

Part 1 Reviews:

- “Please fix the ER Diagram so the relationship lines don't cross one another, it just looks cleaner and easier to read. The relationship line for Products to SalesOrders should have the many symbol on both ends.”
 - Done.
- “The overview here is fairly short and doesn't give a ton of information; however, it does list how much the store makes in sales. It could be a little more specific by listing the time period in which it makes “1 million gold”, adding how large of an inventory the store has at any time, and perhaps adding information about how many employees the store employs.”
 - Added information.
- “The entity/ attribute capitalization is consistent. Entity names are plural and attribute names are singular.”
 - Thank you!
- “A total float is used within SalesOrders for totalPrice and as far as I'm aware we're supposed to utilize decimal because of rounding errors associated with float.”
 - Modified as suggested.
- “Within SalesOrders there is auto_increment listed for customerID and employeeID which they shouldn't be since they're an FK and get assigned auto_incrementing IDs in their respective tables and both of those don't have a data value associated with them.”
 - Fixed.
- “Within SalesOrders productID is missing a data type and not Null.”
 - Fixed.
- “Could more deeply describe the constraints such as “varchar” could be “varchar(50)” and the racial datatypes could be set as specifically either Orc/Elf/Human to be more constrained and ensure there aren't data errors such as human versus Human or orc versus Ork.”
 - Varchar length constraints added.
 - Races and item rarities will be dropdown options in the interface and filtered server-side before being added.

Part 2 Reviews:

Hello, Group 115!

Great job. Below are my thoughts.

Does the schema present a physical model that follows the database outline and the ER logical diagram exactly? If yes, summarize what you see. If not, what changes would you suggest be made?

Currently, you have a mix of an ER diagram and a Schema. For it to be considered a Schema you would need to have all the attributes and the intersection table. I would suggest adding this then updating the information in your outline.

Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming? If yes, summarize what the group uses for each of the above. If not, what changes would you suggest be made for improved consistency?

a) There is an error in your ER diagram in the Employees table. You put customerRace instead of employeeRace.

b) Yes, all your entities were plural and your attributes were singular.

c) Capitalization was consistent. Your entities are capitalized and your attributes use camel case.

Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)? If yes, summarize what makes the schema clear and easy to read. If not, what changes would you suggest to improve the schema's readability?

There is no schema. I would add a schema to improve the schema's readability.

Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)? If yes, summarize the components you see making up the properly

Jason Fanger

Marcos Fabian

formed intersection tables/M:N relationship. If not, what changes would you suggest to improve them?

You did not add any intersection table. I would add an intersection table between SalesOrders and Products.

Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies? If yes, what are the issues, and what can be done to improve upon them? If not, summarize how the sample data shows normalization.

The data does not suggest any non-normalized issues. Each entity holds non-repeating data, a primary key, and no transitive dependencies. Good job!

Is the SQL file syntactically correct? This can be easily verified by using phpMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!) If yes, summarize what you see upon importing (e.g. any status message, which tables were imported AND populated with their corresponding sample data, etc.). If not, what happened with the import/what appears to be the issue(s) with the file that are causing this?

The import is getting an error caused by extra comma and no commas at different places where you insert data. Make sure to have a comma between each of the values you are inserting.

In the SQL, are the data types appropriate considering the description of the attribute in the database outline? If yes, summarize why they seem fitting. If not, what changes would you suggest based on the attribute descriptions?

Yes, the data types are appropriate. You have names, races, and types as varchar and IDs, prices, and amounts as int.

In the SQL, are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared? If yes, summarize how they are correctly defined, and where CASCADE operations are declared. If not, what changes should be made to the key definitions and/or what CASCADE operations would you suggest declaring?

Jason Fanger

Marcos Fabian

All the primary and foreign keys are correctly defined. However, you have no CASCADE operations declared. You might want to add some operations to protect your data if the following scenario happens, You delete an Employee or Customer but this leaves your SalesOrder table without a customer/employee ID. You can either set it up that once an Employee or Customer is deleted all their sales are deleted too or you set it up that an Employee or Customer can't be deleted if they have a sales order. This same scenario also applies to products connected to a SalesOrder.

