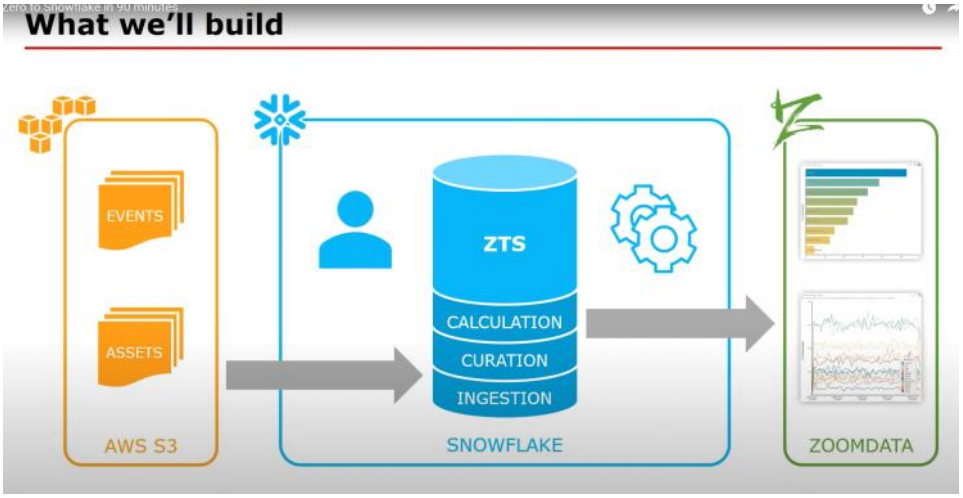
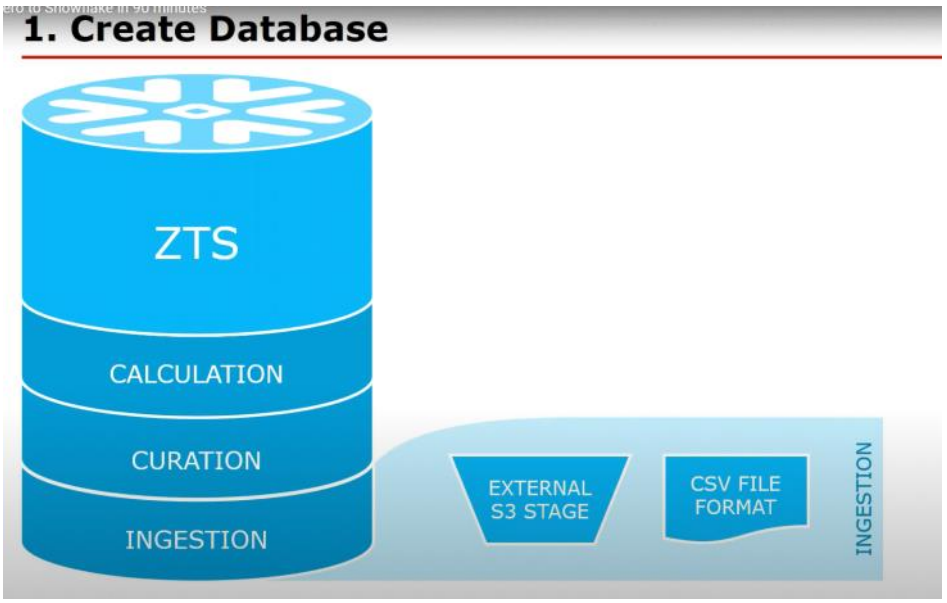
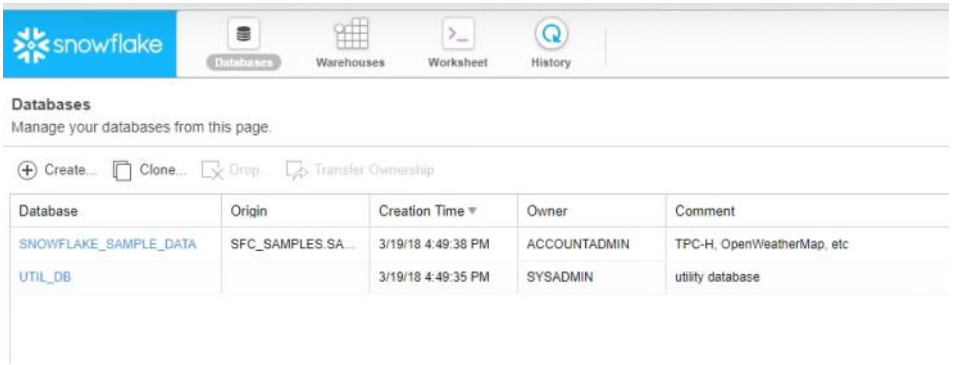
Snowflake

Sunday, April 18, 2021 10:00 AM

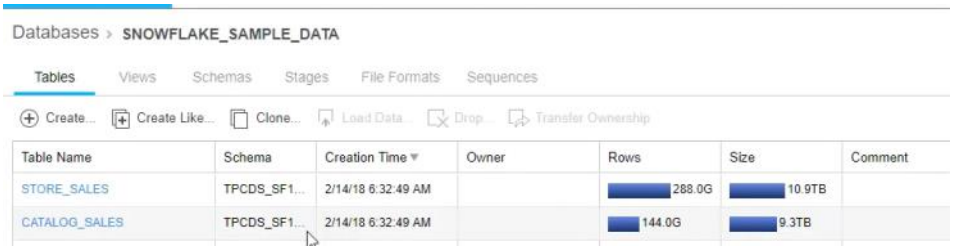


#1.create database





#databse: database gives us lists of the high level database objects that we got.



Warehouses: the pools of compute resources that we use to do operations, e.g. put data in, process data, query daya,
#create database and switch to it

Worksheet : predefined SQL statement

```
CREATE OR REPLACE DATABASE ZTS;
USE ZTS

#create schemas

Create or replace schema ingestion;
Create or replace schema curation;
Create or replace schema calculation;

Show schemas;
```

Row	created_on	name	is_default	is_current	database_name	owner	comment	options	retention_time
1	2018-05-03 10:19:2...	CALCULATION	N	Y	ZTS	ZTS_ADMIN			1
2	2018-05-03 10:19:2...	CURATION	N	N	ZTS	ZTS_ADMIN			1
3	2018-05-03 10:19:3...	INFORMATION_SC...	N	N	ZTS		Views describing th...		1
4	2018-05-03 10:19:2...	INGESTION	N	N	ZTS	ZTS_ADMIN			1
5	2018-05-03 10:18:3...	PUBLIC	N	N	ZTS	ZTS_ADMIN			1

Public is the default schema, information_schema is where all the system views are in terms of the metadata and database itse lf

#create an external, s3 stage(public bucket)

