

## **Section 1: Project Introduction**

Can you design, implement, and start using your very own database, all in one course term? You can, even if you have never touched a database in your life! The theoretical concepts and applied skills you learn throughout the course prepare you well for this task, and have been carefully crafted and sequenced for rapid learning. Further, the teaching team stands ready to help fill any gaps, so you are not dependent on the course materials alone. You will undoubtedly put in many hours of hard work on this project, but you will leave the course with your own database, and more importantly, the skills to build other ones.

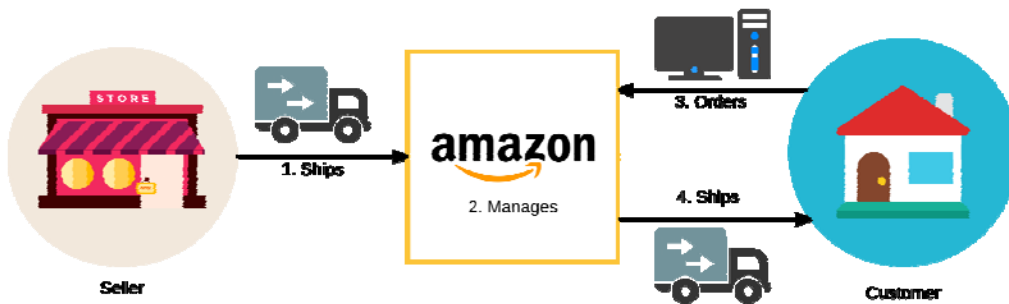
The start-to-finish experience you gain from this project will help prepare you for database work in the industry or academia. If your boss asks you to develop a database, you'll know how and have the experience to back up your confidence. If you decide to pursue database research, you'll have a solid foundation. Take this project to heart. It will pay off!

The remainder of this document provides you important details. Specifically, the next two subsections of this document, "Introduction to Selling on Amazon", and "Use Cases", are a specification describing the core data, requirements, and use cases for selling on Amazon. These describe what you will be capturing in your database. Read through this document carefully, and ask the teaching team if something is unclear. You will be spending a lot of time designing, implementing, and testing a database based off of this specification.

### **Introduction to Selling on Amazon**

Are there any online marketplaces more reputable than Amazon? Probably not. Why not? For one, Amazon has successfully made use of many unique innovations. For another, Amazon has significantly systematized the selling process, structuring a generalized marketplace which virtually any seller can plug into without much difficulty. And for yet another, Amazon has become so large that it can negotiate discounts with international organizations, including significant shipping cost discounts. Simply speaking, other online marketplaces do not compete at this level. Amazon's online marketplace is exceptional.

Amazon's success notwithstanding, sellers on Amazon still need to manage, pack, and ship their own products, and provide their own customer service, right? Wrong. One of Amazon's innovations is Amazon fulfillment; Amazon handles the inventory, orders, shipping, returns, and customer service on behalf of the seller. To plug in to the marketplace, the seller only need deliver the products to one of Amazon's warehouses. Amazon takes over from there. Essentially, with Amazon's marketplace, the roles change – sellers become suppliers, and Amazon becomes the seller. This process looks as follows.



The process makes room for other innovations, perhaps the most effective being rapid and free shipping for Amazon Prime buyers, for any of the seller's products. Individual sellers do not have the clout to make this possible by negotiating shipping discounts and arranging special shipping processes with national and international shipping companies, and they do not need to, because Amazon does so on their behalf. Buyers prize this option, making the seller's products more attractive. Why wait a week or more to receive what you have purchased when you can receive it in two days? And this is made even better by the fact that you can purchase virtually any kind of product.

The relationship between Amazon and sellers is synergistic. Amazon cannot produce the wide variety of products created by sellers worldwide, yet has a superior online marketplace to sell these products effectively. Sellers cannot individually provide such an effective online marketplace, but can provide the products. Both benefit and profit from this relationship.

## Use Cases

Below are a series of use cases that help describe the operations your database supports.

*New Product Use Case* – This occurs when a seller plans to sell a product it has not sold before.

1. The seller searches Amazon's product list to determine if another seller is already selling the product.
2. If a different seller is already selling the product, a new listing is not required; the seller re-uses the same listing.
3. If the product is not yet sold on Amazon, a new listing is created with the product's name, description, price, and other relevant items. Every product added is linked to a product category (all categories are predefined by Amazon), for example, "Computers", "Electronics", "Appliances", and similar.

*Product Delivery Use Case* – This occurs when a seller sends one or more units of a product to Amazon so that they can be sold.

1. The seller ships one or more units of a product to Amazon's warehouse, along with information that indicates to Amazon what the product is, how many units there are, and the condition (new, used, etc ...).
2. After Amazon receives the product(s), it updates the seller's inventory so that customers can purchase the product.