In the SQL, are relationship tables present when compared to the ERD/Schema? If yes, summarize the relationship tables present in both the SQL and the ERD/Schema. If not, which relationship tables aren't present?

You have no intersection table between the SalesOrders and Products. However, the rest of your entity tables (Products, SalesOrders, Customers, and Employees) are present in the SQL. The Employees table is connected to the Customers table with the use of foreign key and SalesOrders table is connected to the Customers, Products, and Employees tables.

In the SQL, is all sample data shown in the PDF INSERTED? If yes, summarize which tables in the SQL have the sample data from the PDF inserted. If not, what sample data is missing or inconsistent?

You have no data in the PDF to compare to the SQL, but the SQL is missing data for your SalesOrder table. Also, you have no data inserted into the employeeID column for the Customer table. This should be impossible because in your outline you say this column is Not Null.

Is the SQL well-structured and commented (e.g. hand authored) or not (e.g. exported from MySQL)? If yes, summarize the structure and comments you see. If not, what changes would you suggest to improve the SQL file?

The SQL document is well-structured and hand authored. However, you have no comments added. I would suggest adding some comments to increase the readability.

Extra Notes:

I would consider separating the name to two columns in the employee and customer tables.

Jason Fanger

Marcos Fabian

When you do complete your data make sure that you are adding in all of the scenarios that can happen. Such as showing that product is not apart of any sales orders (since you made this optional). This would apply for any other tables you made optional.

Hi group 115, very fun and clever idea! Here's my feedback.

Does the schema present a physical model that follows the database outline and the ER logical diagram exactly?

If yes, summarize what you see. If not, what changes would you suggest be made?

At the moment, I do not see a proper schema represented. I see an ER diagram with some of the attributes listed, but not all, (ex. productID: int, auto_increment, unique, not NULL, PK) and I do not see an intersection table facilitating the M:N relationship between SalesOrders and Products. I would suggest adding the intersection table and including all attributes.

Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

If yes, summarize what the group uses for each of the above. If not, what changes would you suggest be made for improved consistency?

Naming is consistent throughout the outline and Diagram. The employee table in the ERD has customerRace instead of employeeRace

Yes, entities are plural and attributes are singular.

Capitalization seems consistent.

Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)?

If yes, summarize what makes the schema clear and easy to read. If not, what changes would you suggest to improve the schema's readability?

Lines are not crossed and the diagram is clear and readable. The image quality is satisfactory to be able to read all of the provided info, however as mentioned above some of the schema is missing.

Jason Fanger

Marcos Fabian

Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)?

If yes, summarize the components you see making up the properly formed intersection tables/M:N relationship. If not, what changes would you suggest to improve them?

The intersection table facilitating the M:N relationship between SalesOrders and Products is missing.

I would suggest adding this intersection table.

Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies?

If yes, what are the issues, and what can be done to improve upon them? If not, summarize how the sample data shows normalization.

There was no sample data provided in the PDF document in excel format. There is sample data provided in the DDL.sql. In the DDL.sql, data for the SalesOrders table is missing. I would add in data for this table as well as writing the code for the intersection table to get a clear view of how all of this data interacts.

Is the SQL file syntactically correct? This can be easily verified by using phpMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!)

If yes, summarize what you see upon importing (e.g. any status message, which tables were imported AND populated with their corresponding sample data, etc.). If not, what happened with the import/what appears to be the issue(s) with the file that are causing this?

There are some commas missing in your INSERT statements after VALUES , there should be commas between the values. There is also one "Values" at the end of the file that is not capitalized while the others are.

In the SQL, are the data types appropriate considering the description of the attribute in the database outline?

If yes, summarize why they seem fitting. If not, what changes would you suggest based on the attribute descriptions?

Jason Fanger

Marcos Fabian

The only questions I have with data types are in the customerRace field in Customers and employeeRace in Employees. I'm wondering if it would make more sense to have this data type as enum if you only have so many options for races (limited to orcs, elves, humans). Other than that the data types match what was laid out in the Outline and ERD.

In the SQL, are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared?

If yes, summarize how they are correctly defined, and where CASCADE operations are declared. If not, what changes should be made to the key definitions and/or what CASCADE operations would you suggest declaring?

Customers- PK and FK matches ERD

Employees- PK matches ERD

Products- PK matches ERD

SalesOrders- Transition table needed to manage these FKS to the Products table.