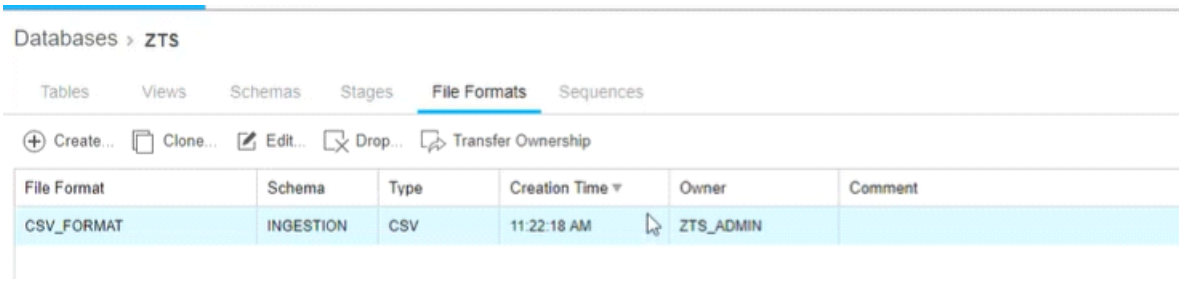
```
Create or replace stage ingestion.s3_stage
        Url = 's3://dlx-zero-to-snowflake'
        Credentials = (
                Aws_key_id = 'AKIAIRIY07UUKKJXX64A'
                                        'YX67150BujDXff1JPeBwtjU8qESyAf3NQkbKZYe9'
                AWS_SECRET_KEY =
        );
```

List @ingestion.s3_stage; #draw us out a list of all the objects and files in that stage

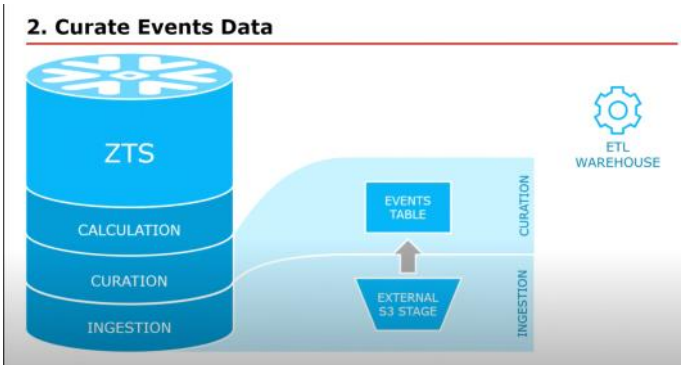
Row	name	size	md5	last_modified
1	s3://dlx-zero-to-snowflake/assets/20180302-102317...	34785	e5665c6c754e92c4210ef86735ad8888	Fri, 2 Mar 2018 10:23:21 GMT
2	s3://dlx-zero-to-snowflake/events/20180302-092512...	5602753	59b53e7c91334c23d46eb8920fff12fb-2	Fri, 2 Mar 2018 09:25:23 GMT
3	s3://dlx-zero-to-snowflake/events/20180302-092512...	5649112	c9c04cd06ff428fb78fb3559c9d9b784-2	Fri, 2 Mar 2018 09:25:23 GMT
4	s3://dlx-zero-to-snowflake/events/20180302-092512...	5613139	e3240727a35d270901dcd56cb24e5d78-2	Fri, 2 Mar 2018 09:25:23 GMT
5	s3://dlx-zero-to-snowflake/events/20180302-092512...	5629833	669ce6f3b74b0c2d7112c55141f87181-2	Fri, 2 Mar 2018 09:25:23 GMT
6	s3://dlx-zero-to-snowflake/events/20180302-092512...	5615627	0bda05ae9f4c12e25fe23b6d130092e3-2	Fri, 2 Mar 2018 09:25:32 GMT

#create CSV files format for ingestion

```
-- Create CSV file format for ingestion

CREATE OR REPLACE FILE FORMAT INGESTION.CSV_FORMAT
    TYPE = 'CSV'
    COMPRESSION = 'AUTO'
    FIELD_DELIMITER = ','
    RECORD_DELIMITER = '\n'
    SKIP_HEADER = 1
    FIELD_OPTIONALLY_ENCLOSED_BY = '\042'
    ERROR_ON_COLUMN_COUNT_MISMATCH = TRUE
    ESCAPE = 'NONE'
    ESCAPE_UNENCLOSED_FIELD = '\134'
    NULL_IF = ('NULL');
```

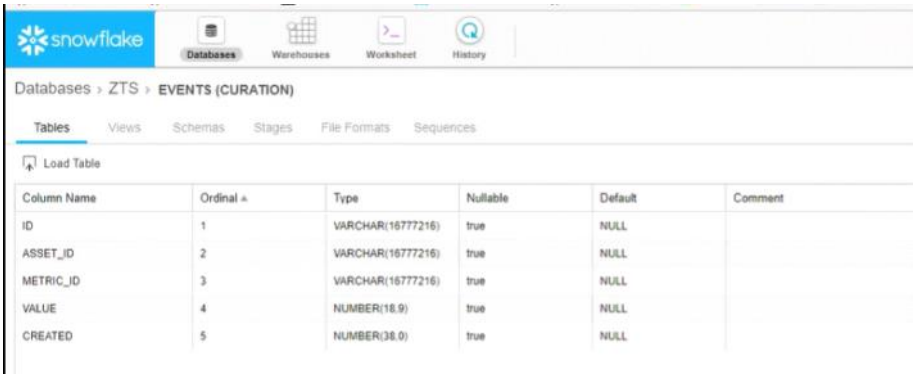


2.curate events data



#create curation table for events
Create or replace table curation.events(
Id string,
Asset_id string,
Metric_id string,
Value number(18,9),
Created int
);

Desc table curation.events; -- Also look at this via the database tab



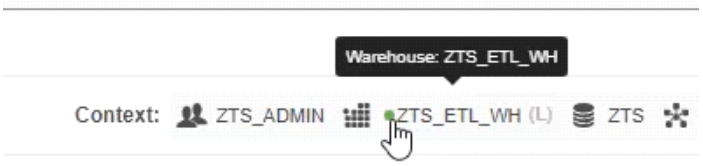
#now need a warehouse

CREATE or replace warehouse zts_etl_mh with
Warehouse_size = 'large'
Auto_suspend = 300
Auto_resume = true
Comment = 'Zero to Snowflake ETL Warehouse';

Grant usage on warehouse zts_etl_wh to role zts_admin;

Use warehouse zts_etl_wh;

#green dot means live



copy in events from s3 stage(LARGE 1M20S)

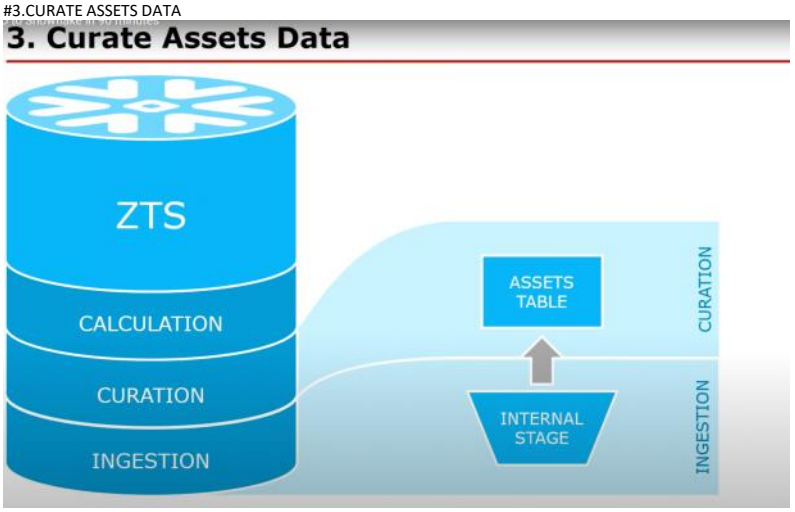
```
Copy into curation.events
    From @ingestion.s3_stage/events/
    FILE_FORMAT = (FORMAT_NAME = 'INGESTION.CSV_FORMAT');

SELECT * FROM CURATION.EVENTS LIMIT 10;
```

