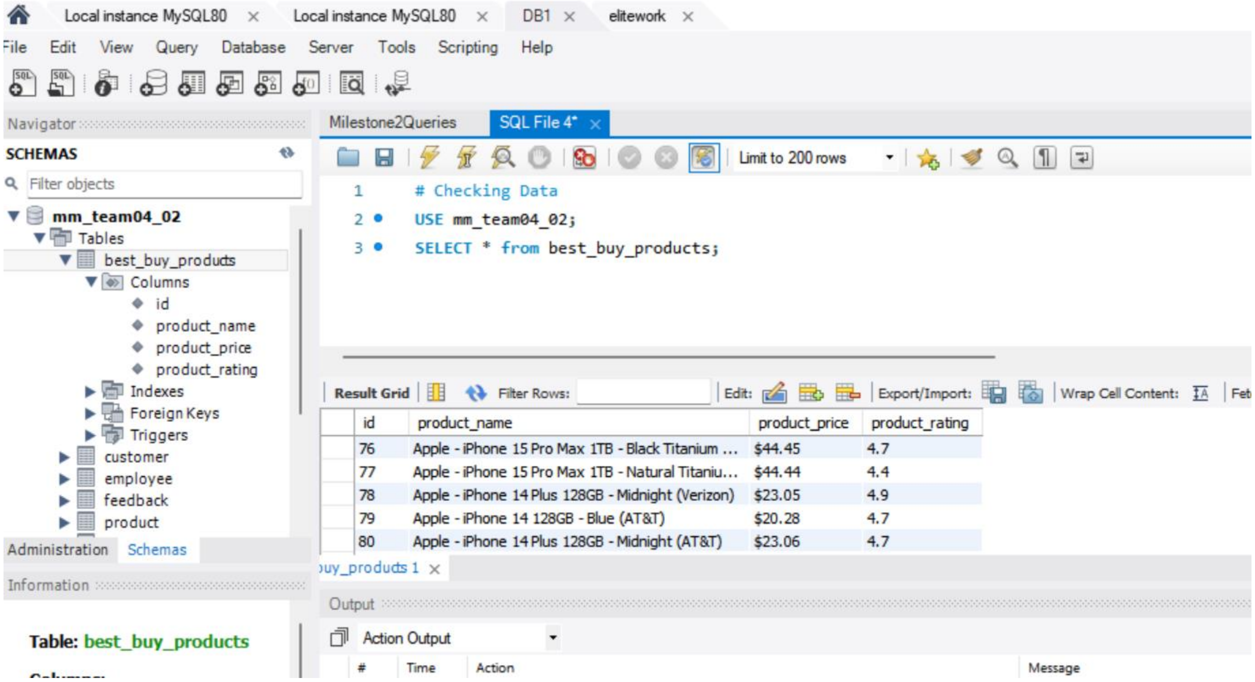
Additional Feature Works:

Part 1:

Web Scraping and Data Storage Application:

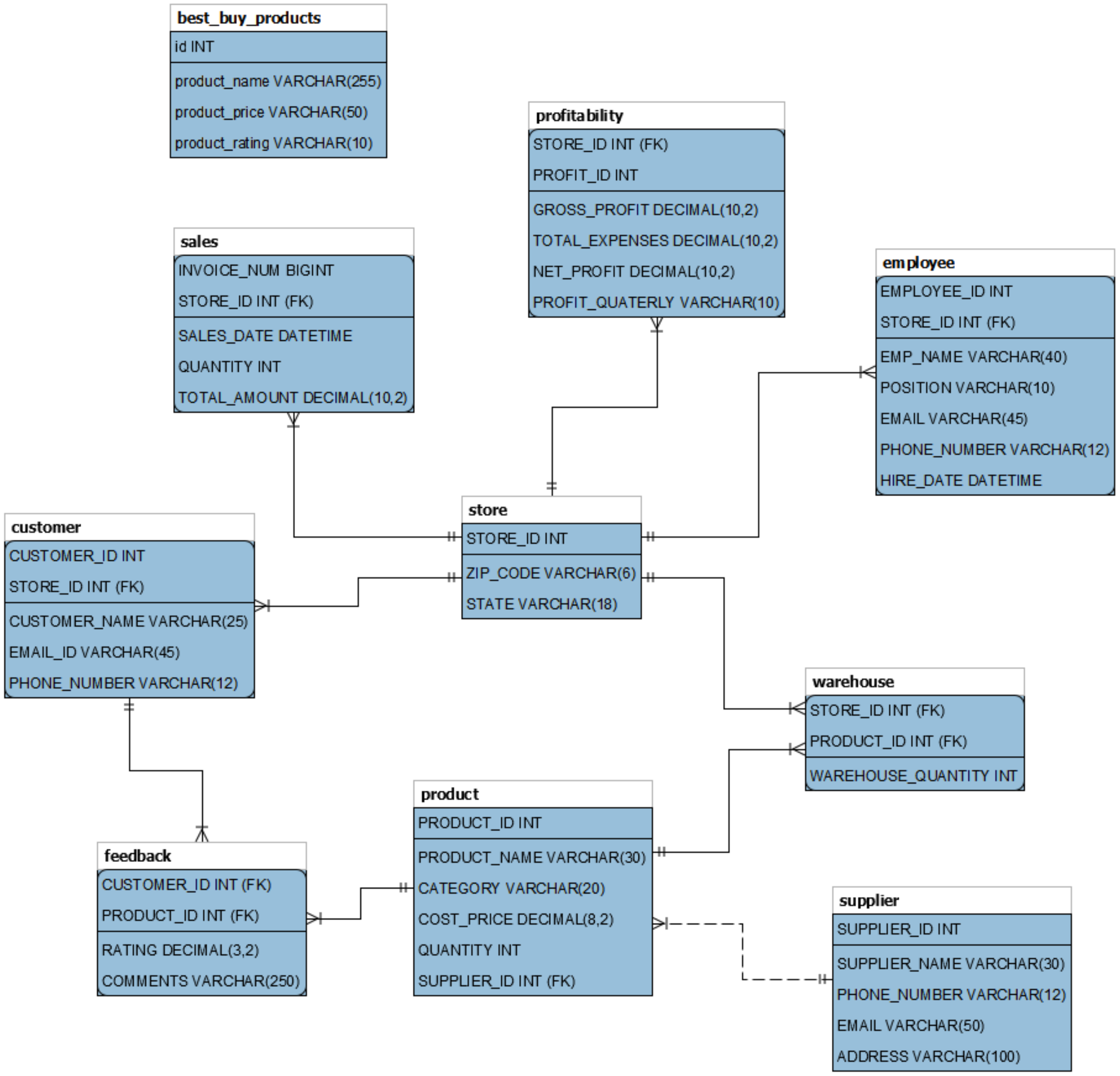
The Python script facilitated web scraping from Best Buy, depicted in the provided screenshot. It extracted product data from the website and stored it in a local database table named "best_buy_products". This automated process streamlined data acquisition from Best Buy's platform, enhancing efficiency in gathering product information. The script's functionality enabled seamless integration of scraped data into the database, facilitating further analysis or utilization as needed. Overall, the implementation showcased effective utilization of Python for web scraping and database management, exemplifying the script's practical application in data retrieval tasks.



```
# Checking Data
USE mm_team04_02;
SELECT * from best_buy_products;
```

id	product_name	product_price	product_rating
76	Apple - iPhone 15 Pro Max 1TB - Black Titanium ...	$44.45	4.7
77	Apple - iPhone 15 Pro Max 1TB - Natural Titaniu...	$44.44	4.4
78	Apple - iPhone 14 Plus 128GB - Midnight (Verizon)	$23.05	4.9
79	Apple - iPhone 14 128GB - Blue (AT&T)	$20.28	4.7
80	Apple - iPhone 14 Plus 128GB - Midnight (AT&T)	$23.06	4.7

