Entity-Relationship Model for Terrible Amazon Clone

Joseph Edradan | March 17, 2020

Credit_Card

FK	User_ID	INTEGER
	Credit_card_number	INTEGER
	Credit_card_cvn	INTEGER
	Expiration_date	DATE

Prime_member

PK	Prime_ID	INTEGER
FK	Customer_ID	INTEGER

Prime_payment

FK	Prime_ID	INTEGER
	Payment_date	INTEGER
	Amount	INTEGER
	Paid	BOOLEAN

Customer

FK	Customer_ID	INTEGER
	First_Name	VARCHAR(255)
	Last_Name	VARCHAR(255)
	Has_credit_card	BOOLEAN
	Has_Prime	BOOLEAN

User

PK	User_ID	INTEGER
	Username	VARCHAR(30)
	Password	VARCHAR(30)

Seller

FK	Seller_ID	INTEGER
	Seller_name	VARCHAR(255)
	Seller_rating	TINYINT UNSIGNED

Company

PK	Company_ID	INTEGER
	CEO	VARCHAR(255)

Manufacturer

FK	Manufacturer_ID	INTEGER
FK	Company_ID	INTEGER
	Manufacturer_name	VARCHAR(255)

Distributor

FK	Distributor_ID	INTEGER
	Distributor_name	INTEGER
	Distributor_rating	TINYINT UNSIGNED

Contact_information

FK	User_ID	INTEGER
	Country	VARCHAR(255)
	Address_1	VARCHAR(255)
	Address_2	VARCHAR(255)
	City	VARCHAR(255)
	State_Province	VARCHAR(255)
	Zip	INTEGER
	Phone_1	VARCHAR(20)
	Phone_2	VARCHAR(20)
	Email	VARCHAR(255)

Super_order

PK	Order_ID	INTEGER
FK	Customer_ID	Type

Shopping_cart_order

FK	Order_ID	INTEGER

Order_item

FK	Order_ID	INTEGER
FK	Product_listing_ID	INTEGER
	Quantity	TINYINT UNSIGNED

Order

FK	Order_ID	INTEGER
FK	Shipping_ID	INTEGER
	Order_notes	VARCHAR(255)

Seller_product_listing

PK	Seller_product_listing_ID	INTEGER
FK	Seller_ID	INTEGER
FK	Product_ID	INTEGER
	Product_pricing	INTEGER

Shipping_Information

PK	Shipping_ID	INTEGER
FK	Distributor_ID	INTEGER
	Shipping_speed	VARCHAR(255)
	Order_date	DATE
	Ship_date	DATE
	Arive_date	DATE
	Delivery_notes	VARCHAR(255)

Product

PK	Product_ID	INTEGER
FK	Manufacturer_ID	INTEGER

Product_information

FK	Product_ID	INTEGER
	Product_name	VARCHAR(255)
	Product_description	TEXT
	Product_MSRP	INTGER
	Product_rating	TINYINT UNSIGNED
	Product_image	BLOB

Product_comments

FK	Product_ID	INTEGER
FK	User_ID	INTEGER
	Date_created	DATE
	Date_modified	DATE
	Comments	TEXT
	Images	BLOB

Product_Tag_Pair

FK	Product_Tag_ID	INTEGER
FK	Product_ID	INTEGER

Tag_Infomration

PK	Product_Tag_ID	INTEGER
	Tag_Title	VARCHAR(255)