*New Customer Account Use Case* – This occurs when a customer signs up for an account on Amazon, so they can begin purchasing products.

1. The customer provides Amazon with basic information including a username, an address, phone number, and an email address.
2. Amazon creates an account for the customer, enabling the customer to purchase products.

*Product Purchase Use Case* – This occurs when a customer purchases a product from Amazon that was provided by a seller.

1. The user logs in to Amazon under their account.
2. A customer selects one or more products on Amazon's website. When selecting a product, the customer is actually selecting a particular seller's inventory while doing so, though they might not realize this because the process is seamless on Amazon's website.
3. The customer selects a shipping speed (super saver shipping, standard shipping, two-day, one-day) and finalizes their choices.
4. Amazon decrements the seller's inventory for the products purchased.
5. Amazon creates an order which tracks which customer purchased which products from which sellers.

*Product Shipment Use Case* – This occurs when Amazon ships the products a customer purchased.

1. Amazon packages up the purchased products, and assigns an identifier to package so that it can be tracked.
2. Amazon links the package to the customer's order.
3. Amazon ships the package to the default address linked to the customer's account.
4. Amazon notifies the customer that it has been shipped and provides the customer with the tracking ID.

## Section 2: Project Breakdown

Wondering exactly what content to put into your project? Do not worry. This section describes what you need to know.

### Structural Business Rules

Structural business rules upfront help guide the design throughout the rest of your sections. An example of this type of business rule not related to Amazon is “A car may be driven by many drivers; each driver drives one or more cars.” Each business rule should describe the entities involved, the relationship between the entities, and the optionality and plurality constraints for each entity.

### Use Case Driven Aspects

The use cases described in Section 1 drive what tables, stored procedures, and queries are needed for your database. There are five aspects of your database you will develop based upon these use cases, and each aspect is based on a single use case. Each aspect requires tables, a stored procedure, and a query for full implementation. The stored procedure for each aspect should be reusable through use of parameters. For example, the stored procedure for Aspect 1 should use at least four parameters for the following elements – product name, product description, product price, and product category. The query for each aspect should be one single query that retrieves precisely the requested information. A complete schema design and implementation addresses these aspects.

*Your database's tables, stored procedures, and queries only need address these aspects.* It is not possible to attempt a full production-capable database for a large enterprise such as Amazon in the time period afforded by this course; such a database would have thousands of tables. Use the information provided in Section 1 as a backdrop, then focus your design on these five aspects, which support the primary and most indispensable areas of the Amazon's online marketplace. Do not include secondary subschemas for areas of the Amazon's marketplace that may be important, but are not part of its core business, such as credit card and payment processing, full-featured address management, marketing, accounting, human relations, and the like; these would unnecessarily bloat the size of your project and are not needed to demonstrate your mastery of course content. Do not cover other businesses supported by Amazon, such as Amazon Web Services, Amazon Video, and Amazon Music. A general guideline is that to effectively create the design for this course, you would have between 8 and 20 tables. This number is not exact and varies according to your specific implementation.

To help prove that your queries work, your tables need to be populated with apt data. In particular, all queries required by the five aspects should return results. Further, the results returned should make manifest the nuances of the query. For example, if a query selects only results matching a condition, there should be some underlying data that meets the condition, and other data that does not meet the condition. Although a production system may well have thousands or millions of rows, it is impractical for you to attempt such volume. Each table only needs a small number of rows with enough variety to help prove out the queries; you can use 10 rows as a rough guideline.

*Aspect 1: New Products* – This aspect is based upon the New Product Use Case in Section 1.

- a. Create the tables, constraints, and data needed to support new products as described in the use case.
- b. Develop a parameterized stored procedure that is used when a seller needs to add any new product.
- c. A seller adds two new products. The first is a floating lamp which floats in air rather than resting on a base. The second is a throwable camera that allows a photographer to throw the camera into the air and take pictures in flight. Invoke the stored procedure twice to add these products, keeping in mind that products have at a minimum a name, description, price, and category.
- d. A seller is considering developing a new electronic product, and requests a list of existing products in the “Computers” or “Electronics” categories that cost \$25 or less. Develop and execute a single query that provides this information.

*Aspect 2: Product Delivery* – This aspect is based upon the Product Delivery Use Case in Section 1.

- a. Create the tables, constraints, and data needed to support product delivery as described in the use case.
- b. Develop a parameterized stored procedure that is used when any seller delivers any product to Amazon’s warehouse.
- c. A seller delivers ten each of the two new products added in Aspect 1 (the floating lamp and throwable camera). Invoke the stored procedure twice to update the inventory of these products for a seller of your choosing.
- d. The seller from b above requests a listing of all of its products that have an inventory of 12 or less. Develop and execute a single query that provides this information (the floating lamp and throwable camera should be among those listed).

*Aspect 3: New Customer Accounts* – This aspect is based upon the New Customer Account Use Case in Section 1.

