Aggregate columns: col1 op col2 AS col3

col1	col2
a	d
b	e
c	f

→

col1	col2	col3
a	d	a+d
b	e	b+e
c	f	c+f

- Aggregate rows: Group BY

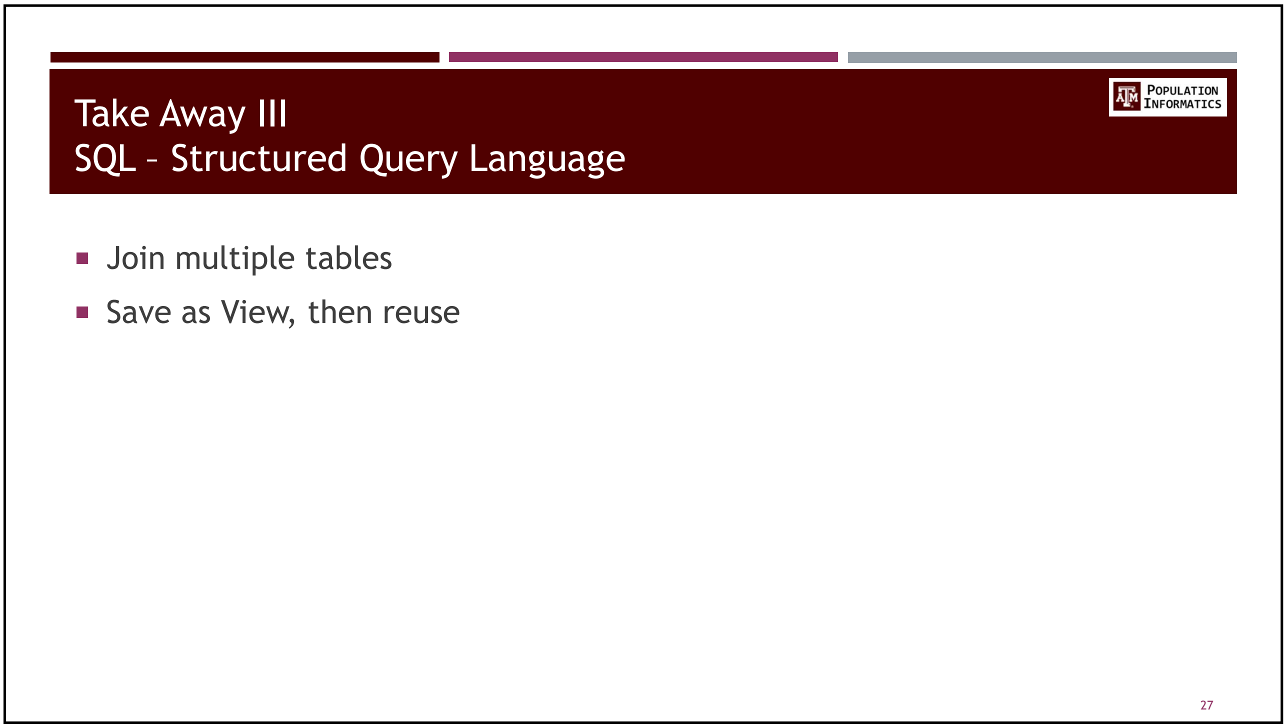
A
B
C

→

D

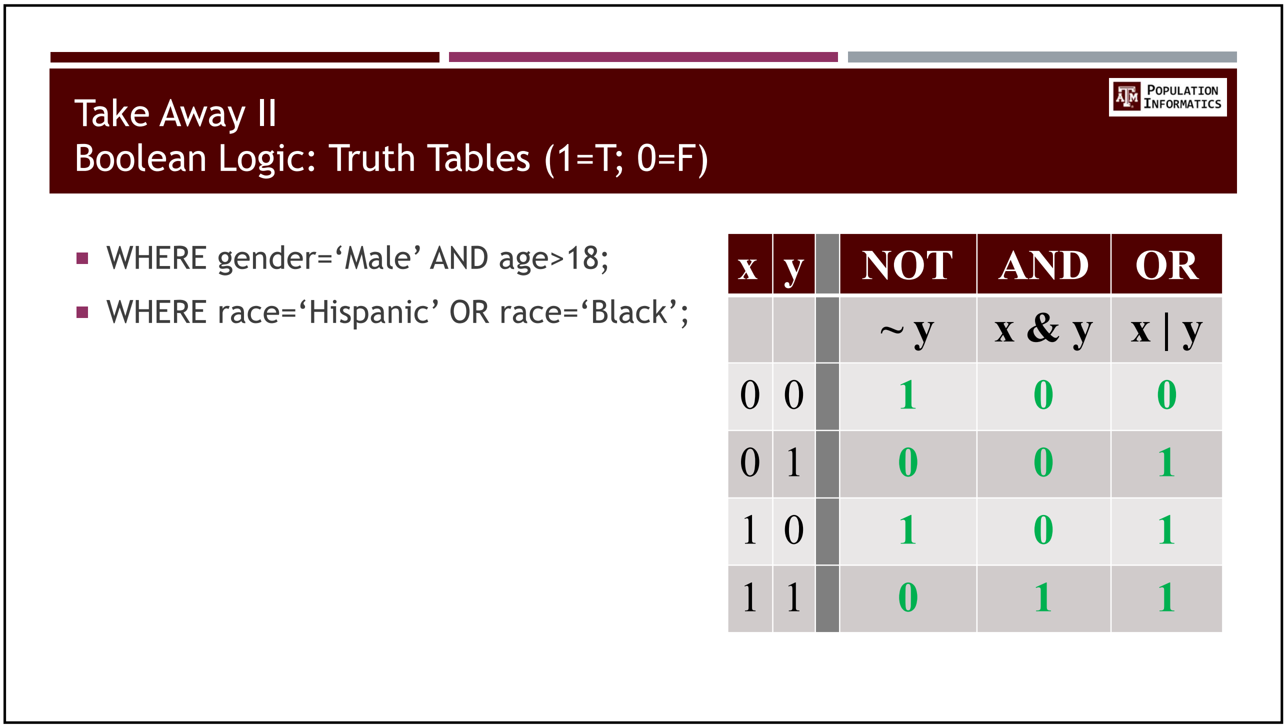
Where D=function(A,B,C)
Examples of function are
Sum(A,B,C) Avg(A,B,C) Max(A,B,C) Min(A,B,C) Count(A,B,C)

26



Take Away III
SQL – Structured Query Language

- Join multiple tables
- Save as View, then reuse