Row	ID	ASSET_ID	METRIC_ID	VALUE	CREATED
1	49cf7594-bcca-4cd7-b4ad-3318d1d7f2cb	aab33e6c-7fcc-33e5-b422-e0cf5e21de12	77a379dc-6603-3d6a-8d2c-55b7ec685f65	0.401249200	1523338880
2	a5f71dac-34c2-4e78-b25f-9106cd04bc1f	98cf19c2-e12e-37cc-9e0c-e73f285da5f6	c40cf251-a6d0-3712-aaee-209ccae1fed3	0.133727875	1523338776
3	8ead7a30-2747-46b2-a186-1c99d9aad4e8	44d6c837-1bca-3c69-9eff-30b235aa624f	376b4c88-9410-325a-8ed5-67deae77bf0d	0.149774081	1523339609
4	11ae4f66-fc2c-4dab-8f3d-ea88a92ef737	dd3aa249-aa9b-308a-8501-412d38630f67	d93864ab-bf1a-3b1c-b551-c89f95cccbc8	0.281515376	1523339479
5	2a3a5efc-e381-4750-82ec-318a09f93c10	b9aafce4-175b-3c92-839a-bfd50ad5e5bc	c40cf251-a6d0-3712-aaee-209ccae1fed3	0.964294689	1523336596
6	eb35f8ca-f515-498d-ae88-52583328259a	0c172076-fc19-3572-af7b-8af1bc29b173	f8e59b70-32de-37b4-9523-fabb198957f2	0.650793384	1523338231

#Try again with 2xl (30S)

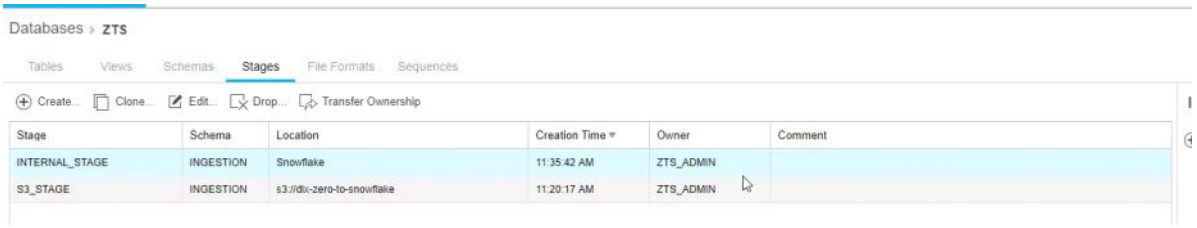
```
TRUNCATE TABLE CURATION.EVENTS;
ALTER WAREHOUSE ZTS_ETL_WH SET WAREHOUSE_SIZE = 'XXLARGE';

#switch the warehouse back to large for now (THIS SORT OF THING CAN BE AUTOMATED)

ALTER WAREHOUSE ZTS_ETL_WH SET WAREHOUSE_SIZE = 'LARGE';
```

#3.CURATE ASSETS DATA



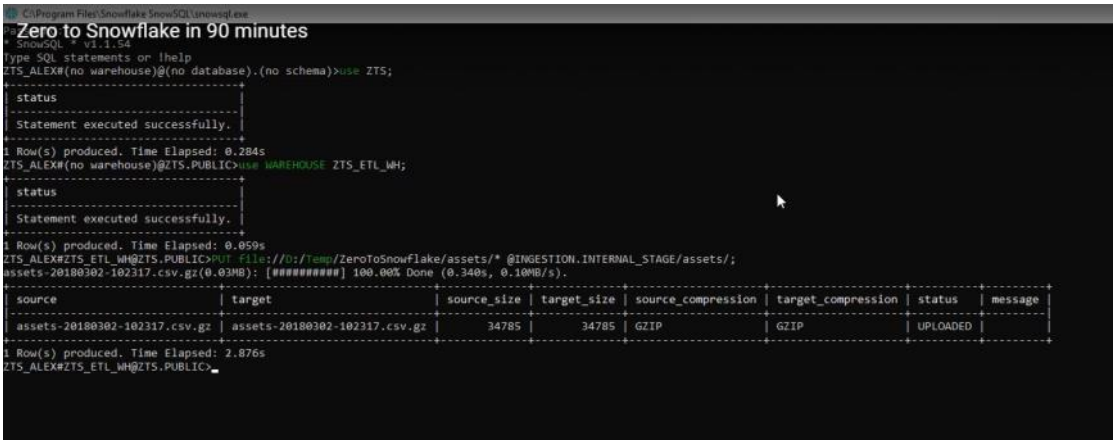
#CREATE AN INTERNAL, SNOWFLAKE-NAMAGED STAGE

```
CREATE OR REPLACE STAGE INGESTION.INTERNAL_STAGE;
```

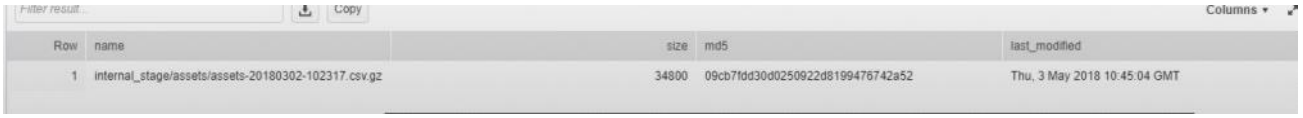


-- push this assets file using snowsql

```
-- Push the assets file using SnowSQL

/*
snowsql --accountname datalytyx.eu-west-1 --username ZTS_ALEX
USE ZTS;
USE WAREHOUSE ZTS_ETL_WH;
PUT file://D:/Temp/ZeroToSnowflake/assets/* @INGESTION.INTERNAL_STAGE/assets/;
*/

LIST @INGESTION.INTERNAL_STAGE;
```



