Hands-on Lab: Setting up a staging area

In this lab you will:

- Setup a staging server for a data warehouse
- Create the schema to store the data
- Load the data into the tables
- Run a sample query

Exercise 1 - Start the PostgreSQL server

We will be using the PostgreSQL server as our staging server.

Start the PostgreSQL server.

Open a new terminal, by clicking on the menu bar and selecting Terminal->New Terminal.

This will open a new terminal at the bottom of the screen.

Run the commands below on the newly opened terminal. (You can copy the code by clicking on the little copy button on the bottom right of the codeblock below and then paste it, wherever you wish.)

Start the PostgreSQL server, by running the command below:

```
theia@theiadocker-craigtrupp8:/home/project$ start_postgres
Unable to find image 'ubuntu:latest' locally
latest: Pulling from library/ubuntu
37aaf24cf781: Pull complete
Digest:
sha256:9b8dec3bf938bc80fbe758d856e96fdfab5f56c39d44b0cff351e847bb1b01ea
Status: Downloaded newer image for ubuntu:latest
Starting your Postgres database....
This process can take up to a minute.

Postgres database started, waiting for all services to be ready....

Your Postgres database is now ready to use and available with username:
postgres password: MjE1NDQtY3JhaWd0

You can access your Postgres database via:
```

```
  • The Browser with pgadmin
    • URL:
https://craigtrupp8-5050.theiadocker-3-labs-prod-theiak8s-4-tor01.proxy.cog
nitiveclass.ai/browser/
    • Database Password: MjE1NDQtY3JhaWd0
  • CommandLine: psql --username=postgres --host=localhost
```

Exercise 2 - Create Database

Create the database on the data warehouse.

Using the createdb command of the PostgreSQL server, we can directly create the database from the terminal.

Run the command below to create a database named billingDW

```
createdb -h localhost -U postgres -p 5432 billingDW
```

In the above command

-h mentions that the database server is running on the localhost
-U mentions that we are using the user name postgres to log into the database
-p mentions that the database server is running on port number 543

Exercise 3 - Create data warehouse schema

- Download and extract schema files (from tar file)

```
wget
https://cf-courses-data.s3.us.cloud-object-storage.appdomain.cloud/IBM-DB02
60EN-SkillsNetwork/labs/Setting%20up%20a%20staging%20area/billing-datawareh
ouse.tgz

tar -xvzf billing-datawarehouse.tgz
ls *.sql
```

```
theia@theiadocker-craigtrupp8:/home/project$ tar -xvzf
billing-datawarehouse.tgz
DimCustomer.sql
DimMonth.sql
FactBilling.sql
star-schema.sql
verify.sql
theia@theiadocker-craigtrupp8:/home/project$ ls *.sql
DimCustomer.sql  DimMonth.sql  FactBilling.sql  star-schema.sql  verify.sql
```

(Individual look at SQL/Schema Files)

- Star Schema

```
BEGIN;


CREATE TABLE public."FactBilling"
(
    billid serial,
    customerid integer NOT NULL,
    monthid integer NOT NULL,
    billedamount integer NOT NULL,
    PRIMARY KEY (billid)
);

CREATE TABLE public."DimMonth"
(
    monthid integer NOT NULL,
    year integer NOT NULL,
    month integer NOT NULL,
    monthname varchar(10) NOT NULL,
    quarter integer NOT NULL,
    quartername varchar(2) NOT NULL,
    PRIMARY KEY (monthid)
);

CREATE TABLE public."DimCustomer"
(
    customerid integer NOT NULL,
    category varchar(10) NOT NULL,
    country varchar(40) NOT NULL,
```

```
    industry varchar(40) NOT NULL,
    PRIMARY KEY (customerid)
);

ALTER TABLE public."FactBilling"
    ADD FOREIGN KEY (customerid)
    REFERENCES public."DimCustomer" (customerid)
    NOT VALID;


ALTER TABLE public."FactBilling"
    ADD FOREIGN KEY (monthid)
    REFERENCES public."DimMonth" (monthid)
    NOT VALID;

END;
```

