1. CREATE SCHEMA `amazoncellphone` ;

CREATE TABLE `items` (

`asin` text,

`brand` text,

`title` text,

`url` text,

`image` text,

`rating` double DEFAULT NULL,

`reviewUrl` text,

`totalReviews` int(11) DEFAULT NULL,

`prices` text

) ;

CREATE TABLE `reviews` (

`asin` text,

`name` text,

`date` text,

`verified` text,

`helpfulVotes` text

) ;

1. LOAD DATA LOCAL INFILE 'MacintoshHD/items'

INTO TABLE amazoncellphone.items

FIELDS TERMINATED BY ','

ENCLOSED BY '"'

LINES TERMINATED BY '\n'

IGNORE 1 ROWS;

LOAD DATA LOCAL INFILE 'MacintoshHD/reviews'

INTO TABLE amazoncellphone.reviews

FIELDS TERMINATED BY ','

ENCLOSED BY '"'

LINES TERMINATED BY '\n'

IGNORE 1 ROWS;

SELECT

`asin` ,

`brand` ,

`title` ,

`url` ,

`image`,

`rating`,

`reviewUrl`,

`totalReviews`,

`prices`

FROM Items

INTO OUTFILE ''MacintoshHD/Export/export\_items.csv'

FIELDS ENCLOSED BY '"'

TERMINATED BY ';'

ESCAPED BY '"'

LINES TERMINATED BY '\r\n';

SELECT

`asin`,

`name`,

`date`,

`verified` ,

`helpfulVotes`

FROM reviews

INTO OUTFILE ''MacintoshHD/Export/export\_reviews.csv'

FIELDS ENCLOSED BY '"'

TERMINATED BY ';'

ESCAPED BY '"'

LINES TERMINATED BY '\r\n';

1. *See “selector\_example.py” & and “page\_selectors.yml” in SCRIPTS folder*

7)

Select items.asin , items.title, count(reviews.asin)

from items , reviews

where items.asin = reviews.asin

group by items.asin,items.title

INTO OUTFILE ''MacintoshHD/Desktop/export\_reviews\_count.csv'

FIELDS ENCLOSED BY '"'

TERMINATED BY ';'

ESCAPED BY '"'

LINES TERMINATED BY '\r\n';

8) *See “top10\_pie.xls” & “export\_reviews\_count.csv” in EXPORTS folder*

9)

Simple ETL -

- Scheduled via a cron job.

- Python and pandas script running inside a docker container and scheduled via a cron job in an instance.

- SQL queries on MYSQL

- Queries are scheduled externally using a cron job from a host.

- ETL tool, eg: Informatica / SSIS.

- Automatic scheduling. / can run transformations, can connect with MYSQL

- Is a paid tool

- Scheduling via Airflow.

- Better suited for complex ETL jobs with multiple dependencies (data sources and ETL scripts).

- Can run on managed environments (Eg: Cloud composer)

10)

Approach A: Just 1 query with small data, and the aggregation takes very less time + not many stakeholders requesting it.

* Have a web-service run the query on demand on the database and output as a web page.

Approach B: There are more complex queries, the stakeholders need to drill down and rollup counts based on dynamically

generated conditions. The report needs more flexibility.

* Create the report in a dashboarding took like tableau, and run the reports from a tableau server which connects to the database and refreshes the queries.