LIST @INGESTION.INTERNAL_STAGE



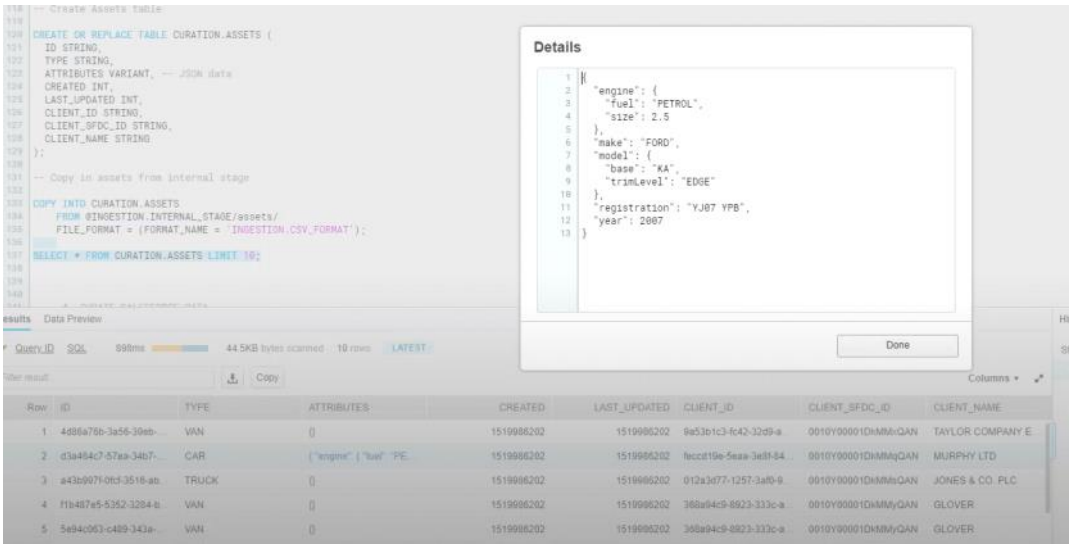
#Create Assets table

```
CREATE OR REPLACE TABLE CURATION.ASSETS (
	ID STRING,
	TYPE STRING,
	ATTRIBUTES VARIANT,  #JASON DATA/UNSTRUCTURE DATA
	CREATED INT,
	LAST_UPDATED INT,
	CLIENT_ID STRING,
	CLIENT_SFDC_ID STRING,
	CLIENT_NAME STRING
);
```

#COPY IN ASSETS FROM INTERNAL STAGE

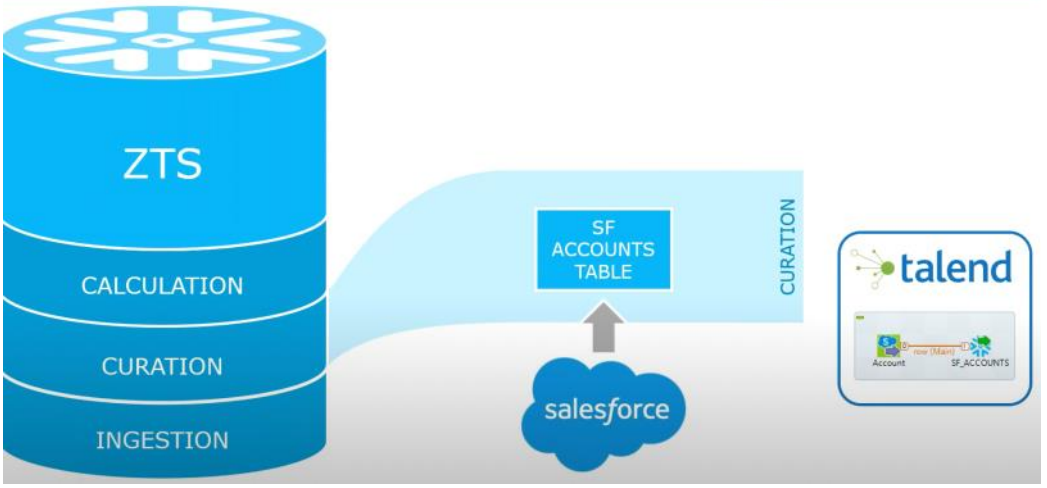
```
COPY INTO CURATION.ASSETS
	FROM @INGESTION.INTERNAL_STAGE/assets/
	FILE_FORMAT = (FORMAT_NAME = 'INGESTION.CSV_FORMAT');
```

SELECT * FROM CURATION.ASSETS LIMIT 10;



4. CURATE SALEFORCE DATA

4. Curate Salesforce Data

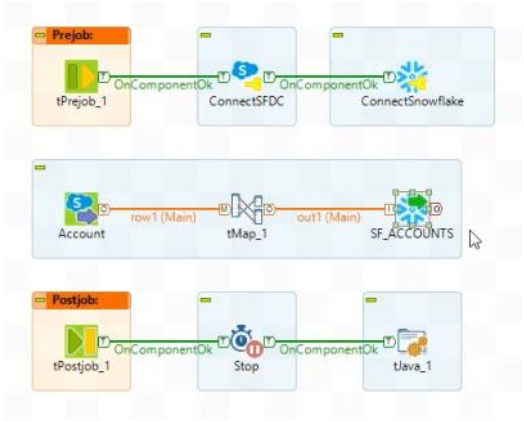


#LOOK AT THE ACCOUNTS IN SALESFORCE
#CREATE SF ACCOUNTS TABLE

```
CREATE OR REPLACE TABLE CURATION.SF_ACCOUNTS (
        ID STRING,
        ISDELETED BOOLEAN,
        MASTERRECORDID STRING,
        NAME STRING,
        TYPE STRING,
        PARENTID STRING,
        BILLINGSTREET STRING,
        BILLINGCITY STRING,
        BILLINGSTATE STRING,
        BILLINGPOSTALCODE STRING,
        BILLINGCOUNTRY FLOAT,
        BILLINGLATITUDE FLOAT,
        BILLINGGEOCODEACCURACY STRING,
        BILLINGADDRESS STRING,
        SHIPPINGSTREET STRING,
        SHIPPINGCITY STRING,
        SHIPPINGSTATE STRING,
        SHIPPINGPOSTALCODE STRING,
        SHIPPINGCOUNTRY FLOAT,
        SHIPPINGLATITUDE FLOAT,
        SHIPPINGGEOCODEACCURACY STRING,
        SHIPPINGADDRESS STRING,
        PHONE STRING,
```

```
    ANNUALREVENUE NUMBER(18,2),
    NUMBEROFEMPLOYEES INTEGER,
    OWNERSHIP STRING,
    TICKERSYMBOL STRING,
    DESCRIPTION STRING,
    RATING STRING,
    SITE STRING,
    OWNERID STRING,
    CREATEDDATE TIMESTAMP_NTZ,
    CREATEDBYID STRING,
    LASTMODIFIEDDATE TIMESTAMP_NTZ,
    LASTMODIFIEDBYID STRING,
    SYSTEMMODSTAMP TIMESTAMP_NTZ,
    LASTACTIVITYDATE TIMESTAMP_NTZ,
    LASTVIEWEDDATE TIMESTAMP_NTZ,
    LASTREFERENCEDDATE TIMESTAMP_NTZ,
    JIGSAW STRING,
    JIGSAWCOMPANYID STRING,
    CLEANSTATUS STRING,
    ACCOUNTSOURCE STRING,
    DUNSNUMBER STRING,
    TRADESTYLE STRING,
    NAICSCODE STRING,
    NAICSDESC STRING,
    YEARSTARTED STRING,
    SICDESC STRING,
    DANDBCOMPANYID STRING,
    CUSTOMERPRIORITY__C STRING,
    SLA__C STRING,
    ACTIVE__C STRING,
    NUMBEROFLOCATIONS__C FLOAT,
    UPSELLOPPORTUNITY__C STRING,
    SLASERIALNUMBER__C STRING,
    SLAEXPIRATIONDATE__C TIMESTAMP_NTZ
);
```

