Department of Computer Science

420-201-VA Programming 2

Week 11 part 1: Classes and Objects

Task 1:

Implement the following classes based on descriptions below:

A. Design a `Ship Class` that has the following members:
   a. A field for the name of the ship (a string).
   b. A field for the year that the ship was built (a string).
   c. A constructor and appropriate accessors and mutators.
   d. A `toString` method that displays the ship's name and the year it was built.

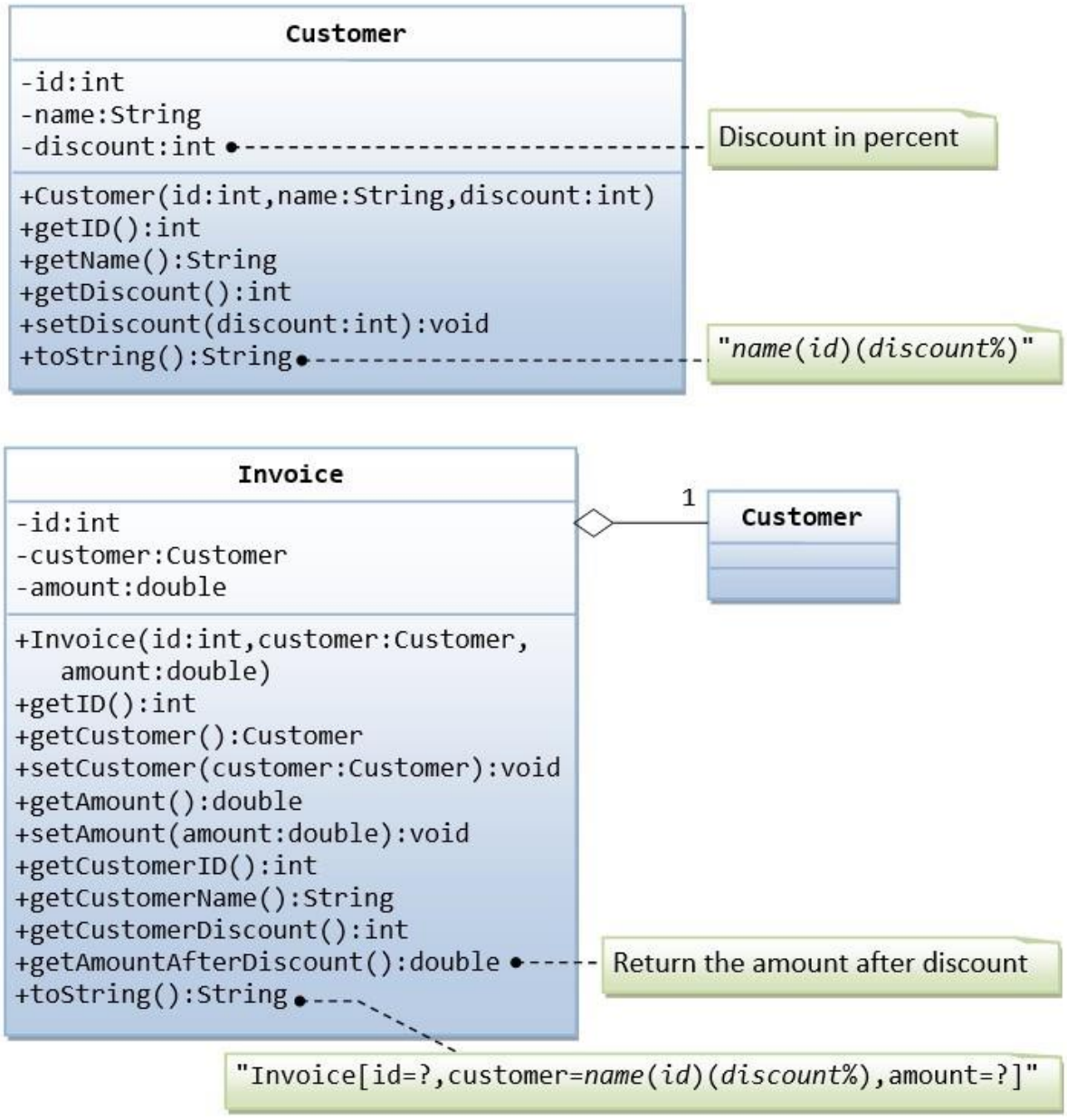
B. Design a `CruiseShip` class that extends the `Ship` class. The `CruiseShip` class should have the following members:
   1. A field for the maximum number of passengers (an int).
   2. A constructor and appropriate accessors and mutators.
   3. A `toString` method that overrides the toString method in the base class. The CruiseShip class's toString method should display only the ship's name and the maximum number of passengers.

C. Design a `CargoShip` class that extends the `Ship` class. The `CargoShip` class should have the following members:
   1. A field for the cargo capacity in tonnage (an int).
   2. A constructor and appropriate accessors and mutators.
   3. A `toString` method that overrides the toString method in the base class. The `CargoShip` class's toString method should display only the ship's name and the ship's cargo capacity.

D. Demonstrate the classes in a program that has a Ship array. Assign various Ship, CruiseShip, and CargoShip objects to the array elements. **The program should then step through the array,** calling each object's toString method.

Task 02

Implement the following classes along with their driver class based on the UML class diagrams: (implement a ***has-a*** relationship)


Customer
-id:int
-name:String
-discount:int
Discount in percent
+Customer(id:int,name:String,discount:int)
+getID():int
+getName():String
+getDiscount():int
+setDiscount(discount:int):void
+toString():String
"name(id)(discount%)"
Invoice
-id:int
-customer:Customer
-amount:double
+Invoice(id:int,customer:Customer,
amount:double)
+getID():int
+getCustomer():Customer
+setCustomer(customer:Customer):void
+getAmount():double
+setAmount(amount:double):void
+getCustomerID():int
+getCustomerName():String
+getCustomerDiscount():int
+getAmountAfterDiscount():double
Return the amount after discount
+toString():String
"Invoice[id=?,customer=name(id)(discount%),amount=?]"
1
Customer