There are no CASCADE operations. I would consider adding cascade options to the Customers and Employees tables for example so that if the data changes the effects will propagate to update the data elsewhere.

In the SQL, are relationship tables present when compared to the ERD/Schema?

If yes, summarize the relationship tables present in both the SQL and the ERD/Schema. If not, which relationship tables aren't present?

All relationship tables which are present in the SQL which were outlined in the ERD:

SalesOrders, Products, Customers, Employees.

In the SQL, is all sample data shown in the PDF INSERTED?

If yes, summarize which tables in the SQL have the sample data from the PDF inserted. If not, what sample data is missing or inconsistent?

There was no sample data in the PDF. There is sample data in the DDL, however I have nothing to compare it to.

Jason Fanger

Marcos Fabian

Is the SQL well-structured and commented (e.g. hand authored) or not (e.g. exported from MySQL)?

If yes, summarize the structure and comments you see. If not, what changes would you suggest to improve the SQL file?

The SQL is well structured, there are no comments. I would add some comments when you add your transition table to further demonstrate how it facilitates joining the two tables. I would also add comments to your sample data and CASCADE operations so that it's clear what the intended affect will do.

Hello Jason and Marcos,

Does the schema present a physical model that follows the database outline and the ER logical diagram exactly?

You guys have a great ERD, but your PDF has no schema. I would generate a schema to insert after the ERD that includes your intersection table and all attributes of a table/entity.

Is there consistency in a) naming between overview, outline, ER and schema entity/attributes b) entities plural, attributes singular c) use of capitalization for naming?

Yes, all entities are plural, attributes are singular, and the use of capitalization is correct.

Is the schema easy to read (e.g. diagram is clear and readable with relationship lines not crossed)?

There is no schema in your PDF. I recommend tossing that in there once you have it generated and looking good.

Are intersection tables properly formed (e.g. two FKs and facilitate a M:N relationship)?

You don't have any intersection tables in your ERD. I would recommend adding one for SaleOrders and Products in your ERD and have it represented in your schema once that is created.

Jason Fanger

Marcos Fabian

Does the sample data suggest any non-normalized issues, e.g. partial dependencies or transitive dependencies?

I don't see any non-normalized issues in your outline. Every table has a PK, I'm not noticing any transitive dependencies, and there doesn't seem to be repeating data in any of the tables. I don't think it's necessary to add the normalization comments after each table's outline in the PDF though. That's just my opinion though, not a recommendation.

Is the SQL file syntactically correct? This can be easily verified by using phpMyAdmin and your CS 340 database (do not forget to take backup of your own database before you do this!)

No, it is not. There are several syntax errors mainly related to commas and comma placement. For example, when inserting data,

VALUES

('Joseph Carlson', 'Adventurer', 'Human')

('Emma Rose', 'Merchant', 'Elf')

('Grumly Rockstone', 'Miner', 'Dwarf')

('Grog Skullcrusher', 'Mercenary', 'Orc')

Add commas behind each row/list of data like so:

VALUES

('Joseph Carlson', 'Adventurer', 'Human'),

('Emma Rose', 'Merchant', 'Elf'),

('Grumly Rockstone', 'Miner', 'Dwarf'),

('Grog Skullcrusher', 'Mercenary', 'Orc')

Jason Fanger

Marcos Fabian

In the SQL, are the data types appropriate considering the description of the attribute in the database outline?

Yes, data types correspond with the correct attribute and description from the outline. This also matches the ERD.

In the SQL, are the primary and foreign keys correctly defined when compared to the Schema? Are appropriate CASCADE operations declared?

You haven't included any CASCADE operations in your SQL. I'd recommend adding those to keep your database updated correctly and reduce anomalies or errors in data entry or removal. Your FK's and PK's are all correctly defined in the outline, ERD, and SQL though!

In the SQL, are relationship tables present when compared to the ERD/Schema?

All your main tables are there (Customers, Employees, SalesOrders, and Products) but you don't have an intersection table to show the relationship between SalesOrders and Products. All other relationships are shown in the tables through FK's.

In the SQL, is all sample data shown in the PDF INSERTED?

