// You need to have completed the "Loading Data" Labs prior to running

// this lab. This lab also uses the "redshift-query-benchmark.xlsx"

// spreadsheet… open it

// SELECT TOO MUCH DATA

select \* from lineorder; // too big!

// CREATE TABLE AS SELECT

create table customer\_asai as select \* from customer where c\_region = "ASIA";

// BLOWS UP, DOUBLE QUOTES

create table customer\_asai as select \* from customer where c\_region = 'ASIA';

// OOPS… NAMED IT WRONG… EASY FIX

alter table customer\_asai rename to customer\_asia;

select count(\*) from customer;

select count(\*) from customer\_asia;

// SHOW TABLES

select distinct(tablename) from pg\_table\_def where schemaname = 'public';

// BETWEEN CLAUSE (ORDERS ON DEC 1st, 1994)

select \* from lineorder where lo\_orderdate between 19941201 and 19941202;

// CAREFUL WITH PULLING LOTS OF DATA DIRECTLY OUT OF REDSHIFT

select count(\*) from lineorder where lo\_orderdate between 19941201 and 19941202;

// NOW LET'S LOOK AT THE IMPACT OF OUR SORT KEYS, DISTRIBUTION KEYS, AND ENCODING TYPES

// MUCH MORE INFO ON QUERY OPTIMIZATION AVAILABLE AT:

// http://docs.aws.amazon.com/redshift/latest/dg/c-query-tuning.html

// http://docs.aws.amazon.com/redshift/latest/dg/tutorial-tuning-tables-test-performance.html

// RUN THE FOLLOWING QUERIES TWICE… THE FIRST TIME, THEY COMPILE, AND THE SECOND TIME

// IS THEIR RAW RUN TIME. IMPORTANT TO RECORD THE SPEED OF THE QUERY THE SECOND TIME IT RUNS

// NOTE: NONE OF THE TABLES HAVE ANY SORT KEYS ON THEM AT ALL

// Q1: RESTRICTION ON ONE DIMENSION

select sum(lo\_extendedprice\*lo\_discount) as revenue

from lineorder, dwdate

where lo\_orderdate = d\_datekey

and d\_year = 1997

and lo\_discount between 1 and 3

and lo\_quantity < 24;

// Q2: RESTRICTION ON TWO DIMENSIONS

select sum(lo\_revenue), d\_year, p\_brand1

from lineorder, dwdate, part, supplier

where lo\_orderdate = d\_datekey

and lo\_partkey = p\_partkey

and lo\_suppkey = s\_suppkey

and p\_category = 'MFGR#12'

and s\_region = 'AMERICA'

group by d\_year, p\_brand1

order by d\_year, p\_brand1;

// Q3: DRILL DOWN TO JUST ONE MONTH'S WORTH OF DATA IN JUST ONCE CITY

select c\_city, s\_city, d\_year, sum(lo\_revenue) as revenue

from customer, lineorder, supplier, dwdate

where lo\_custkey = c\_custkey

and lo\_suppkey = s\_suppkey

and lo\_orderdate = d\_datekey

and (c\_city='UNITED KI1' or

c\_city='UNITED KI5')

and (s\_city='UNITED KI1' or

s\_city='UNITED KI5')

and d\_yearmonth = 'Dec1997'

group by c\_city, s\_city, d\_year

order by d\_year asc, revenue desc;

// WHAT WOULD MAKE GOOD SORT KEYS BASED ON THOSE 3 QUERIES?

// UPDATE YOUR SELECTIONS FOR SORT KEY IN THE EXCEL WORKSHEET.

// WHAT WOULD MAKE GOOD DISTRIBUTION KEYS BASED ON THOSE 3 QUERIES?

// UPDATE YOUR SELECTIONS FOR SORT KEY IN THE EXCEL WORKSHEET.