- (Fact Table Insert Script - Partial - Shell for total script count)

```
theia@theiadocker-craigtrupp8:/home/project$ ls
billing-datawarehouse.tgz  DimCustomer.sql  DimMonth.sql  FactBilling.sql
postgres  star-schema.sql  verify.sql
theia@theiadocker-craigtrupp8:/home/project$ cat FactBilling.sql | wc -l
132002
theia@theiadocker-craigtrupp8:/home/project$ cat FactBilling.sql | head -20
INSERT INTO "FactBilling"
(billid,customerid,billedamount,monthid)
VALUES
(1,1,5060,20091),
(2,614,9638,20091),
(3,615,11573,20091),
(4,616,18697,20091),
(5,617,944,20091),
(6,618,3539,20091),
(7,41,6591,20091),
(8,619,16061,20091),
(9,620,1250,20091),
(10,956,15105,20091),
(11,621,6644,20091),
(12,622,2409,20091),
(13,623,5792,20091),
```

```
(14,624,11626,20091),
(15,625,11776,20091),
(16,626,14722,20091),
(17,40,14962,20091),
```

- Dimension Tables

```
INSERT INTO "DimCustomer"(customerid,category,country,industry)
VALUES
(1,'Individual','Indonesia','Engineering'),
(614,'Individual','United States','Product Management'),
(615,'Individual','China','Services'),
(616,'Individual','Russia','Accounting'),
(617,'Individual','Chile','Business Development'),
(618,'Individual','Nicaragua','Human Resources'),
(41,'Company','Brazil','Marketing'),
(619,'Individual','Russia','Business Development'),
(620,'Individual','China','Business Development'),
(956,'Individual','Peru','Research and Development'),
(621,'Individual','Angola','Services'),
(622,'Individual','Poland','Legal'),
(623,'Individual','Italy','Training'),
(624,'Individual','Indonesia','Sales'),
(625,'Individual','Portugal','Services'),
(626,'Individual','Mexico','Engineering'),
(40,'Individual','Portugal','Human Resources'),
(627,'Company','Philippines','Services'),
(628,'Individual','France','Business Development'),
(629,'Company','France','Training'),
(630,'Individual','Indonesia','Services'),
(631,'Company','Poland','Marketing'),
(632,'Company','Brunei','Training'),
(633,'Individual','Russia','Sales'),
(634,'Company','Mexico','Business Development'),
(39,'Company','Ivory Coast','Engineering'),
(613,'Individual','Czech Republic','Sales'),
(955,'Individual','Sudan','Accounting'),
(612,'Individual','Argentina','Product Management'),
(611,'Individual','Uzbekistan','Engineering'),
(591,'Individual','China','Support'),
(592,'Company','Philippines','Training'),
(593,'Company','China','Services'),
```

```
(958,'Company','Indonesia','Sales'),
(594,'Individual','Germany','Engineering'),
(595,'Company','Portugal','Legal'),
(44,'Individual','Paraguay','Human Resources'),
(596,'Company','Philippines','Human Resources'),