Adding some sample data tables in your PDF would be great. It's kind of necessary to show sample data in the outline so we can get a grasp of what each table would look like without looking at a schema or the SQL itself. There's no data input for the SalesOrders table.

Is the SQL well-structured and commented (e.g. hand authored) or not (e.g. exported from MySQL)?

I think it's well structured but there are no comments. I don't believe these queries require super detailed comments, but some brief relationship explanations or just title comments for each table would be good. It seems to be hand-authored.

I think there is some work to be done to clean up, correct syntax errors, and improve the overall readability of the SQL but the base is there. Just needs some polishing and minor additions.

Jason Fanger

Marcos Fabian

PART 3 REVIEWS:

Does the UI show where it will utilize a SELECT for every table in the schema?

No there is no utilization of a SELECT for every table in the schema. The tables Customers, Employees, Products, Sales, and Order and its attributes are displayed correctly. But there is no select specific part of the entity itself. For the final draft, make sure to implement this correctly.

Does at least one form utilize a search/filter/dropdown with a dynamically populated list of properties?

The page does not seem to have included a search, filter, or dropdown with dynamically populated lists for properties. There are no features of filtering or dynamically loading lists for the properties of customers, employees, or products. Places where this could be implemented would be customer filter to add a dropdown or search bar to select customers from the Customers table to filter by customer name or ID. Or to add a product search to search for item type, price range, etc.

Does each INSERT also add the corresponding FK attributes, including at least one M:M relationship?

No, the page does not seem to include INSERT process that adds corresponding foreign key attributes that handles M: M relationship. INSERTs that need to be altered would be when adding a Sale order to be able to add a Transaction ID, this is a foreign key referencing the SalesOrders table. As Well as when adding a product, to be able to enter a ProductID, this is a foreign key referencing the Products table. These are the foreign key attributes in the OrderProducts table that acts as the intersection table for the M:M relationship.

- ***Does the UI show where it will utilize a SELECT for every table in the schema? In other words, a data table for each table in the schema should be displayed on the UI (which are not required to be populated with sample data, but should at least have column names). If yes, which tables from the schema do you see fully represented in the UI with a SELECT? If not, which tables and/or attributes are missing? Note: it is generally not acceptable for just a single query to join all tables and display them.***

For each table like Customers, Employees, Products, Sales, and Orders, the attributes are displayed. There's no select or filter parts of any single entity. The tables show up with data, but there's no direct SELECT option to narrow down to specific columns or values.

- ***Does each INSERT also add the corresponding FK attributes, including at least one M:M relationship? In other words if there is a M:M relationship between Orders and Products,***

Jason Fanger

Marcos Fabian

INSERTing a new Order (e.g. orderID, customerID, date, total), should also INSERT row(s) in the intersection table, e.g. OrderDetails (orderID, productID, qty, price and line_total). Or, alternatively, there should be an INSERT for INSERTing into the intersection table(s) directly. If yes, list all the table INSERTs that correctly add their corresponding FK attributes, and describe the group's implementation for INSERTing into the intersection of their M:M relationship. If not, which INSERTs need to be altered, and in what way?

No, it's not fully there. For example, if you're adding a new Sales Order, it doesn't ask for the foreign key related to the Product. It seems to only ask for info specific to the Sales Order itself without linking back to the Product, which is essential in a many-to-many relationship setup.

- ***Does at least one form utilize a search/filter/dropdown with a dynamically populated list of properties? If yes, which form(s) have which features incorporated? For which attribute(s)? If not, where are a couple places this could be implemented?***

I don't see any search, filter, or dropdown functionality that dynamically populates based on the tables. None of these are added to the UI for any of the tables in the website, so that's missing here.

Is at least one relationship NULLable? In other words, there should be at least one optional relationship, e.g. having an Employee might be optional for any Order. Thus, it should be feasible to edit an Order and change the value of Employee to be empty. If yes, which NULLable relationship(s) do you see, and does it seem to make sense? If not, where would you suggest adding a NULLable relationship and why?

No, based on their database outline and my interpretation of their website, each relationship is required and none can be NULL. For example, the Loans table has FKs for book and customer ID's. Neither can be NULLable. I feel that this is appropriate for the table, however it does leave the requirement for one FK to be NULLable left unsatisfied.

