ESOF 322, Homework 1

September 8, 2019

YuehChen Tsou and John Singleton

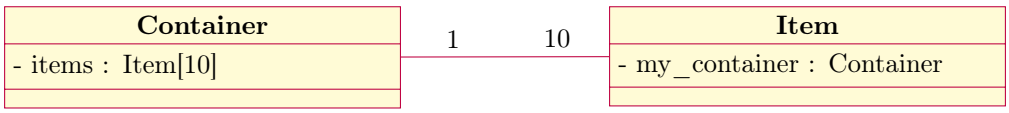
Exercises Part A (15pts) .

For each of the following (pseudo) code snippets provide the UML class diagram.

1 Question 1, 2pts

```
public class Container {
    private Item[10] items;
}
public class Item {
    private Container my_container;
}
```

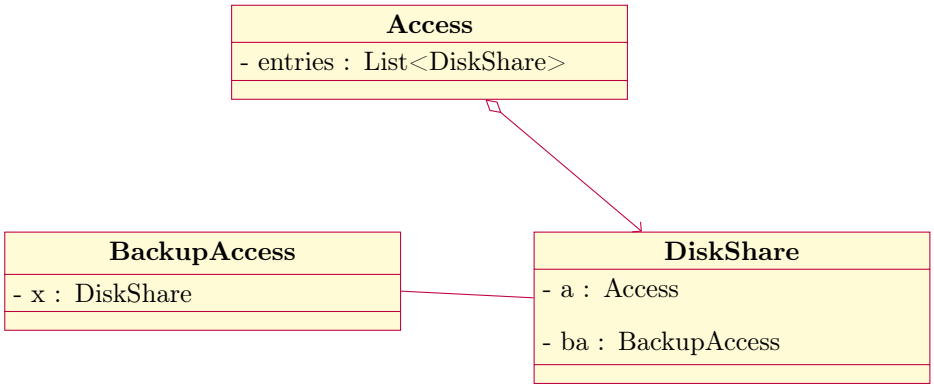
Solution :



2 Question 2, 3pts

```
public class Access {
    private List<DiskShare> entries;
}
public class BackupAccess {
    private DiskShare x;
}
public class DiskShare {
    private Access a;
    private BackupAccess ba;
}
```

Solution :

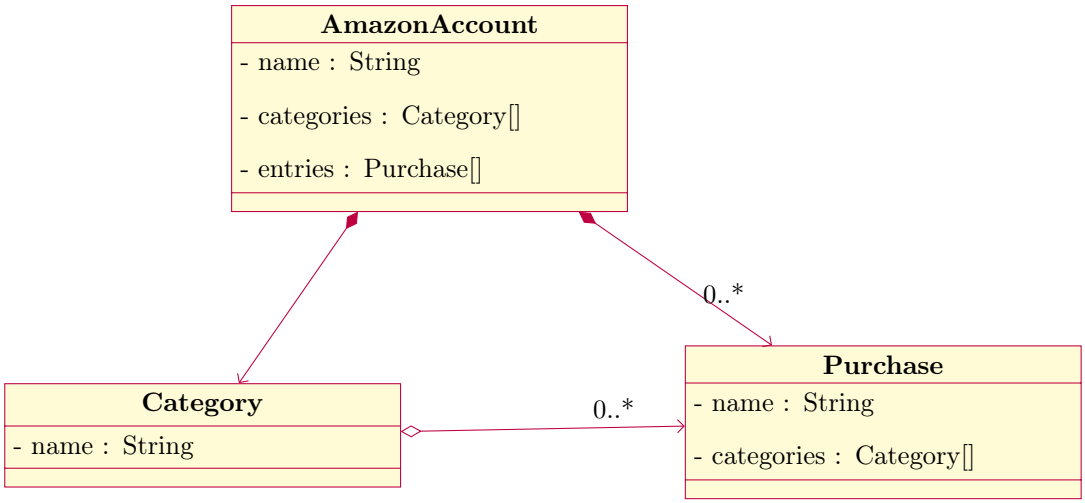


*We are picturing Access as something like SETI@home, where people are volunteering to share their disks.

3 Question 3, 5pts

```
public class AmazonAccount {
    private String name;
    private Category[] categories;
    private Purchase[] entries;
}
public class Category {
    private String name;
}
public class Purchase {
    private String name;
    private Category[] categories;
}
```

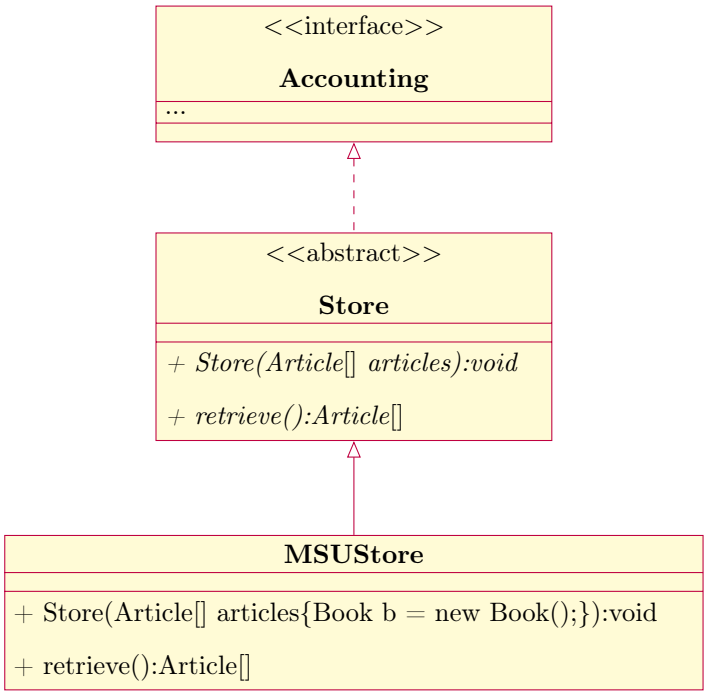
Solution :



