Create A Class

The JackCard from Jackson Café is a convenient way for customers to pay at the restaurant. Associate with each JackCard is the following information

- Card Holder – the name of the card holder
- Balance – the amount balance on the card
- Discount rate – the discount percent for all purchases. It changes with the balance:
  - $100 and over – 2%
  - $200 and over – 3%
  - $500 and over – 5%
- Points – the number of points collected on this card. One point is rewarded for every $2 purchase (before discount). The points can be redeemed with every 20 points equivalent to $1 (before discount).
- PIN – the security feature (numbers only) that is required every time money is spent or points are redeemed

Besides, there is also a maximum balance set by Jackson Café on the maximum balance that all JackCards can carry.

1. Create a class `JackCard` with the required instance fields.

2. Write the accessor and mutator methods for the instance fields.

3. Write a constructor method that has no parameters. The method should construct a `JackCard` object with name "`default`", balance $`0`, discount percent `0`%, and `0` point and PIN `1234`

4. Write a constructor method that has three parameters representing the name, initial balance, and PIN. The discount percent should be determined accordingly.

5. Write an instance method `refill` to add the specified amount to the card. In case the amount will bring the balance over the maximum, the balance will only be top up to the maximum. Be sure to update the discount rate with the new balance. The method should return a `boolean`, indicating if the full amount can be added.

6. Write a method `pay` to pay for a purchase. It should take three parameters, `amount`, `PIN`, `usePoint` (`boolean`). It should first validate the PIN before spending any money / points. The `usePoint` (`boolean`) indicate if the customer wants to redeem the points if they are sufficient for the purchase. Be sure to update the discount rate with the new balance. The method should return a `boolean`, indicating if the purchase is successful (correct PIN, enough money / points).

7. Write a method `higherBalance` that could be called by a statement like

   ```
   c3 = c1.higherBalance(c2);
   ```

   where `c1`, `c2`, and `c3` are objects of type `JackCard`. The method should make `c3` refer to whichever c1 or c2 that has the higher balance (or `c1` if `c1` and `c2` have the same balance)

8. Write a method `morePoints` that could be called by a statement like
   ```
   c3 = c1.morePoints(c2);
   ```

   where `c1`, `c2`, and `c3` are objects of type `JackCard`. The method should make `c3` refer to whichever c1 or c2 that has the more points (or `c1` if `c1` and `c2` have the same number of points)

9. Write a method `combinePoints` that could be called by a statement like
   ```
   c1.combinePoints(c2);
   ```

   where `c1`, and `c2` are objects of type `JackCard`. The method would combine the points of c1 and c2 and put them in c1.

10. Write a method `changePIN` that takes in a name and a new PIN. The PIN can only be changed if the name matches the card holder name exactly. It should return a `boolean`, indicating the PIN is changed successfully.

11. Write a `toString` method for the JackCard class. The `toString` method should return a `String` with the format:
    ```
    Card Holder: Bobby
    Balance: $3.45
    Discount rate: 0%
    Points: 45
    PIN: 4554
    ```

12. Create a class `TestJackCard` which contains the main method. The main method should perform the following actions. Output error message where applicable.
    a. Create two JackCard objects c1, representing a card with the default property, and c2 representing a card with holder “Bobby”, an initial balance of $100 and PIN 3443.
    b. Prompt user for the maximum balance that is allowed on a card.
    c. Refill c1 with $50
    d. Make a purchase of $60, without redeeming the points, using c2.
    e. Make a purchase of $10, without redeeming the points, using the card with higher balance.
    f. Use points to redeem a purchase of $5, using the card with higher points.
    g. Combine the points of two cards into c1.
    h. User points to redeem a purchase of $5 using c1
    i. Change the PIN for c1. Prompt user for the card holder name and the new PIN.