Changelog:

Jason Fanger
Marcos Fabian

1. Updated ERD as the TA suggested. Made lines more clear and updated the relationship between *SalesOrders* and *Products* to be M:M.
2. Updated Price and totalPrice attribute from float to decimal, as one reviewer suggested.
3. Updated all varchar attributes to have sizes.
4. Updated description to be more detailed. Removed *Inventory* from the description as that was not used in our final design.
5. Updated CustomerID and Product ID, removed auto_increment as they are FK, added int as data type.
6. Changed *Employee* participation to be optional instead of mandatory in relation to *SalesOrders* as suggested by the TA. It's possible a *SalesOrder* may not require an *employee*, therefore the FK can be set to NULL when using UPDATE. Updated ERD to reflect this change.
7. Included Normalization check in each table description.
8. Added ItemName attribute to Products. Need to update on ERD and SQL
9. Added CASCADE operations to primary and foreign keys.
10. Created OrderProducts as an intersection table to show the M:M relationship between SalesOrders and Products. Updated the SQL and the ERD to reflect these changes.
11. Added Sample Data Screenshots
12. Created DML file and deleted Insert queries from DDL file
13. Added Search and dropdown options to HTML
14. Updated Sales Order HTML so the user can input FKs into the table via dropdown to prevent errors
15. Created OrderProducts page
16. Updated PDF to reflect changes made in DDL file
17. Created Update Form for each table in HTML, ideally will implement the update function with JS

Jason Fanger

Marcos Fabian

Proposal for Cast Pro Shop Management System:

Cast Pro Shop is a general store with an average of 15 employees at any given time that sells 100,000 gold in spells, potions and other items a year. The store generally stocks a large quantity of rope, torches, rations, tools, and miscellaneous camping supplies while occasionally adding various low-cost magical items such as Bags of Holding when one turns up for a relatively low price.

The recent war between the elves and the orcs has increased the amount of customers they receive each day and are looking to create a database to keep track of their day to day operations. They are predicting up to 1 million gold in sales this year. They require the database to keep track of *SalesOrders* of *Products* to *Customers*. The recent war between the elves and the orcs also requires them to keep track of *Employees* to ensure safety between the two races. By keeping track of an employee's information, managers can make sure the two races are not on the same schedule.

With the addition of the database, Cast Pro Shop aims to improve customer service while improving employee efficiency in order to increase sales.

Customers: records the details of customers we do business with.

- customerID: int, auto_increment, unique, not NULL, PK
- customerName: varchar(64), not NULL
- customerType: varchar(16), not NULL (adventurer, merchant, townsfolk, etc.)
- customerRace: varchar(16), not NULL (human, orc, elf, etc.)
- employeeID: int, FK
- Relationships:
 - A 1:M relationship between *Customers* and *SalesOrders* is implemented with customerID as a FK inside of *SalesOrders*.
 - A 1:M relationship between *Employees* and *Customers* is implemented with employeeID as a FK inside of *Customers*. One *Employee* is dedicated to many *Customers* to avoid racial conflict and promote friendliness.
- Normalization:
 - Table is N1 because it has only one primary key and there are no repeating groups.
 - Table is N2 because each non-primary attribute is dependent on the primary key. *customerName*, *customerType*, etc all depend on the customerID.
 - Table is N3 because there are no transitive dependencies in each non-PK attribute. Each non-PK attribute is ONLY dependent on the primary key.

Jason Fanger

Marcos Fabian

Employees: records the details of employees we hire.

- employeeID: int, auto_increment, unique, not NULL, PK
- employeeName: varchar(64), not NULL
- employeeRace: varchar(16), not NULL (human, orc, elf, etc.)
- Relationships:
 - A 1:M relationship between *Employees* and *SalesOrders* is implemented with employeeID as a FK inside of *SalesOrders*. One *Employee* can process many *SalesOrders*.
 - A 1:M relationship between *Employees* and *Customers* is implemented with employeeID as a FK inside of *Customers*. One *Employee* is dedicated to many *Customers* to avoid racial conflict and promote friendliness.
- Normalization:
 - Table is N1 because it has only one primary key and there are no repeating groups.
 - Table is N2 because each non-primary attribute is dependent on the primary key. *employeeName*, *employeeType*, etc all depend on the employeeID.
 - Table is N3 because there are no transitive dependencies in each non-PK attribute. Each non-PK attribute is ONLY dependent on the primary key.