(597,'Individual','South Africa','Accounting'),
(598,'Company','United Kingdom','Training'),
(599,'Individual','Cyprus','Engineering'),
(600,'Individual','Sweden','Sales'),
(601,'Individual','Nigeria','Training'),
(602,'Individual','China','Human Resources'),
(603,'Individual','Iran','Engineering'),
(43,'Individual','Portugal','Marketing'),
(604,'Individual','Sweden','Human Resources'),
(998,'Individual','China','Engineering'),
(605,'Individual','Philippines','Engineering'),
(606,'Individual','Poland','Marketing'),
(607,'Individual','Philippines','Support'),
(957,'Company','Philippines','Legal'),
(608,'Company','Dominican Republic','Support'),
(609,'Individual','Indonesia','Training'),
(610,'Company','China','Legal'),
(42,'Individual','Poland','Marketing'),
(590,'Company','China','Human Resources'),
(635,'Individual','Czech Republic','Sales'),
(637,'Company','Madagascar','Support'),
(953,'Company','Vietnam','Business Development'),
(663,'Company','Cuba','Research and Development'),
(664,'Company','Poland','Sales'),
(665,'Company','France','Support'),
(35,'Individual','Republic of the Congo','Business Development'),
(666,'Individual','Japan','Legal'),
(667,'Individual','China','Training'),
(668,'Company','France','Services'),
(669,'Individual','Pakistan','Engineering'),
(670,'Individual','Brazil','Sales'),
(671,'Individual','Czech Republic','Support'),
(672,'Individual','Slovakia','Support'),
(673,'Company','China','Product Management'),
(674,'Company','Ukraine','Support'),
(34,'Company','Croatia','Research and Development'),
(675,'Company','France','Marketing'),
(952,'Individual','China','Sales'),
```

```
(676,'Individual','Lithuania','Accounting'),
(677,'Individual','Dominican Republic','Human Resources'),
(678,'Company','China','Product Management'),
(679,'Individual','Latvia','Support'),
(680,'Company','France','Services'),
(681,'Individual','Mauritius','Accounting'),
(33,'Individual','China','Business Development'),
(682,'Company','China','Sales'),
(662,'Individual','Nigeria','Training'),
(636,'Company','Tanzania','Legal'),
(661,'Company','China','Services'),
(659,'Individual','Poland','Engineering'),
(638,'Company','Colombia','Support'),
(639,'Individual','Finland','Legal'),
(640,'Company','Colombia','Support'),
(641,'Individual','Vietnam','Engineering'),
(642,'Company','China','Business Development'),
(38,'Individual','Philippines','Sales'),
(643,'Company','Egypt','Support'),
(644,'Individual','Slovenia','Legal'),
(645,'Individual','Indonesia','Human Resources'),
(646,'Individual','Portugal','Sales')
```