#LOAD THE sf ACCOUNTS USING Talend

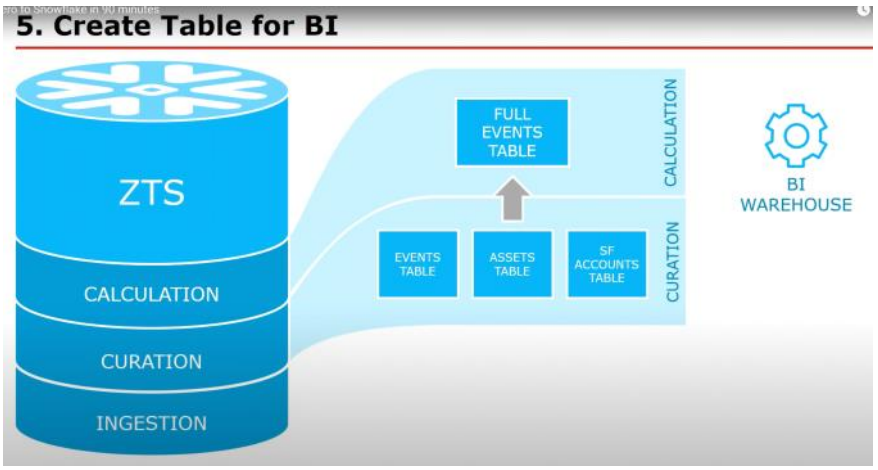


#run job Demo/ZeroToSnowflake/ZeroToSnowflake30_IngestSFDCAccountsToSnoeflake

SELECT * FROM CURATION.SF_ACCOUNTS LIMIT 10;

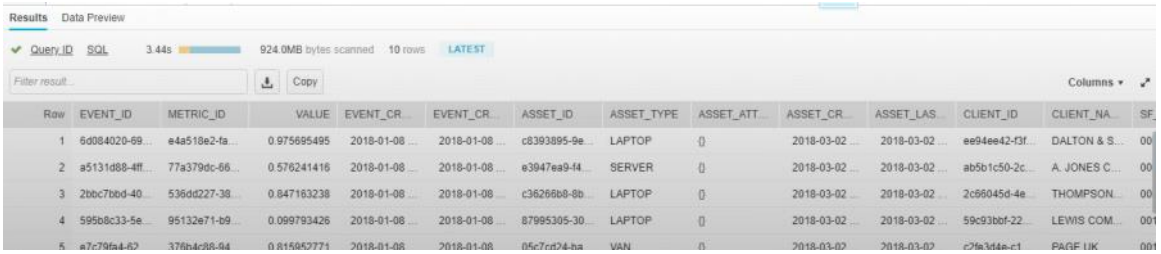
Row	ID	ISDELETED	MASTERRE...	NAME	TYPE	PARENTID	BILLINGSTR...	BILLINGCITY	BILLINGSTATE	BILLINGPO...	BILLINGCO...	BILLINGLATI...	BILL
1	0010Y00001...	FALSE	NULL	Dalton & Ste...	Customer - ...	NULL	NULL	NULL	NULL	NULL	NULL	NULL	
2	0010Y00001...	FALSE	NULL	A. Jones Co...	Customer - ...	NULL	NULL	NULL	NULL	NULL	NULL	NULL	
3	0010Y00001...	FALSE	NULL	Taylor Comp...	Customer - ...	NULL	NULL	NULL	NULL	NULL	NULL	NULL	
4	0010Y00001...	FALSE	NULL	Thompson &...	Customer - ...	NULL	NULL	NULL	NULL	NULL	NULL	NULL	
5	0010Y00001...	FALSE	NULL	Stokes & Bar...	Customer - ...	NULL	NULL	NULL	NULL	NULL	NULL	NULL	
6	0010Y00001...	FALSE	NULL	Edwards Fur...	Customer - ...	NULL	NULL	NULL	NULL	NULL	NULL	NULL	

5. Create Table for BI



-- Create a flattened events table, joining with the other tables

```
CREATE OR REPLACE TABLE CALCULATION.EVENTS AS
SELECT
        E.ID AS EVENT_ID,
        E.METRIC_ID,
        E.VALUE,
        TO_TIMESTAMP_NTZ(E.CREATED) AS EVENT_CREATED,
        DATE_TRUNC('MINUTE', TO_TIMESTAMP_NTZ(E.CREATED)) AS EVENT_CREATED_MINUTE,
        A.ID AS ASSET_ID,
        A.TYPE AS ASSET_TYPE,
        A.ATTRIBUTES AS ASSET_ATTRIBUTES,
        TO_TIMESTAMP_NTZ(A.CREATED) AS ASSET_CREATED,
        TO_TIMESTAMP_NTZ(A.LAST_UPDATED) AS ASSET_LAST_UPDATED,
        A.CLIENT_ID,
        A.CLIENT_NAME,
        S.ID AS SF_ACCOUNT_ID,
        S.NAME AS SF_ACCOUNT_NAME,
        S.TYPE AS SF_ACCOUNT_TYPE,
        S.INDUSTRY AS SF_ACCOUNT_INDUSTRY
 FROM CURATION.EVENTS E
INNER JOIN CURATION.ASSETS A ON E.ASSET_ID = A.ID
INNER JOIN CURATION.SF_ACCOUNTS S ON A.CLIENT_SFDC_ID = S.ID;

#take a look at the table, noting that data comes from the WH cache
SELECT * FROM CALCULATION.EVENTS LIMIT 10;
```

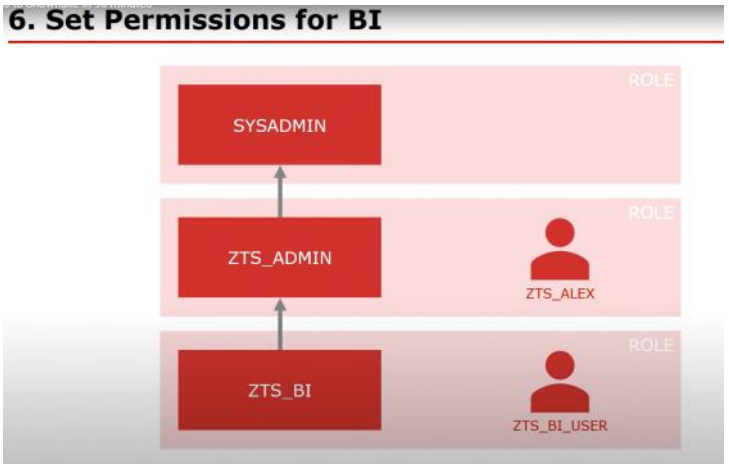


#RUN THE SELECT AGAIN, NOTING THAT DO DATA IS SCANNED(RESULTS CACHE)

#CREATE WAREHOUSE FOR bi

```
CREATE OR REPLACE WAREHOUSE ZTS_BI_WH WITH
       WAREHOUSE_SIZE = 'MEDIUM'
       AUTO_SUSPEND = 300
       AUTO_RESUME = TRUE
       COMMENT = 'Zero to Snowflake BI Warehouse';
```

6. Set permission for BI



#assume accountadmin role

```
Use role accountadmin;  #or use menu
```

#create role for BI

```
Create or replace role ZTS_BI COMMENT = 'Zero to Snowflake - BI Role';
GRANT ROLE ZTS_BI TO ROLE ZTS_ADMIN;

GRANT USAGE ON DATABASE ZTS TO ROLE ZTS_BI;
GRANT USAGE ON SCHEMA ZTS.CALCULATION TO ROLE_ZTS_BI;
GRANT USAGE ON WAREHOUSE ZTS_BI_WH TO ROLE ZTS_BI;
```