Has
Can have a
Has a
Is a
Has a
Can have
Has many
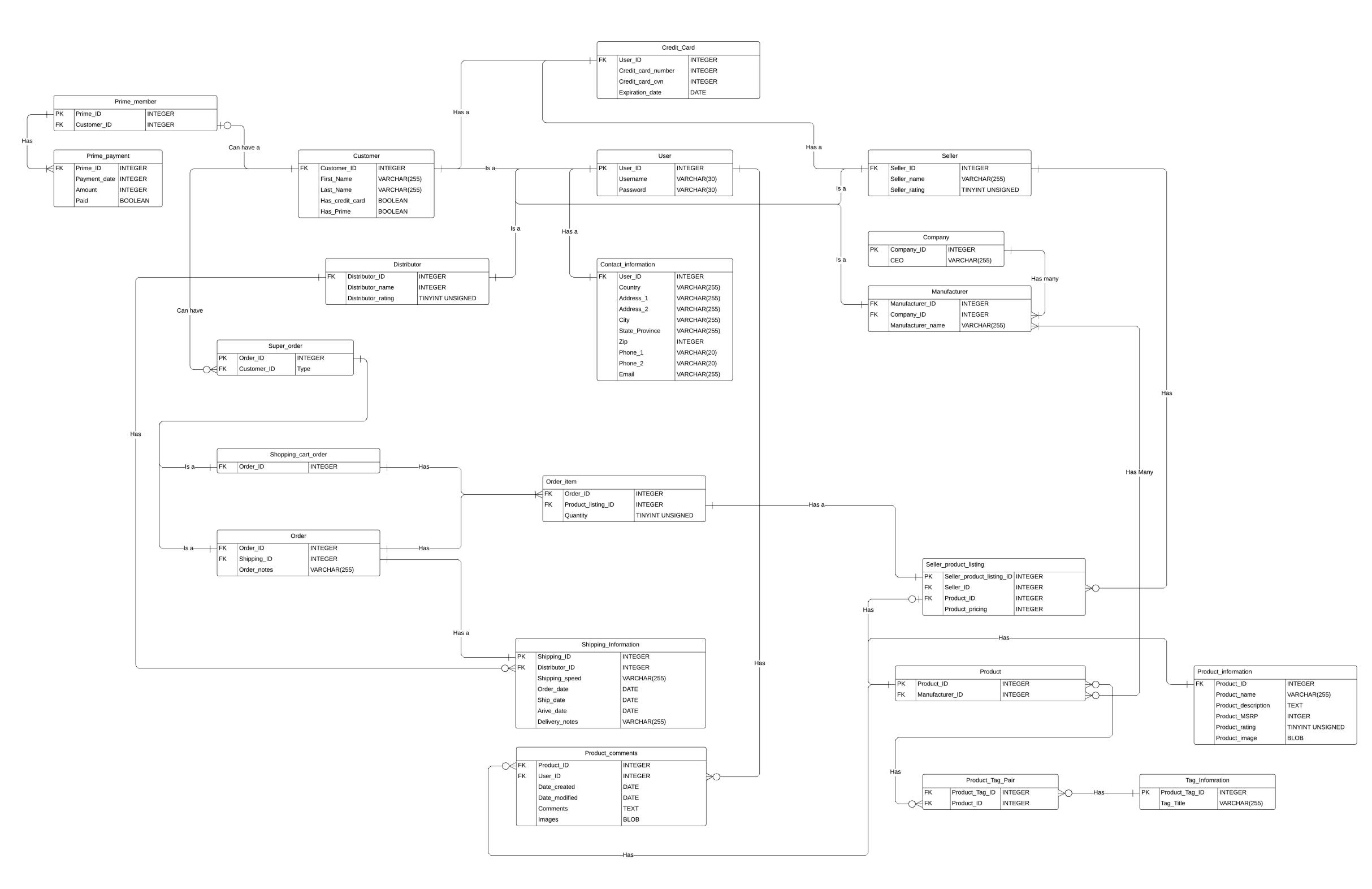Is a
Has a
Has
Has Many
Is a
Has
Has a
Has
Has
Has a
Has
Has
Has

Joseph Edradan
3/17/20
CIS-363-OLH-Enterprise Data Mgmt. w MySQL-CRN 42508

Enterprise Database Management with MySQL Personal Collection Database
Application (PC-DBA)
<u>Phase 1: Proposal and Entity-Relationship Model</u>

       I was going to work on the **Novel Corona Virus 2019 Dataset** given on Kaggle.com, but this dataset does not have a lot of relationship sets and would require formatting the csv files and using a lot of regex for outputting and analyzing the data. As an alternative, I will be making a *Terrible Amazon Clone* which should be an example of how not to use a database but should have a decent number of entity and relationship sets.

       My *Terrible Amazon Clone* should be a rip off of Amazon in terms of item listing, orders, and shipping. It even has Amazon Prime. The Entity-Relationship diagram shows the various connections between the tables. Starting with the Users, there are the Customers, Sellers, Distributors, and the Manufacturers. They should all be apart of the Users table. The Customers and the Sellers both should have credit cards as they are the ones who do monetary transactions. The Credit_card and User tables should obviously be unsafe as they do not have salted and hashed values which I may need to write an algorithm for if I were to change it to be safe. The Customers can have Prime and can make Orders via Shopping_cart_order or Order table. On the other hand, the Manufacturers can make the same product under the same company name and can make multiple products. If an Order is made, then it will contain Shipping information of both the Customer and Seller and a unique product made by a Manufacturer and sold by the Seller. The product itself is simplified by being split into multiple relationship set to minimize table size. Those relationship sets are Product_information, which is self-explanatory, Product_tag_pair which is tagging for finding similar products, and Product_comments which are made by the Users on the product page of the product.

The difficulty with the project would probably be:
1. Generating data for the database which I would write a python script for.
2. Making nice html pages to display and do things.