- Month Dimension

```
INSERT INTO "DimMonth"
(monthid,year,month,monthname,quarter,quartername)
VALUES
(20091, 2009, 1, 'Janauary', 1, 'Q1'),
(200910, 2009, 10, 'October', 4, 'Q4'),
(200911, 2009, 11, 'November', 4, 'Q4'),
(200912, 2009, 12, 'December', 4, 'Q4'),
(20092, 2009, 2, 'February', 1, 'Q1'),
(20093, 2009, 3, 'March', 1, 'Q1'),
(20094, 2009, 4, 'April', 2, 'Q2'),
(20095, 2009, 5, 'May', 2, 'Q2'),
(20096, 2009, 6, 'June', 2, 'Q2'),
(20097, 2009, 7, 'July', 3, 'Q3'),
(20098, 2009, 8, 'August', 3, 'Q3'),
(20099, 2009, 9, 'September', 3, 'Q3'),
(20101, 2010, 1, 'Janauary', 1, 'Q1'),
(201010, 2010, 10, 'October', 4, 'Q4'),
(201011, 2010, 11, 'November', 4, 'Q4'),
```

```
(201012, 2010, 12, 'December', 4, 'Q4'),
(20102, 2010, 2, 'February', 1, 'Q1'),
(20103, 2010, 3, 'March', 1, 'Q1'),
(20104, 2010, 4, 'April', 2, 'Q2'),
(20105, 2010, 5, 'May', 2, 'Q2'),
(20106, 2010, 6, 'June', 2, 'Q2'),
(20107, 2010, 7, 'July', 3, 'Q3'),
(20108, 2010, 8, 'August', 3, 'Q3'),
(20109, 2010, 9, 'September', 3, 'Q3'),
(20111, 2011, 1, 'Janauary', 1, 'Q1'),
(201110, 2011, 10, 'October', 4, 'Q4'),
(201111, 2011, 11, 'November', 4, 'Q4'),
(201112, 2011, 12, 'December', 4, 'Q4'),
(20112, 2011, 2, 'February', 1, 'Q1'),
(20113, 2011, 3, 'March', 1, 'Q1'),
(20114, 2011, 4, 'April', 2, 'Q2'),
(20115, 2011, 5, 'May', 2, 'Q2'),
(20116, 2011, 6, 'June', 2, 'Q2'),
(20117, 2011, 7, 'July', 3, 'Q3'),
(20118, 2011, 8, 'August', 3, 'Q3'),
(20119, 2011, 9, 'September', 3, 'Q3'),
(20121, 2012, 1, 'Janauary', 1, 'Q1'),
(201210, 2012, 10, 'October', 4, 'Q4'),
(201211, 2012, 11, 'November', 4, 'Q4'),
(201212, 2012, 12, 'December', 4, 'Q4'),
(20122, 2012, 2, 'February', 1, 'Q1'),
(20123, 2012, 3, 'March', 1, 'Q1'),
(20124, 2012, 4, 'April', 2, 'Q2'),
(20125, 2012, 5, 'May', 2, 'Q2'),
(20126, 2012, 6, 'June', 2, 'Q2'),
(20127, 2012, 7, 'July', 3, 'Q3'),
(20128, 2012, 8, 'August', 3, 'Q3'),
(20129, 2012, 9, 'September', 3, 'Q3'),
(20131, 2013, 1, 'Janauary', 1, 'Q1'),
(201310, 2013, 10, 'October', 4, 'Q4'),
(201311, 2013, 11, 'November', 4, 'Q4'),
(201312, 2013, 12, 'December', 4, 'Q4'),
(20132, 2013, 2, 'February', 1, 'Q1'),
(20133, 2013, 3, 'March', 1, 'Q1'),
(20134, 2013, 4, 'April', 2, 'Q2'),
(20135, 2013, 5, 'May', 2, 'Q2'),
(20136, 2013, 6, 'June', 2, 'Q2'),
(20137, 2013, 7, 'July', 3, 'Q3'),
```

```
(20138, 2013, 8, 'August', 3, 'Q3'),
(20139, 2013, 9, 'September', 3, 'Q3'),
(20141, 2014, 1, 'Janauary', 1, 'Q1'),
(201410, 2014, 10, 'October', 4, 'Q4'),
(201411, 2014, 11, 'November', 4, 'Q4'),
(201412, 2014, 12, 'December', 4, 'Q4'),