#GRANT BI ROLE SELECT ACCESS TO FLATTENED EVENTS TABLE

```
GRANT SELECT ON CALCULATION.EVENTS TO ROLE ZTS_BI;
```

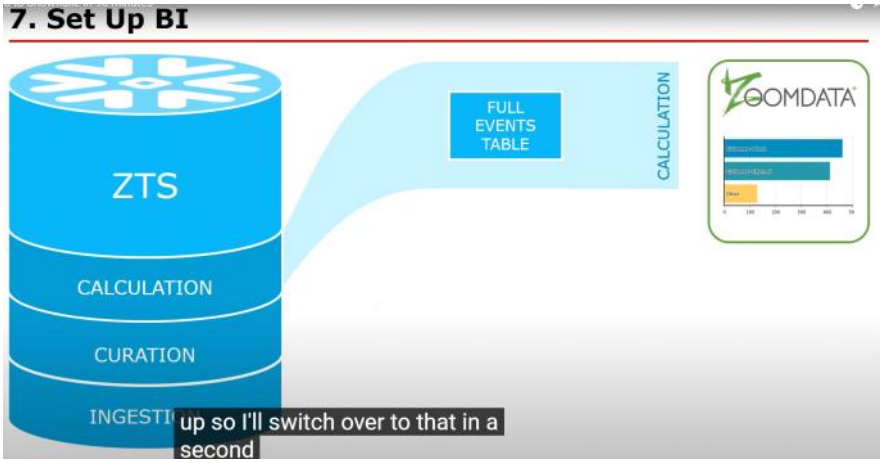
#CREATE USER FOR BI (USER UI - SWITCH ROLE TO ACCOUNTADMIN

```
CREATE OR REPLACE USER ZTS_BI_USER
       PASSWORD = 'Brear5'
       DISPLAY_NAME = 'BI User'
       COMMENT = 'Zero to Snowflake - BI user'
       DEFAULT_ROLE = ZTS_BI
       DEFAULT_WAREHOUSE = ZTS_BI_WH
       ;

GRANT ROLE ZTS_BI TO USER ZTS_BI_USER;
```

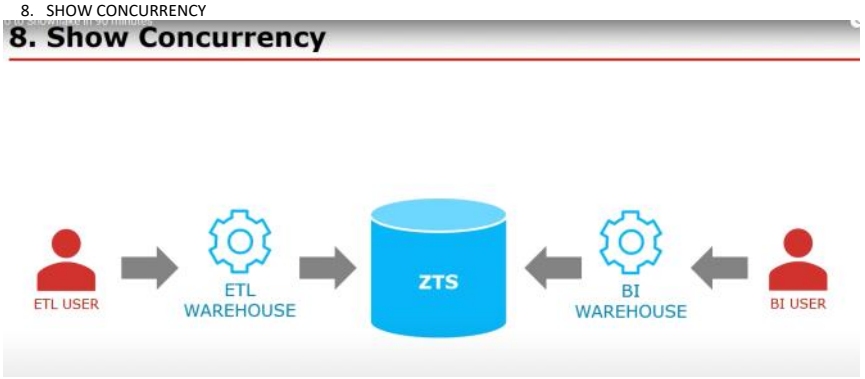
#switch back to ZTS_ADMIN ROLE

```
USE ROLE ZTS_ADMIN;
```

7. SETUP BI



8. SHOW CONCURRENCY

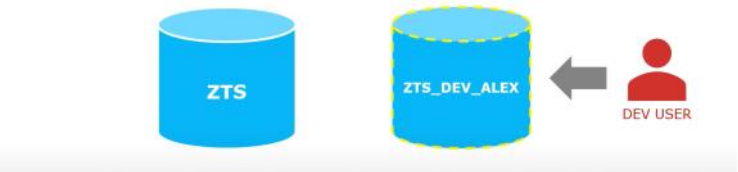



```
#USE etl WAREHOUSE

USE WAREHOUSE ZTS_ETL_WH;

#TURN OFF THE RESULTS CACHE FOR NOW

ALTER SESSION SET USE_CASHED_RESULT = FALSE;

#CALCULATE DIFFERENCE OF DIFFERENCES (3 POINTS) AVERAGED BY ASSET TYPE

SELECT ASSET_TYPE, AVG(DIFF) FROM (
	SELECT
		ASSET_TYPE,
		VALUE AS CURRRENT_VALUE,
		LAG(VALUE) OVER(PARTITION BY ASSET_ID, METRIC_ID ORDER BY EVENT_CREATED) AS PREV_VALUE,
		LEAD(VALUE) OVER ( PARTITION BY ASSET_ID, METRIC_ID ORDER BY EVENT_CREATED) AS NEXT_VALUE,
		CURRENT_VALUE - PREV_VALUE AS PREV_DIFF,
		NEXT_VALUE - CURRENT_VALUE AS NEXT-DIFF,
		NEXT_DIFF - PREV_DIFF AS DIFF
	FROM CALCULATION.EVENTS
	)
GROUP BY ASSET_TYPE
ORDER BY ASSET_TYPE
;

#Show zoomdata still working
#look at the query profile
#turn the results cache back on
```

9. CREATE DEVELOPMENT SANDBOX

ZTS_DEV_ALEX :It's a zero copy clone which means it takes a copy of the database and all the metadata but it refers back to t he original underlying blocks on storage now.

9. Create Development Sandbox

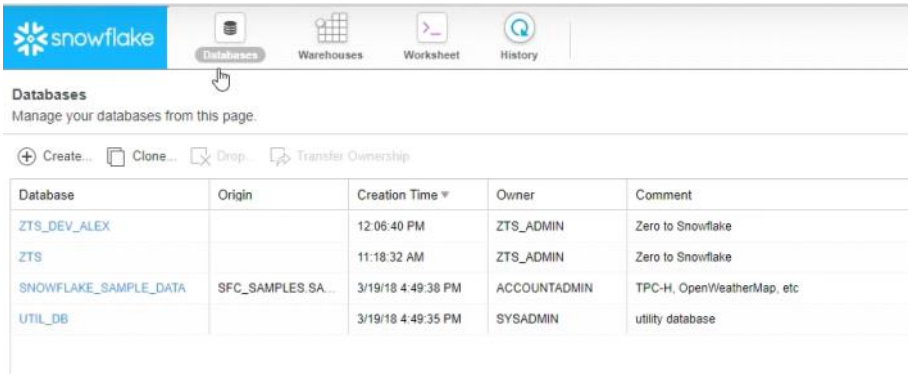


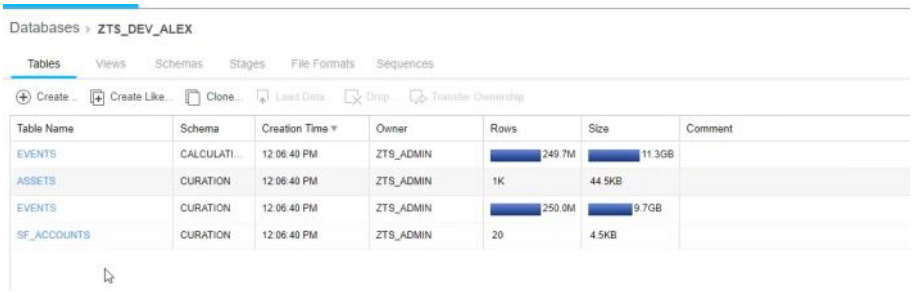
#assume ZTS_ADMIN ROLE (ALSO SWITCH ui BACK TO ZTS_ADMIN)