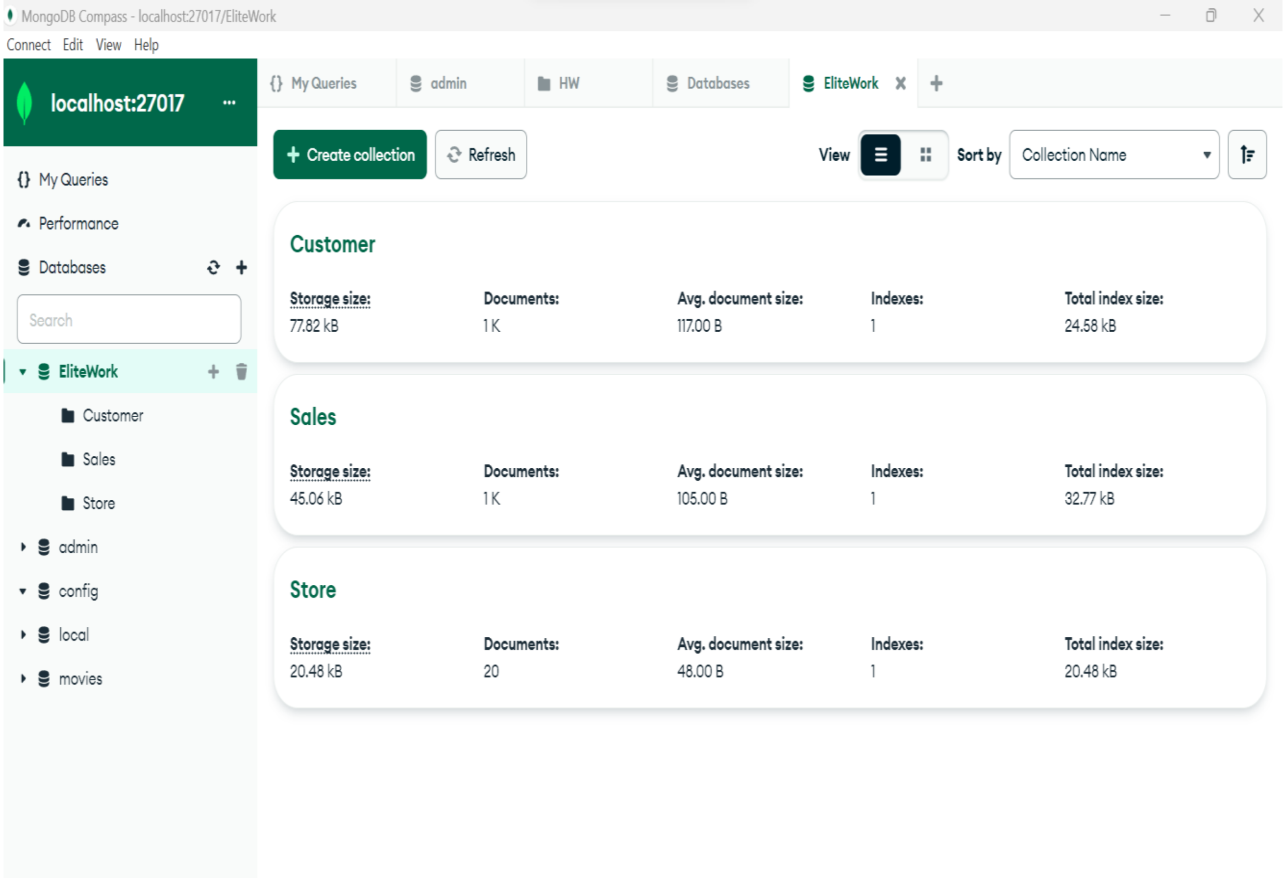
Here is the Model of our database where you can observe that the table from Best Buy has been integrated into the database.



Part 2:

Moving Data from Relational Database to MongoDB:

Through Python code, we successfully transferred data from a Relational Database to MongoDB, as evidenced by the provided screenshot. The migration encompassed records from three tables—Customer, Sales, and Store—contained within the EliteWork database. This seamless data transfer demonstrates the versatility of Python in bridging dissimilar database systems, enabling smooth transition and integration of relational data into MongoDB's document-oriented structure. The screenshot illustrates the presence of two tables within the EliteWork database post-transfer, underscoring the effectiveness of the Python script in facilitating cross-database data movement, a critical aspect of modern data management workflows.



