**Description:**

*Juicy Vendors* distribute vending machines to various organizations. They want to promote their business by giving access to their vending machines from a customer’s mobile phone. A part of their software requirement is given below. You are required to do additional self-study on how vending machines function and create a design for managing it.

**Requirements:** *Each item in a vending machine has item ID, item names, manufacturer name, price, and the quantity available in a machine. The items are of three types, liquids, sweet snacks, and salty snacks. The app on the phone displays all items that are available. A customer is a registered user of the mobile app whose name, address, and mobile number are stored in the system. When the phone application starts, the user puts the ID of the vending machine and the app displays the items and quantity of each available item in the machine. The customer should be able to select and pay for multiple items during the same session. When a customer pays for all selected item, the application should reduce the quantity of the selected item and also update and display the total amount of sales. Once the sale is complete, the customer is shown a random and unique alphanumeric key on the phone. When the customer punches in the key on the physical machine, the purchased items are dispensed to the customer.*

**Submission:**

* Submit a report that has the following:
  + A paraphrased description of the system based on the requirements given above and a self-study of the system.
  + A list of all classes (related attributes and behaviors), relationships between classes, and assumptions made.
  + UML class diagram with all class relationships included.
  + Python code that represent classes, which includes the constructor, setter/getter, and other functions for the given requirements.
  + 5% of the total score will be allocated for good documentation of the code and timely submission

**Report Format**

* Title Page: Include case-study title, student ID, and full name
* Problem Analysis (20%): In this section, based on given requirements and self-study, a detailed list of all the requirements is given. The list of all classes, their attributes, and behaviors are also listed with data types.
* Functional Design (20%): In this section, the algorithm or flow chart is provided to explain the logical flow that will drive the use of the system. File structure and information stored in files are described.
* Class Design (30%): The UML class diagram, with class relationships, and cardinality for the business case is provided. Each relationship is explained, and assumptions are listed.
* Pseudocode (20%): In this section, provide the Python class structures for all the identified classes with required functionalities. The testing of the system is NOT required.
* Conclusion (5%): In this section include a reflection on what was learned in this exercise, the challenges faced while working on this assignment, and how the system can be further expanded.

class item:

item ID

item names

manufacturer name

price

quantity available in a machine

items are of three types, (3 child classes)

liquids,

sweet snacks,

and salty snacks.