USE ROLE ZTS_ADMIN;

#CREATE A TEMPORARY DEVELOPMENT DATABSE

CREATE OR REPLACE DATABASE ZTS_DEV_ALEX CLONE ZTS;
USE ZTS_DEV_ALEX;





#show the asset types we currently have

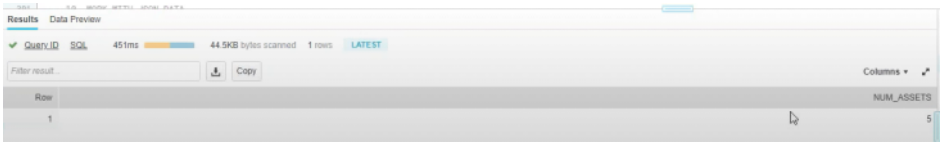
SELECT TYPE, COUNT(ID) AS NUM_ASSETS
FROM CUEATION.ASSETS
GROUP BY TYPE
ORDER BY NUM_ASSETS;

#INSERT A LOAD OF TEST ASSETS

INSERT INTO CURATION.ASSETS (ID, TYPE) VALUES
('00000000-0000-0000-0000-00000000001','TEST_ASSET'),
('00000000-0000-0000-0000-00000000002','TEST_ASSET'),
('00000000-0000-0000-0000-00000000003','TEST_ASSET'),
('00000000-0000-0000-0000-00000000004','TEST_ASSET'),
('00000000-0000-0000-0000-00000000005','TEST_ASSET');

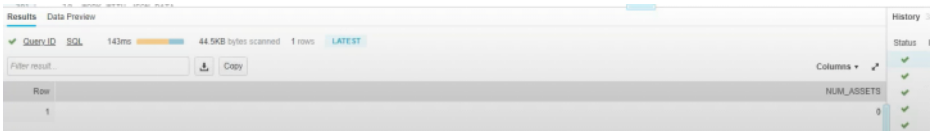
#show the test assets are now in our sandbox

```
SELECT COUNT(ID) AS NUM_ASSETS
FROM ZTS_DEV_ALEX.CURATION.ASSETS
WHERE TYPE = 'TEST_ASSET';
```

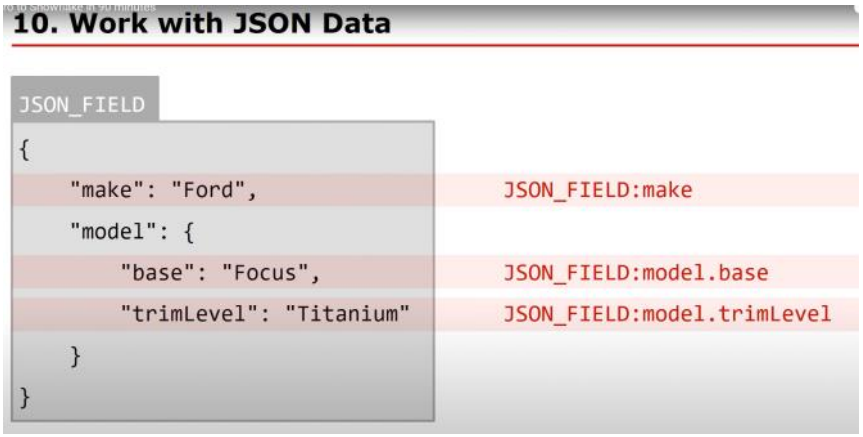


#but not in the production database

```
SELECT COUNT(ID) AS NUM_ASSETS
FROM ZTS.CURATION.ASSETS
WHERE TYPE = 'TEST_ASSET';
```



10. WORK WITH JASON DATA



#still in the sandbox, look at some JSON data

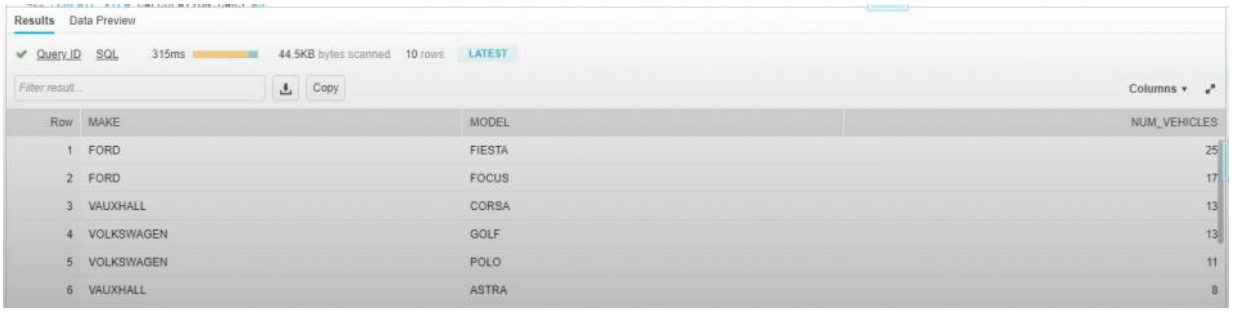
```
SELECT ATTRIBUTES FROM CURATION.ASSETS
WHERE TYPE = 'CAR'
LIMIT 1;
```

#SEE WHAT THE DATA LOOKS LIKE:

#SHOW TOP10 MAKES/MODELS

```
SELECT
        ATTRIBUTES:make::STRING AS MAKE,  #NEED TO DEFINE THE DATA FORMATS
        ATTRIBUTES:model.base::STRING AS MODEL,
        COUNT(*) AS NUM_VEHICLES
FROM CURATION.ASSETS
```

```
WHERE TYPE = 'CAR'
GROUP BY MAKE, MODEL
ORDER BY NUM_VEHICLES DESC
LIMIT 10;
```



Results | Data Preview

Query ID | SQL | 315ms | 44.5KB bytes scanned | 10 rows | LATEST

Filter result... | Copy | Columns

Row	MAKE	MODEL	NUM_VEHICLES
1	FORD	FIESTA	25
2	FORD	FOCUS	17
3	VAUXHALL	CORSA	13
4	VOLKSWAGEN	GOLF	13
5	VOLKSWAGEN	POLO	11
6	VAUXHALL	ASTRA	8

#show the avg(mid) year of all GG prefix registrations

```
SELECT
	MEDIAN(ATTRIBUTES:year::INTEGER)::INTEGER AS AVG_YEAR
FROM CURATION.ASSETS
WHERE
	TYPE = 'CAR'
	AND LEFT(ATTRIBUTES:registration::STRING,2) = 'GG';
```