(20142, 2014, 2, 'February', 1, 'Q1'),
(20143, 2014, 3, 'March', 1, 'Q1'),
(20144, 2014, 4, 'April', 2, 'Q2'),
(20145, 2014, 5, 'May', 2, 'Q2'),
(20146, 2014, 6, 'June', 2, 'Q2'),
(20147, 2014, 7, 'July', 3, 'Q3'),
(20148, 2014, 8, 'August', 3, 'Q3'),
(20149, 2014, 9, 'September', 3, 'Q3'),
(20151, 2015, 1, 'Janauary', 1, 'Q1'),
(201510, 2015, 10, 'October', 4, 'Q4'),
(201511, 2015, 11, 'November', 4, 'Q4'),
(201512, 2015, 12, 'December', 4, 'Q4'),
(20152, 2015, 2, 'February', 1, 'Q1'),
(20153, 2015, 3, 'March', 1, 'Q1'),
(20154, 2015, 4, 'April', 2, 'Q2'),
(20155, 2015, 5, 'May', 2, 'Q2'),
(20156, 2015, 6, 'June', 2, 'Q2'),
(20157, 2015, 7, 'July', 3, 'Q3'),
(20158, 2015, 8, 'August', 3, 'Q3'),
(20159, 2015, 9, 'September', 3, 'Q3'),
(20161, 2016, 1, 'Janauary', 1, 'Q1'),
(201610, 2016, 10, 'October', 4, 'Q4'),
(201611, 2016, 11, 'November', 4, 'Q4'),
(201612, 2016, 12, 'December', 4, 'Q4'),
(20162, 2016, 2, 'February', 1, 'Q1'),
(20163, 2016, 3, 'March', 1, 'Q1'),
(20164, 2016, 4, 'April', 2, 'Q2'),
(20165, 2016, 5, 'May', 2, 'Q2'),
(20166, 2016, 6, 'June', 2, 'Q2'),
(20167, 2016, 7, 'July', 3, 'Q3'),
(20168, 2016, 8, 'August', 3, 'Q3'),
(20169, 2016, 9, 'September', 3, 'Q3'),
(20171, 2017, 1, 'Janauary', 1, 'Q1'),
(201710, 2017, 10, 'October', 4, 'Q4'),
(201711, 2017, 11, 'November', 4, 'Q4'),
(201712, 2017, 12, 'December', 4, 'Q4'),
(20172, 2017, 2, 'February', 1, 'Q1'),
```

```
(20173, 2017, 3, 'March', 1, 'Q1'),
(20174, 2017, 4, 'April', 2, 'Q2'),
(20175, 2017, 5, 'May', 2, 'Q2'),
(20176, 2017, 6, 'June', 2, 'Q2'),
(20177, 2017, 7, 'July', 3, 'Q3'),
(20178, 2017, 8, 'August', 3, 'Q3'),
(20179, 2017, 9, 'September', 3, 'Q3'),
(20181, 2018, 1, 'Janauary', 1, 'Q1'),
(201810, 2018, 10, 'October', 4, 'Q4'),
(201811, 2018, 11, 'November', 4, 'Q4'),
(201812, 2018, 12, 'December', 4, 'Q4'),
(20182, 2018, 2, 'February', 1, 'Q1'),
(20183, 2018, 3, 'March', 1, 'Q1'),
(20184, 2018, 4, 'April', 2, 'Q2'),
(20185, 2018, 5, 'May', 2, 'Q2'),
(20186, 2018, 6, 'June', 2, 'Q2'),
(20187, 2018, 7, 'July', 3, 'Q3'),
(20188, 2018, 8, 'August', 3, 'Q3'),
(20189, 2018, 9, 'September', 3, 'Q3'),
(20191, 2019, 1, 'Janauary', 1, 'Q1'),
(201910, 2019, 10, 'October', 4, 'Q4'),
(201911, 2019, 11, 'November', 4, 'Q4'),
(201912, 2019, 12, 'December', 4, 'Q4'),
(20192, 2019, 2, 'February', 1, 'Q1'),
(20193, 2019, 3, 'March', 1, 'Q1'),
(20194, 2019, 4, 'April', 2, 'Q2'),
(20195, 2019, 5, 'May', 2, 'Q2'),
(20196, 2019, 6, 'June', 2, 'Q2'),
(20197, 2019, 7, 'July', 3, 'Q3'),
(20198, 2019, 8, 'August', 3, 'Q3'),
(20199, 2019, 9, 'September', 3, 'Q3');
```