4 Question 4, 5pts

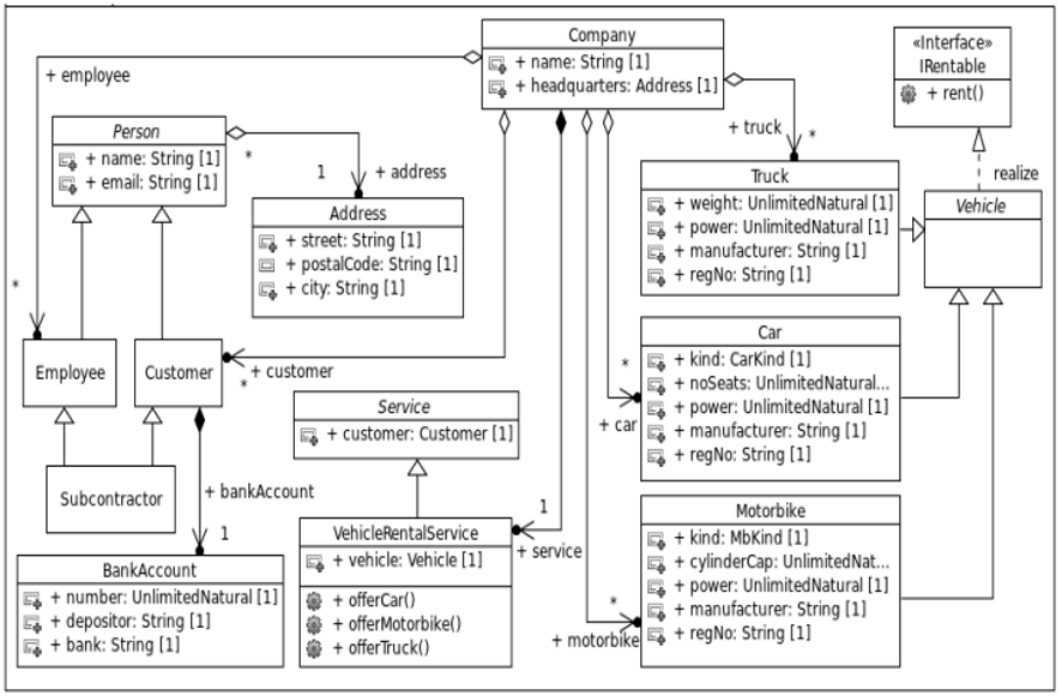
```
public abstract class Store {
    public abstract void Store(Article[] articles);
    public abstract Article[] retrieve();
}
public interface Accounting {...}
public class MSUStore extends Store implements Accounting {
    public void Store(Article[] articles) {Book b=new Book(); // other code...}
    public Article[] retrieve(){...}
}
```

Solution :



Exerciss Part B (15 pts)

Write pseudo code to describe the following UML class diagram:



Solution :

```
public class Company {
    public static void main(String[] args) {
        public String[1] name;
        public Address[1] headquarters;
        public Truck truck;
        public Car car;
        public MotorBike mb;
        public VehicleRentalService vrService;
        public Customer customer;
        public Employee employee;
    }
}
/*
* Classes of Truck, Car, and MotorBike extends abstract class Vehicle, and implements interface Rentable
*/
public interface Rentable {
    public rent()
}
public abstract class Vehicle{
```

```
}
public class Truck extends Vehicle implements Rentable {
    public UnlimitedNatural[1] weight;
    public UnlimitedNatural[1] power;
    public String[1] manufacturer;
    public String[1] regNo;
}
public class Car extends Vehicle implements Rentable {
    public CarKind[1] kind;
    public UnlimitedNatural[1] noSeats;
    public UnlimitedNatural[1] power;
    public String[1] manufacturer;
    public String[1] regNo;
}
public class Motorbike extends Vehicle implements Rentable {
    public MbKind[1] kind;
    public UnlimitedNatural[1] cylinderCap;
    public UnlimitedNatural[1] power;
    public String[1] manufacturer;
    public String[1] regNo;
}
public abstract class Service {
    public Customer[1] customer;
}
public class VehicleRentalService extends Service {
    public Vehicle[1] vehicle;
    public offerCar();
    public offerMotorbike();
    public offerTruck();
}
public abstract class Person {
    public String[1] name;
    public String[1] email;
    public Address address;
}
public class Address{
    public String[1] street;
    public String[1] postalCode;
    public String[1] city;
```

```
}
public class Customer extends Person {
    public BankAccount[1] bankAccount;
}
public class Employee extends Person {}
public class Subcontractor extends Customer extends Employee {}
public class BankAccount {
    public UnlimitedNatural[1] number;
    public String[1] depositor;
    public String[1] bank;
}
```