Customer

Storage size:	Documents:	Avg. document size:	Indexes:	Total index size:
77.82 kB	1 K	117.00 B	1	24.58 kB

Sales

Storage size:	Documents:	Avg. document size:	Indexes:	Total index size:
45.06 kB	1 K	105.00 B	1	32.77 kB

Store

Storage size:	Documents:	Avg. document size:	Indexes:	Total index size:
20.48 kB	20	48.00 B	1	20.48 kB

Part 3:

Moving Data from CSSQL Database to Amazon RDS Database

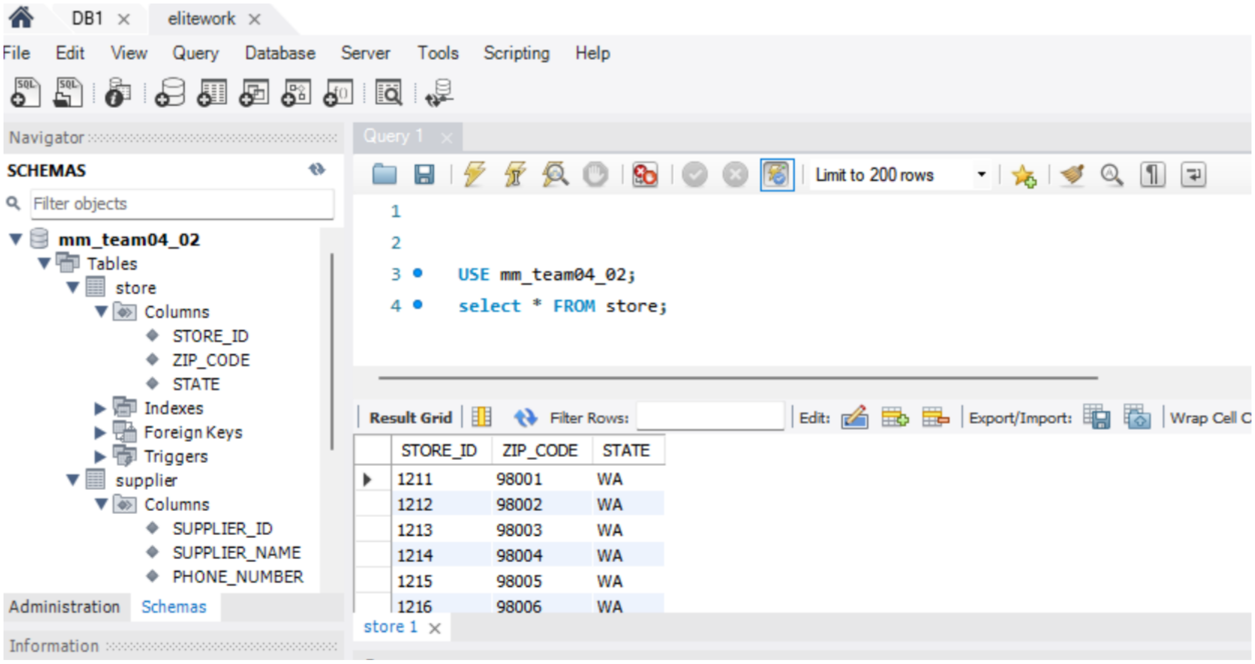
Utilizing a Python script, we established a connection from CSSQL to an Amazon RDS Database, facilitating the seamless transfer of two tables—Store and Suppliers—from the mm_team04_02 database. The process was executed under the database connection name "elitework".

The AWS RDS Database credentials provided were as follows:

Username: admin

Password: Bhusal9891t

This successful data migration exemplifies the script's efficacy in bridging disparate database environments and underscores its utility in facilitating smooth data transfers between CSSQL and Amazon RDS Database, thereby enhancing data management capabilities.



```
USE mm_team04_02;
select * FROM store;
```

STORE_ID	ZIP_CODE	STATE
1211	98001	WA
1212	98002	WA
1213	98003	WA
1214	98004	WA
1215	98005	WA
1216	98006	WA