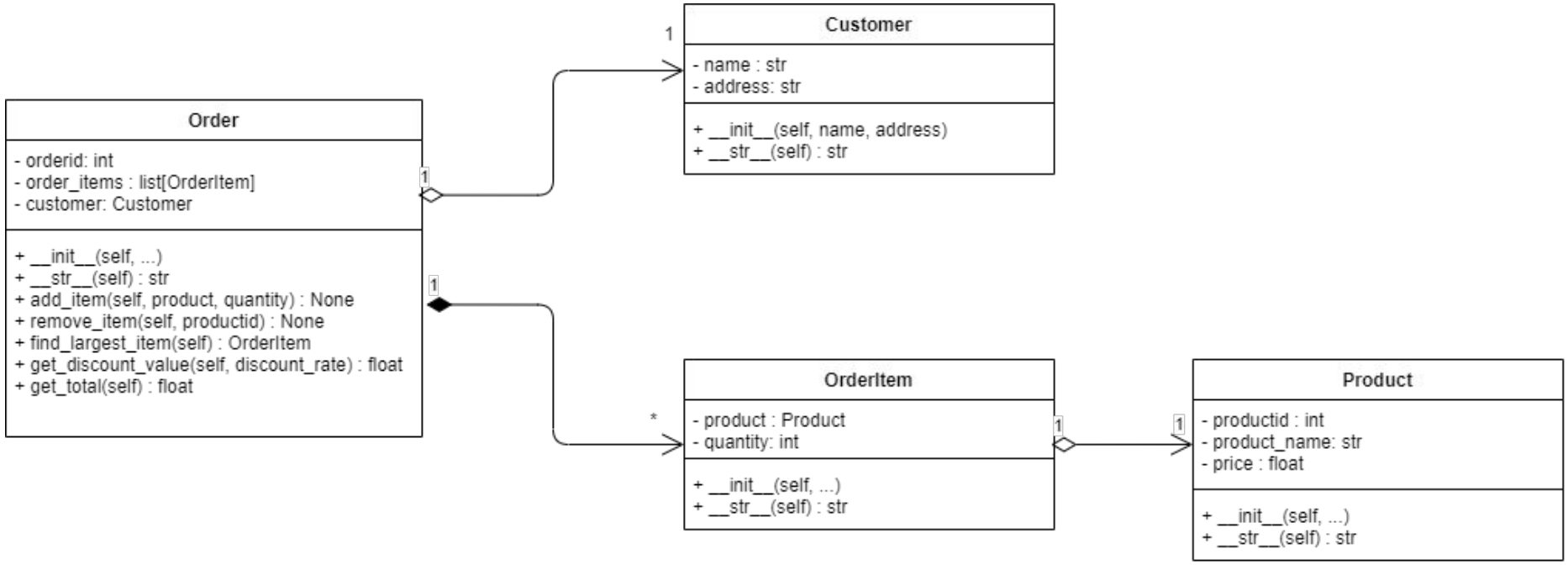
Simple Python Classes

Python Lab Exercises

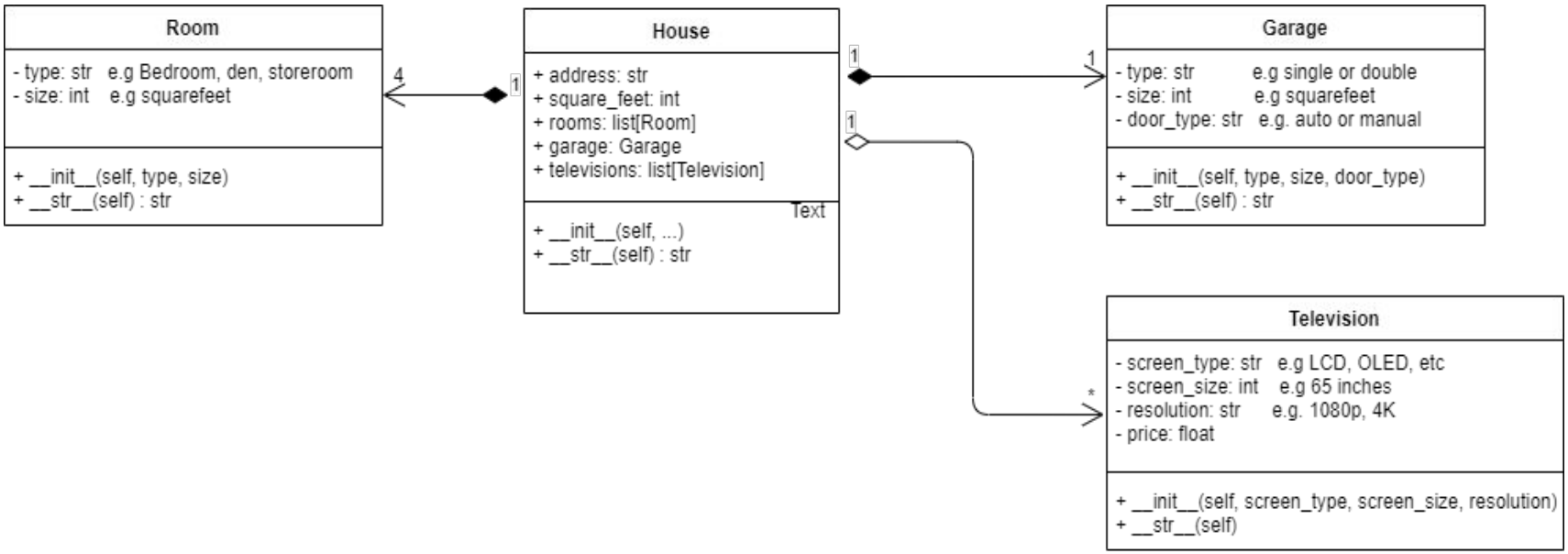
Question - Order Class

- Implement the following classes according to the class diagrams.
- You need to define necessary function parameters for all the classes' constructors (...) to complete constructor implementations.
- After you implement all the classes, write a main method to create objects of the classes you defined and print out their contents.
- Each class should have appropriate setters and getters using property annotation to access private attributes
- Enhance the Order class so that it allows the main method to add and remove the order items from the Order instance.



Question - House Class

- Implement the following classes according to the following class diagrams.
- After you implement all the classes, write a main method to create objects of the classes you defined and print out their contents.
- Each class should have appropriate setters and getters using property annotation for accessing private attributes
- The House class should allow other classes to
  - add and remove a TV object the televisions list.
  - change the garage object's size
- The House class should also have the following public methods:
  - def get_biggest_room(self) -> Room
    - Based on the size of the room, find the largest one.
  - def get_oled_televisions(self) -> list[Television]
    - Get a list of televisions with an OLED display
  - def is_siimilar_house(self, other) -> bool
    - If two houses have the same square footage and number of rooms, they are considered similar



Question - Month class

Implement a class named Month whose objects represent months.

- Data member:
  - month_number: represents a unique month of the year. i.e. 0 <= month_number < 12
- Constructor takes a string argument (e.g. January, February, …. , or December) so it needs to converts it to month_number.
- Other public methods:
  - advance(self); advances to the next month
  - prev(self); go back to the previous month.
  - display(self) that prints the month_number in string i.e. January, February, …. , or December
  - compare(m: Month) that takes a Month object,
    - if this object is greater than the m object, returns 1.
    - if this object is smaller than the m object, returns -1
    - if this object is equal to the m object, returns 0.
- Write a main() function to test your Month class.