Take Away II
Boolean Logic: Truth Tables (1=T; 0=F)

- WHERE gender=‘Male’ AND age>18;
- WHERE race=‘Hispanic’ OR race=‘Black’;

x	y	NOT	AND	OR
		~ y	x & y	x | y
0	0	1	0	0
0	1	0	0	1
1	0	1	0	1
1	1	0	1	1

Take Away III
Variable Types

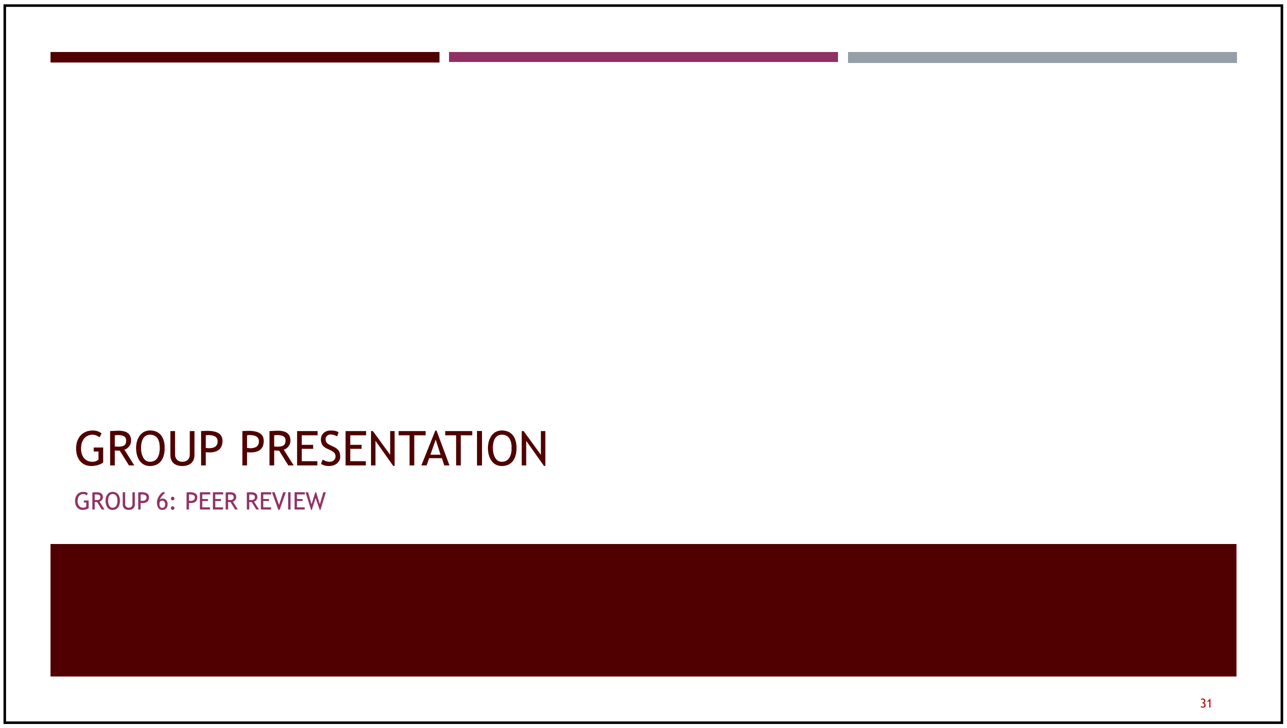
Type	Stored value	Interpreted value	Label Interpreted Value
int	1000001 (65)	65	65 or older
Char/string (ASCII)	1000001 (65)	A	Asian
date	1000001 (65)	1960/3/6 (SAS)	