SalesOrders: records the details of the sales made to the customers.

- transactionID: int, auto_increment, unique, not NULL, PK
- customerID: int, not NULL, FK
- employeeID: int, FK
- orderDate: date, not NULL
- totalPrice: decimal, not NULL
- Relationships:
 - A 1:M relationship between *Customers* and *SalesOrders* is implemented with customerID as a FK inside of *SalesOrders*. One *Customer* can submit many *SalesOrders*.
 - A 1:M relationship between *Employees* and *SalesOrders* is implemented with employeeID as a FK inside of *SalesOrders*. One *Employee* can process many *SalesOrders*.
 - A M:M relationship between *SalesOrders* and *Products* is implemented with productID as a FK inside of *SalesOrders*. Many products can be in many sales orders many times.
- Normalization:

Jason Fanger

Marcos Fabian

- Table is N1 because it has only one primary key and there are no repeating groups.
- Table is N2 because each non-primary attribute is dependent on the primary key. *orderDate*, *customerID*, *productID*, etc all depend on the *transactionID*.
- Table is N3 because there are no transitive dependencies in each non-PK attribute. Each non-PK attribute is ONLY dependent on the primary key.

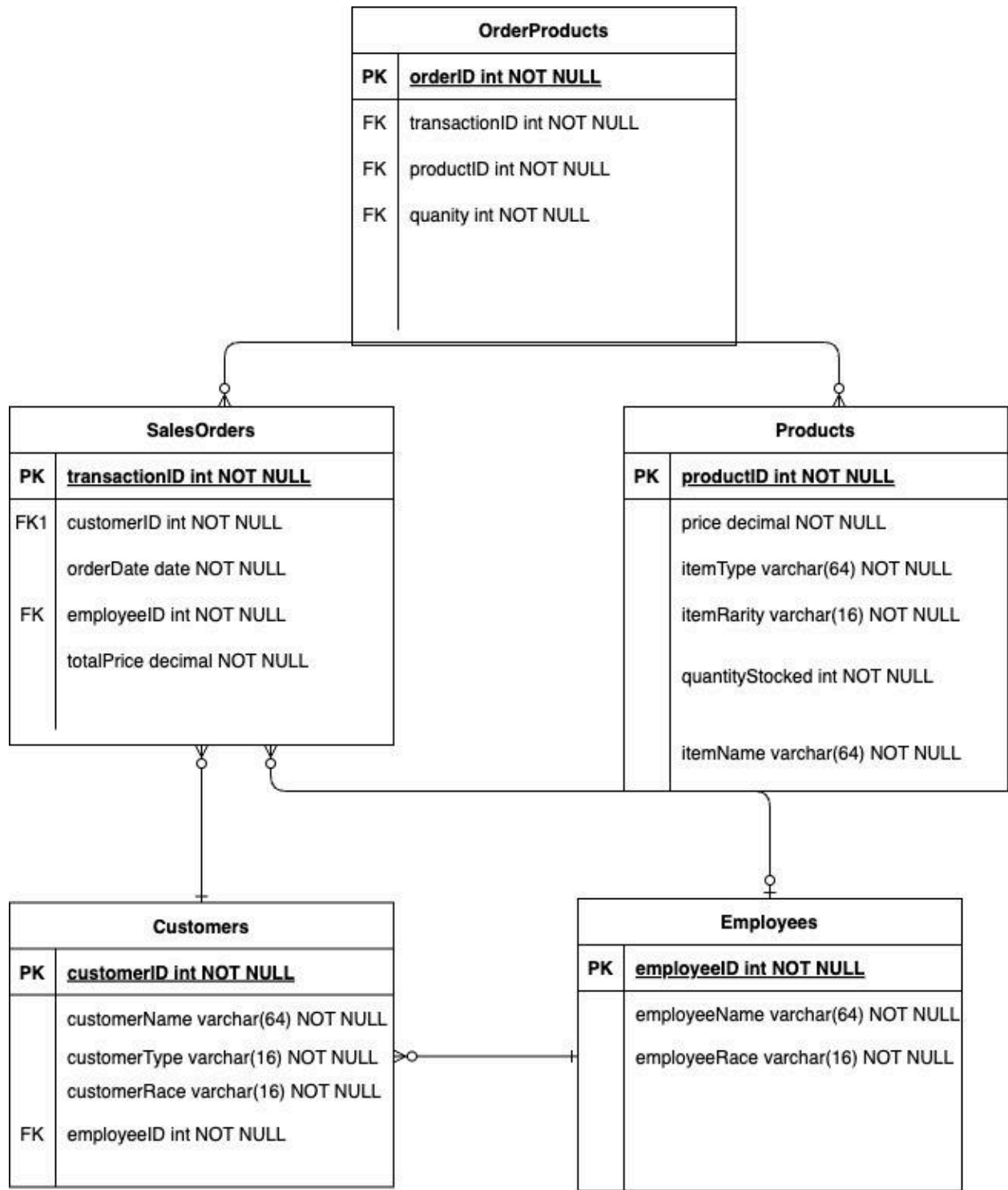
Products: records the details of the items we sell.