- Verify sql script (looks like it's runned in a postgres terminal session)

```
theia@theiadocker-craigtrupp8:/home/project$ cat verify.sql
\echo "Checking row in DimMonth Table"
select count(*) from "DimMonth";
\echo "Checking row in DimCustomer Table"
select count(*) from "DimCustomer";
\echo "Checking row in FactBilling Table"
select count(*) from "FactBilling";
```

- Create the Schema
    - Run the command below to create the schema in the billingDW database.

```
psql  -h localhost -U postgres -p 5432 billingDW < star-schema.sql
```

```
theia@theiadocker-craigtrupp8:/home/project$ psql -h localhost -U postgres
-p 5432 billingDW < star-schema.sql
BEGIN
CREATE TABLE
CREATE TABLE
CREATE TABLE
ALTER TABLE
ALTER TABLE
COMMIT
theia@theiadocker-craigtrupp8:/home/project$
```

Exercise 4 - Load data into Dimension tables

- When we load data into the tables, it is a good practice to load the data into dimension tables first.

```
theia@theiadocker-craigtrupp8:/home/project$ psql  -h localhost -U postgres
-p 5432 billingDW < DimCustomer.sql
INSERT 0 1000
theia@theiadocker-craigtrupp8:/home/project$ psql  -h localhost -U postgres
-p 5432 billingDW < DimMonth.sql
INSERT 0 132
theia@theiadocker-craigtrupp8:/home/project$
```

Exercise 5 - Load data into Fact table

```
theia@theiadocker-craigtrupp8:/home/project$ psql  -h localhost -U postgres
-p 5432 billingDW < FactBilling.sql
INSERT 0 132000
```

Exercise 6 - Run a sample query

```
theia@theiadocker-craigtrupp8:/home/project$ psql  -h localhost -U postgres
-p 5432 billingDW < verify.sql
"Checking row in DimMonth Table"
 count
-------
   132
(1 row)

"Checking row in DimCustomer Table"
 count
-------
  1000
(1 row)

"Checking row in FactBilling Table"
 count
--------
 132000
(1 row)

theia@theiadocker-craigtrupp8:/home/project$
```

- Another Sample Query after connection to Database, notice the string encapsulation of the table name after connecting

```
theia@theiadocker-craigtrupp8:/home/project$ psql --username=postgres
--host=localhost
psql (15.2 (Ubuntu 15.2-1.pgdg18.04+1), server 13.2)
Type "help" for help.

postgres=# \l
                                                 List of databases
   Name    |  Owner   | Encoding |  Collate   |   Ctype    | ICU Locale |
Locale Provider |   Access privileges
-----------+----------+----------+------------+------------+------------+--
---------------+-----------------------
 billingDW | postgres | UTF8     | en_US.utf8 | en_US.utf8 |            |
libc            |
 postgres  | postgres | UTF8     | en_US.utf8 | en_US.utf8 |            |
libc            |
```

```
 template0 | postgres | UTF8     | en_US.utf8 | en_US.utf8 |            |
libc            | =c/postgres          +
           |          |          |            |            |            |
| postgres=CTc/postgres
 template1 | postgres | UTF8     | en_US.utf8 | en_US.utf8 |            |
libc            | =c/postgres          +
           |          |          |            |            |            |
| postgres=CTc/postgres
(4 rows)

postgres=# \c billingDW
psql (15.2 (Ubuntu 15.2-1.pgdg18.04+1), server 13.2)
You are now connected to database "billingDW" as user "postgres".
billingDW=# \dt
            List of relations
 Schema |    Name     | Type  |  Owner
--------+-------------+-------+----------
 public | DimCustomer | table | postgres
 public | DimMonth    | table | postgres
 public | FactBilling | table | postgres
(3 rows)

billingDW=# SELECT * FROM FactBilling LIMIT 5;
ERROR:  relation "factbilling" does not exist
LINE 1: SELECT * FROM FactBilling LIMIT 5;
                      ^
billingDW=# SELECT * FROM public.FactBilling LIMIT 5;
ERROR:  relation "public.factbilling" does not exist
LINE 1: SELECT * FROM public.FactBilling LIMIT 5;
                      ^
billingDW=# SELECT * FROM "FactBilling" LIMIT 5;
 billid | customerid | monthid | billedamount
--------+------------+---------+--------------
      1 |          1 |   20091 |         5060
      2 |        614 |   20091 |         9638
      3 |        615 |   20091 |        11573
      4 |        616 |   20091 |        18697
      5 |        617 |   20091 |          944
(5 rows)
```