Victor Mao

CS 2336.003 Project 2 Pseudocode

Disneyland Dining Rewards

Customer Class

* Variables
  + private String firstName
  + private String lastName
  + private int id
  + private double amountSpent
* Methods
  + Overloaded Constructor
    - public Customer(String fn, String ln, int id, double as)
  + Accessors/Getters/return variable value
    - public String getFirstName() { return firstName; }
    - public String getLastName() { return lastName; }
    - public int getId() { return id; }
    - public double getAmountSpent() { return amountSpent; }
  + Mutators/Setters/change variable value
    - public void setFirstName(String fn) { firstName = fn; }
    - public void setLastName(String ln) { lastName = ln; }
    - public void setId(int id) { this.id = id; }
    - public void setAmountSpent(double as) { amountSpent = as; }

Preferred Customer Class

* Variable
  + private int discountPercentage
* Methods
  + Overloaded Constructor
    - public preferredCustomer(String fn, String ln, int id, double as, int dp)
      * discountPercentage = dp;
      * super(fn, ln, id, as);
  + Accessor/Getter/returns variable value
    - public int getDiscountPercentage() { return discountPercentage; }
  + Mutator/Setter/change variable value
    - public void setDiscountPercentage(int dp) { discountPercentage = dp; }

Main Class

* Imports
  + java.io.\*;
  + java.util.Scanner;
* Logic
  + Customer[] customers = readCustomerFile(“customer.dat”);
  + PreferredCustomer[] pCustomers = new PreferredCustomer[];
  + boolean exists = preferredCustomerExists(“preferred.dat”);
  + If (exists)
    - pCustomers = readPreferredCustomerFile(“preferred.dat”);
  + Scanner scan = new Scanner(new File(“orders.dat”);
  + While order has next line
    - Read line of order
    - Fill int customerId, quantity with values, discount
    - Fill double radius, height, ounces, ouncesPrice, squareInchPrice with values
    - Loop through customers to see if in customers
    - If in customers
      * double totalPrice = getTotalPrice(radius, height, ounces, ouncesprice, squareInchPrice) \* quantity;
    - Else loop through preferredCustomers to see if is a preferredCustomers
      * discount = pCustomer.getDiscountPercentage();
      * double totalPrice = getTotalPrice(radius, height, ounces, ouncesprice, squareInchPrice) \* quantity;
    - Update amountSpent
    - Check if amountSpent has surpassed threshold
      * If pass $150
        + Create new PreferredCustomer pc
        + addPreferredCustomer(pCustomers, c)
        + removeCustomer(customers, c)
      * If pass $200
        + setDiscountPercentage to 7
      * If pass $350
        + setDiscountPercentage to 10
  + Print information to files
    - writeCustomerFile(customers, “customer.dat”);
    - writePreferredCustomerFIle(preferredCustomers, “preferred.dat”);
* Variables
  + Customer[] customers = new Customer[]
  + PreferredCustomer[] pCustomers = new PreferredCustomer[]
* Methods
  + public Customer[] readCustomerFile(String customerFilePath)
    - Open file at “customer.dat”
    - While not EOF
      * Read line and create new Customer
      * Add new Customer to Customer array
    - Close file
    - Return Customer array
  + public boolean preferredCustomerExists(String preferredCustomerFilePath)
    - Return true if “preferred.dat” exists
    - Else return false
  + public PreferredCustomer[] readPreferredCustomerFile(String preferredCustomerFilePath)
    - Open file at “preferred.dat”
    - While not EOF
      * Read line and create new PreferredCustomer
      * Add new PreferredCustomer to PreferredCustomer array
    - Close file
    - Return PreferredCustomer array
  + public PreferredCustomer[] addPreferredCustomer(PreferredCustomer[] pc, Customer c)
    - Create a new PreferredCustomer[] array that is one larger than the size of pc
    - Transfer PreferredCustomers from pc into new PreferredCustomer[] array
    - Upgrade Customer c into a PreferredCustomer
    - Add the new PreferredCustomer into the end of new PreferredCustomer[] array
    - Return the new PreferredCustomer[] array
  + public Customer[] removeCustomer(Customer[] ca, Customer c)
    - For loop through to find the index of c in ca
    - Create a new Customer[] array that is one smaller than the size of ca
    - Transfer Customers from c[0] to c[index] into new Customer[] array
    - Skip Customer c
    - Transfer Customers from c[index+1] to c[c.length] into new Customer[] array
    - Return the new Customer[] array
  + public double getVolume(double r, double h)
    - return Math.PI \* r \* r \* h;
  + public double getSurfaceArea(double r, double h)
    - return (2 \* Math.PI \* r \* h) + (2 \* Math.PI \* r \* r);
  + public double getLiquidPrice(double oz, double ozPrice)
    - return (ounces \* ozPrice);
  + public double getPersonalizationPrice(double r, double h, double squareInchPrice)
    - return getSurfaceArea(r, h) \* squareInchPrice;
  + public double getTotalPrice(double r, double h, double oz, double ozPrice, double squreInchPrice)
    - return getLiquidPrice(oz, ozPrice) + getPersonalizationPrice(r, h, squareincePrice);
  + public double getTotalPrice(double r, double h, double oz, double ozPrice, double squreInchPrice, int discount)
    - double prePrice = getLiquidPrice(oz, ozPrice) + getPersonalizationPrice(r, h, squareincePrice);
    - Apply discount
    - Return totalPrice
  + public void writeCustomerFile(Customer[] c, String customerFilePath)
    - BufferedWriter bw = new BufferedWriter(“customer.dat”);
    - For all customers in c
      * Write information to file
    - Close bw
  + public void writePreferredCustomerFile(PreferredCustomer[] pc, String preferredCustomerFilePath)
    - BufferedWriter bw = new BufferedWriter(“preferred.dat”);
    - For all preferred customers in pc
      * Write information to file
    - Close bw