Results | Data Preview

Query ID | SQL | 91ms | 44.5KB bytes scanned | 1 rows | LATEST

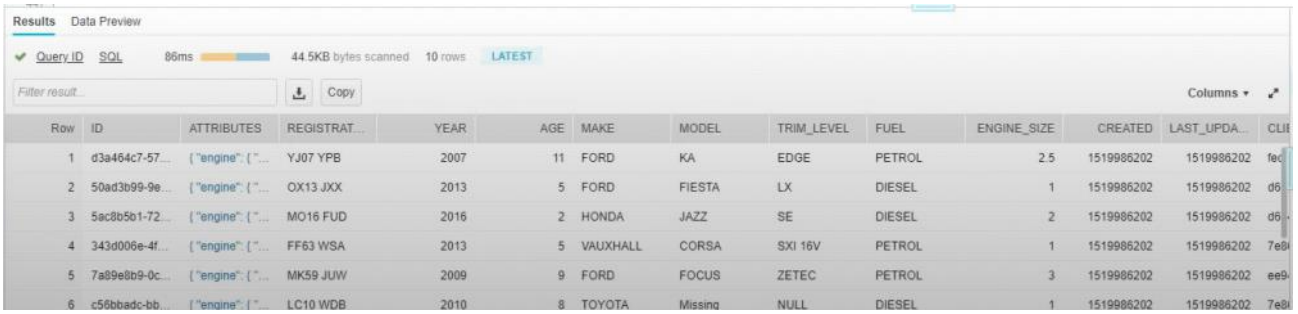
Filter result... | Copy | Columns

Row	AVG_YEAR
1	2013

#CREATE A VIEW OF ALL car ASSETS

```
CREATE VIEW CALCULATION.CARS AS
SELECT
	ID,
	ATTRIBUTES,
	ATTRIBUTES:registration::STRING AS REGISTRATION,
	ATTRIBUTES:year::INTEGER AS YEAR,
	DATE_PART(YEAR,CURRENT_DATE()) - YEAR AS AGE,
	ATTRIBUTES:make::STRING AS MAKE,
	ATTRIBUTES:model.base::STRING AS MODEL,
	ATTRIBUTES:model.trimLevel::STRING AS TRIM_LEVEL,
	ATTRIBUTES:engine.fuel::STRING AS FUEL,
	ATTRIBUTES:engine.size::FLOATAS ENGINE_SIZE,
	CREATED,
	LAST_UPDATED,
	CLIENT_ID
FROM CURATION.ASSETS
WHERE TYPE = 'CAR'
;
```

```
SELECT * FROM CALCULATION.CARS LIMIT 10;
```



Results | Data Preview

Query ID | SQL | 86ms | 44.5KB bytes scanned | 10 rows | LATEST

Filter result... | Copy | Columns

Row	ID	ATTRIBUTES	REGISTRAT...	YEAR	AGE	MAKE	MODEL	TRIM_LEVEL	FUEL	ENGINE_SIZE	CREATED	LAST_UPDA...	CLI
1	d3a464c7-57...	{ "engine": { "...	YJ07 YPB	2007	11	FORD	KA	EDGE	PETROL	2.5	1519986202	1519986202	fe
2	50ad3b99-9e...	{ "engine": { "...	OX13 JXX	2013	5	FORD	FIESTA	LX	DIESEL	1	1519986202	1519986202	d6
3	5ac8b5b1-72...	{ "engine": { "...	MO16 FUD	2016	2	HONDA	JAZZ	SE	DIESEL	2	1519986202	1519986202	d6
4	343d006e-4f...	{ "engine": { "...	FF63 WSA	2013	5	VAUXHALL	CORSA	SXI 16V	PETROL	1	1519986202	1519986202	7e8
5	7a89e8b9-0c...	{ "engine": { "...	MK59 JUW	2009	9	FORD	FOCUS	ZETEC	PETROL	3	1519986202	1519986202	ee9
6	c56bbadc-bb...	{ "engine": { "...	LC10 WDB	2010	8	TOYOTA	Missing	NULL	DIESEL	1	1519986202	1519986202	7e8

#DROP THE SANDBOX DATABASE

```
DROP DATABASE ZTS_DEV_ALEX;
```

11. ACCIDENTS HAPPEN -- UPDROP OUR SANDBOX DATABASE

```
SHOW DATABASES LIKE 'ZTS%';

SHOW DATABASE HISTORY LIKE 'ZTS%';

UNDROP DATABASE ZTS_DEV_ALEX;
```

#SWITCH TO PRODUCTION

USE ZTS;

#BREAK SOME DATA : HOW MANY CARS ARE THERE?

SELECT TYPE, COUNT(ID) AS NUM_ASSETS
FROM CURATION.ASSETS
WHERE TYPE = 'CAR'
GROUP BY TYPE'

#NOW ACCIDENTALLY DELETE THEM ALL

DELETE FROM CURATION.ASSETS WHERE TYPE = 'CAR';

#LOOK THEY'RE GONE

SELECT COUNT(ID) AS NUM_ASSETS
FROM CURATION.ASSETS
WHERE TYPE = 'CAR';

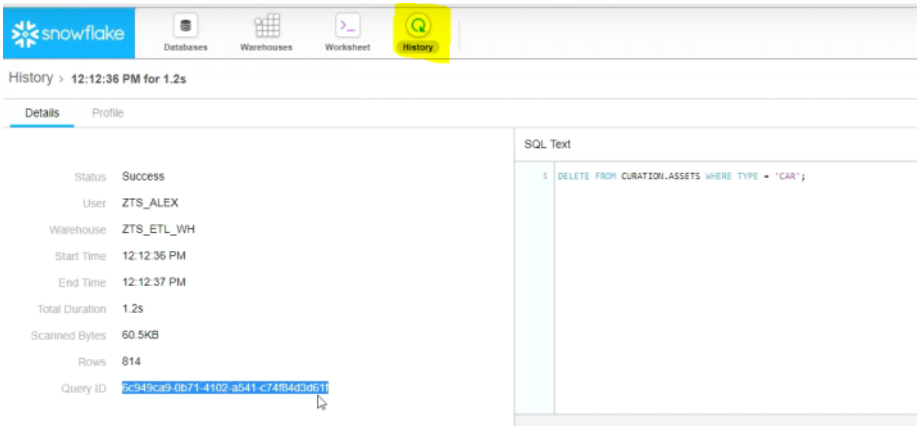
#LET'S GO BACK IN TIME TO 5 MINUTES AGO

SELECT COUNT(ID) AS NUM_ASSETS
FROM CURATION.ASSETS
AT (OFFSET >= -60*5)
WHERE TYPE = 'CAR';

#WE CAN USE THE TABLE AT A POINT IMMEDIATELY BEFORE WE RAN THE DELETE

SELECT COUNT(ID) AS NUM-ASSETS
FROM CURATION.ASSETS
BEFORE (STATEMENT => '')
WHERE TYPE = 'CAR'
;

#GO TO THE HISTORY AND FIND THE HISTORY QUERY ID



#SO THAT'S SELECT THOSE ROWS BACK INTO THE CURRENT TABLE

SELECT COUNT(ID) AS NUM-ASSETS
FROM CURATION.ASSETS
BEFORE (STATEMENT => '6c949ca9-0b71-410-a541-c74f84d3d61f')
WHERE TYPE = 'CAR'
;

#look they're back

SELECT COUNT(ID) AS NUM_ASSETS
FROM CURATION.ASSETS
WHERE TYPE = 'CAR'
;