- a. Create the tables, constraints, and data needed to support customer accounts as described in the use case.
- b. Develop a parameterized stored procedure that is used when any new customer signs up for a new account on Amazon.
- c. People named Carter Channing and Delaney Emerson sign up for new accounts on Amazon. Invoke the stored procedure twice to add these customers.
- d. For research purposes, Amazon requests the last names of customers where there are at least 3 accounts associated with the last name. Amazon would like to see the actual number of accounts associated with those last names. Develop and execute a single query that provides this information.

*Aspect 4: Product Purchases* – This aspect is based upon the Product Purchase Use Case in Section 1.

- a. Create the tables, constraints, and data needed to support product purchases as described in the use case.
- b. Develop a parameterized stored procedure that is used when any customer purchases any product.
- c. Carter (from Aspect 3) purchases two floating lamps (from Aspect 1), and Delaney purchases one throwable camera. Invoke the stored procedure twice, once for each purchase.
- d. The marketing department at Amazon wants to reach out to customers who buy popular products. The department requests the names and addresses of all customers who bought any product that was purchased by at least four people. Develop and execute a single query that provides this information.

*Aspect 5: Product Shipment* – This aspect is based upon the Product Shipment Use Case in Section 1.

- a. Create the tables, constraints, and data needed to support product shipments as described in the use case.
- b. Develop a parameterized stored procedure that is used when Amazon ships any order.
- c. Amazon ships the orders listed in Aspect 4, one to Carter and the other to Delaney. Invoke the stored procedure twice, once for each order.
- d. Here you define your own query. Define a request for information for this aspect that is implemented with either aggregation or with a subquery. Then develop and execute a single query that provides this information.

### **Entity-Relationship Diagrams**

You will create a conceptual and logical Entity-Relationship diagram (ERDs) for this database. Your logical ERD will be mapped to a relational database schema through the use of SQL. The schema should contain tables, primary and foreign keys, and an index. The primary and foreign keys will help enforce the relationships indicated in the logical ERD, and help enforce referential integrity. Your tables should be normalized to BCNF, or accompanied with a justification as to why the table was not normalized to BCNF.

### **Index**

Create and justify an index that is beneficial to at least one query in your implementation. Include screenshots illustrating the creation of the index, along with an explanation as to why the index is beneficial (be specific).

## Section 3: Project Summary

This section wraps up what was described to help summarize what should be included with your project, and how your project will be evaluated.

### Deliverables

Your final project submission will include two files -- a Word document and a zip file. The contents of each are explained below.

#### *Word Document Contents*

Your Word document should contain:

- a. your structural business rules.
- b. your conceptual ERD or EERD.
- c. your logical ERD or EERD.
- d. screenshots of the SQL addressing the five aspects (make sure to include screenshots of the query or stored procedure creation as well as the results of execution).
- e. a screenshot illustrating the creation of the index, along with an explanation as to why the index is beneficial (be specific).

The Word document does not need to include screenshots of other SQL, such as your table creation and data inserts, unless you want to clarify something. All of your SQL will be included in the zip file, so your instructor or facilitator can refer to these scripts for additional details. *Keep in mind that part of the term project grade is exposition, so make sure to use sections and headers to keep the document organized, and to include explanations of any aspects of your project that warrant it.*

#### *Zip File Contents*

Your zip file will contain SQL scripts that enable your instructor or facilitator to re-create your schema and execute your use case queries against the schema. A SQL script is nothing more than a text file that has a “.sql” extension in the filename. The following three scripts should be included, and you may include additional scripts if it makes sense for your implementation. Your zip file should contain:

- a. your “create” script which contains the SQL DDL to create the tables, constraints, and stored procedures.
- b. your “insert” script which inserts data into the tables.
- c. your “aspect” script which contains the queries and stored procedure invocations for each of the 5 aspects.

## Grading Criteria

We will grade your project submission according to the following percentage breakdown.

Percent of project grade	Item graded
30%	The completeness and correctness of your design. The business rules, conceptual and logical ERDs, and database schema will be examined, including the entities, relationships, cardinalities, and primary and foreign keys. Your tables should be normalized to BCNF, or accompanied with a valid reason why the table was not normalized to BCNF.
5%	Submitting all five iterative term project deliverables. Each deliverable should represent a good faith attempt for an initial iteration of each section.
5%	Correct, working SQL scripts that are consistent with the other deliverables in the project. Using a database user with an empty schema, your facilitator should be able to execute your create script, your insert script, and your query script(s) without errors.
5%	Correct, working primary, foreign, and not null constraints.
5%	The index and explanation.
30%	SQL for the five aspects.
20%	Exposition – How clearly and persuasively the ideas and designs are presented, and how well organized your Term Project submission is.



**Attribution**

Amazon logo retrieved from <http://amazon.com> on Oct-07-2017.