- `1  0  0  0  0  0 1 =64+1=65`
- `64 32 16  8  4  2 1`
- Understand variables types? What is the difference below?
  - SELECT * FROM patients where race=H;
  - SELECT * FROM patients where race='H';
  - SELECT * FROM patients where race=‘h';

29

Raise your hand when you are back

30

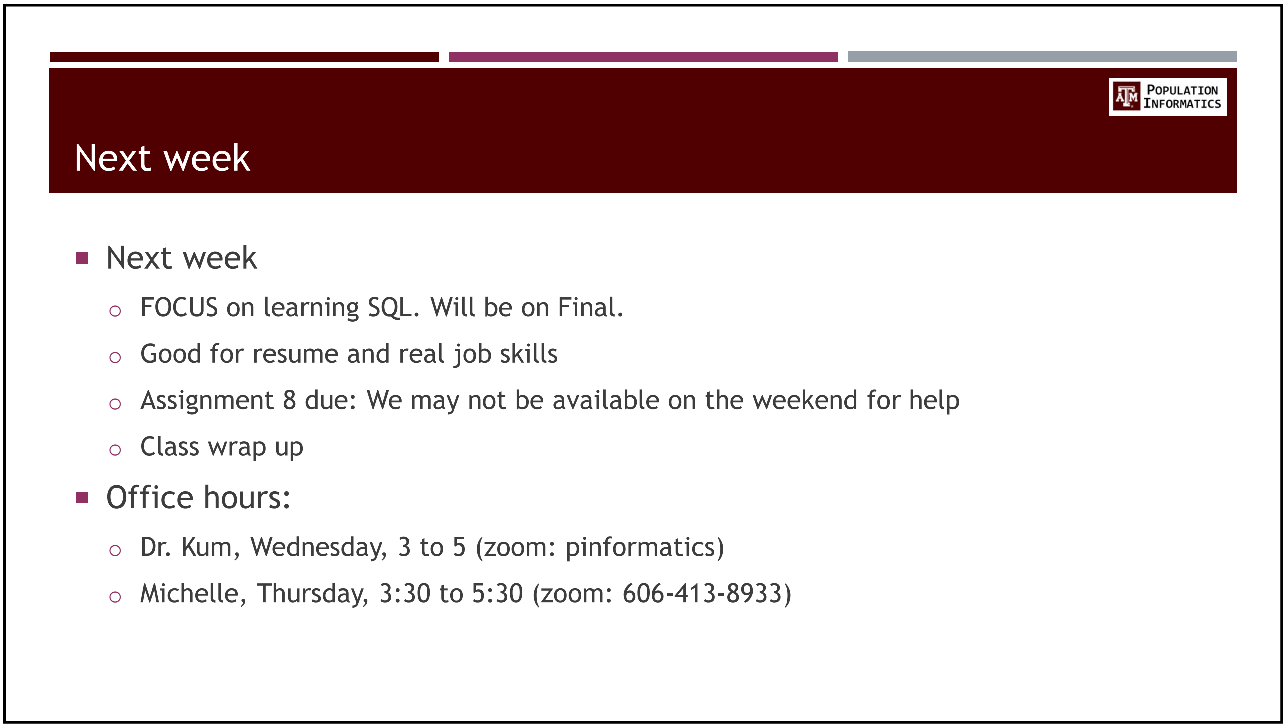


GROUP PRESENTATION

GROUP 6: PEER REVIEW

31

31



POPULATION INFORMATICS

Next week

- Next week
  - FOCUS on learning SQL. Will be on Final.
  - Good for resume and real job skills
  - Assignment 8 due: We may not be available on the weekend for help
  - Class wrap up
- Office hours:
  - Dr. Kum, Wednesday, 3 to 5 (zoom: pinformatics)
  - Michelle, Thursday, 3:30 to 5:30 (zoom: 606-413-8933)

32