// LETS GO AHEAD & DROP ALL EXISTING TABLES

drop table part cascade;

drop table supplier cascade;

drop table customer cascade;

drop table dwdate cascade;

drop table lineorder cascade;

// RE-CREATE TABLES WITH SPECIFIED DISTKEY AND SORTKEY

CREATE TABLE part (

p\_partkey integer not null sortkey distkey,

p\_name varchar(22) not null,

p\_mfgr varchar(6) not null,

p\_category varchar(7) not null,

p\_brand1 varchar(9) not null,

p\_color varchar(11) not null,

p\_type varchar(25) not null,

p\_size integer not null,

p\_container varchar(10) not null

);

CREATE TABLE supplier (

s\_suppkey integer not null sortkey,

s\_name varchar(25) not null,

s\_address varchar(25) not null,

s\_city varchar(10) not null,

s\_nation varchar(15) not null,

s\_region varchar(12) not null,

s\_phone varchar(15) not null)

diststyle all;

CREATE TABLE customer (

c\_custkey integer not null sortkey,

c\_name varchar(25) not null,

c\_address varchar(25) not null,

c\_city varchar(10) not null,

c\_nation varchar(15) not null,

c\_region varchar(12) not null,

c\_phone varchar(15) not null,

c\_mktsegment varchar(10) not null)

diststyle all;

CREATE TABLE dwdate (

d\_datekey integer not null sortkey,

d\_date varchar(19) not null,

d\_dayofweek varchar(10) not null,

d\_month varchar(10) not null,

d\_year integer not null,

d\_yearmonthnum integer not null,

d\_yearmonth varchar(8) not null,

d\_daynuminweek integer not null,

d\_daynuminmonth integer not null,

d\_daynuminyear integer not null,

d\_monthnuminyear integer not null,

d\_weeknuminyear integer not null,

d\_sellingseason varchar(13) not null,

d\_lastdayinweekfl varchar(1) not null,

d\_lastdayinmonthfl varchar(1) not null,

d\_holidayfl varchar(1) not null,

d\_weekdayfl varchar(1) not null)

diststyle all;

CREATE TABLE lineorder (

lo\_orderkey integer not null,

lo\_linenumber integer not null,

lo\_custkey integer not null,

lo\_partkey integer not null distkey,

lo\_suppkey integer not null,

lo\_orderdate integer not null sortkey,

lo\_orderpriority varchar(15) not null,

lo\_shippriority varchar(1) not null,

lo\_quantity integer not null,

lo\_extendedprice integer not null,

lo\_ordertotalprice integer not null,

lo\_discount integer not null,

lo\_revenue integer not null,

lo\_supplycost integer not null,

lo\_tax integer not null,

lo\_commitdate integer not null,

lo\_shipmode varchar(10) not null

);

// COPY & PASTE THE "COPY" STATEMENTS FROM YOUR 'data-loading-lab-instructions.txt' file & then remove the 'compupdate off' parameter

copy dwdate from 's3://awssampledbuswest2/ssbgz/dwdate'

credentials 'aws\_access\_key\_id=AKIAITDF6KWO5JNU4IHQ;aws\_secret\_access\_key=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'

gzip

copy part from 's3://awssampledbuswest2/ssbgz/part'

credentials 'aws\_access\_key\_id=AKIAITDF6KWO5JNU4IHQ;aws\_secret\_access\_key=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'

gzip

copy supplier from 's3://awssampledbuswest2/ssbgz/supplier'

credentials 'aws\_access\_key\_id=AKIAITDF6KWO5JNU4IHQ;aws\_secret\_access\_key=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'

gzip

copy customer from 's3://awssampledbuswest2/ssbgz/customer'

credentials 'aws\_access\_key\_id=AKIAITDF6KWO5JNU4IHQ;aws\_secret\_access\_key=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'

gzip

copy lineorder from 's3://awssampledbuswest2/ssbgz/lineorder'

credentials 'aws\_access\_key\_id=AKIAITDF6KWO5JNU4IHQ;aws\_secret\_access\_key=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'

gzip

// RE-RUN ALL OF THOSE COPY STATEMENTS AS A BULK OPERATION INTO SQL WORKBENCH (SHOULD TAKE 10 - 15 MINS)

// RE-RUN THE QUERIES ABOVE TO CHECK THEIR PERFORMANCE