The ***Store*** *object* is the center of our design and allows us to manage customers, transactions, and inventory on a store per store basis. The ***Store*** is responsible for loading in new items and customers and handling commands. The ***Store*** uses a hash function to map customer IDs to customer objects. The ***Store*** object is also responsible for creating and storing **Transactions**.

The ***Customer object*** allows us to keep track of customers information including **Transaction** history which holds references to the **Customer’s** associated **Transactions**.

The **Transaction** class is a template for all the different types of **Transaction** objects (borrow, rent, buy, etc.). A **Transaction** is associated with a unique **Customer** and its corresponding **Item**. The creation of a particular **Transaction** is managed by the **TxFactory**. We use this to facilitate adding new children of transactions in the future.

The **Item** class is a template for all the different types of **Items** that are in the store (movie, food, etc.). The creation of a particular **Item** is managed by the **itemFactory**. We use this to facilitate adding new children of items in the future. The inventory is tracked by storing items alongside quantity in the **Store** object.

The **checkedItems** struct allows for the store to keep track of due dates for checked items and their associated **Transactions**.

The **HashMap** class allows us to implement a hashmap data structure for storing customers.

The **main()** function will create **Store** object(s), manage iterating through the data file inputs, and loading that file data into the **Store’s** relevant functions. To manage multiple stores, we could create and load the relevant data into those objects. **Main()** will also iterate through the command file inputs and pass the commands into the **Store** object.