- *productID*: int, auto_increment, unique, not NULL, PK
- *itemName*: varchar(64), not NULL
- *itemType*: varchar(64), not NULL (spellbook, armor, weapon, misc.)
- *itemRarity*: varchar(16), not NULL (common, rare, epic, legendary)
- *price*: decimal, not NULL
- *quantityStocked*: int, not NULL
- Relationship: a M:M relationship between *SalesOrders* and *Products* is implemented with *productID* as a FK inside of *SalesOrders*. Many products can be in many sales orders many times.
- Normalization:
 - Table is N1 because it has only one primary key and there are no repeating groups.
 - Table is N2 because each non-primary attribute is dependent on the primary key. *price*, *itemType*, *itemRarity*, etc all depend on the *productID*.
 - Table is N3 because there are no transitive dependencies in each non-PK attribute. Each non-PK attribute is ONLY dependent on the primary key.

OrderProducts: records the details of a transaction with the number of products bought. .

- *OrderID*: int, auto_increment, unique, not NULL, PK
- *Transaction ID*: int, not NULL, FK
- *productID* INT NOT NULL, FK
- *quantity* INT NOT NULL,`
- Relationship: an intersection table for an M:M relationship between *SalesOrders* and *Products*. It is implemented with *productID* as a FK inside of *OrderProducts* and a *transactionID* from *SalesOrders*.











Jason Fanger
Marcos Fabian



Jason Fanger
Marcos Fabian

SAMPLE DATA:

OrderProducts:

		orderProductID	transactionID	productID	quantity
<input type="checkbox"/>	 Edit  Copy  Delete	1	1	1	2
<input type="checkbox"/>	 Edit  Copy  Delete	2	1	3	5
<input type="checkbox"/>	 Edit  Copy  Delete	3	2	2	1

Products:

		productID	price	itemType	itemRarity	quantityStocked
<input type="checkbox"/>	 Edit  Copy  Delete	1	50.00	Hammer	Common	5
<input type="checkbox"/>	 Edit  Copy  Delete	2	25.00	Rations	Common	15
<input type="checkbox"/>	 Edit  Copy  Delete	3	500.00	Sword +1	Rare	1

Employees:

		employeeID	employeeName	employeeRace
<input type="checkbox"/>	 Edit  Copy  Delete	1	Bob	Human
<input type="checkbox"/>	 Edit  Copy  Delete	2	Carlenia	Elf
<input type="checkbox"/>	 Edit  Copy  Delete	3	Bort Rockstone	Dwarf

Customers:

		customerID	customerName	customerType	customerRace	employeeID
<input type="checkbox"/>	 Edit  Copy  Delete	1	Joseph Carlson	Adventurer	Human	NULL
<input type="checkbox"/>	 Edit  Copy  Delete	2	Emma Rose	Merchant	Elf	NULL
<input type="checkbox"/>	 Edit  Copy  Delete	3	Grumly Rockstone	Miner	Dwarf	NULL
<input type="checkbox"/>	 Edit  Copy  Delete	4	Grog Skullcrusher	Mercenary	Orc	NULL