**Assumption: Code Fragment 1.6 Structure**

Code Fragment 1.6 likely defines a CreditCard class with three credit card objects, then iterates through a loop to make charges.

Assuming the original structure looks something like this:

public class CreditCard {

private String customer;

private String bank;

private String account;

private int limit;

private double balance;

// Constructor

public CreditCard(String cust, String bk, String acnt, int lim, double initialBal) {

customer = cust;

bank = bk;

account = acnt;

limit = lim;

balance = initialBal;

}

// Charge method

public boolean charge(double price) {

if (balance + price > limit) {

System.out.println("Transaction denied! Credit limit exceeded for " + customer);

return false;

}

balance += price;

return true;

}

// Method to make a payment

public void makePayment(double amount) {

balance -= amount;

}

// Display method

public void printSummary() {

System.out.println("Customer: " + customer);

System.out.println("Bank: " + bank);

System.out.println("Account: " + account);

System.out.println("Balance: $" + balance);

System.out.println("Credit Limit: $" + limit);

System.out.println();

}

public static void main(String[] args) {

// Create three credit card objects

CreditCard[] wallet = new CreditCard[3];

wallet[0] = new CreditCard("John Doe", "Bank A", "1234", 5000, 0);

wallet[1] = new CreditCard("Jane Doe", "Bank B", "5678", 3000, 0);

wallet[2] = new CreditCard("Jim Beam", "Bank C", "9876", 2000, 0);

// Modify the first for loop so that one credit card exceeds its limit

for (int i = 1; i <= 3; i++) {

wallet[0].charge(1000 \* i); // Charges: 1000, 2000, 3000 (Total = 6000, within limit)

wallet[1].charge(1500 \* i); // Charges: 1500, 3000, 4500 (Total = 9000, exceeds limit)

wallet[2].charge(500 \* i); // Charges: 500, 1000, 1500 (Total = 3000, within limit)

}

// Print the summary of all cards

for (CreditCard card : wallet) {

card.printSummary();

}

}

}

### ****Changes & Explanation:****

* Modified the loop so that:
  + wallet[0] charges 1000, 2000, 3000 → Total = **6000 (within limit)**
  + wallet[1] charges 1500, 3000, 4500 → Total = **9000 (exceeds limit of 3000)**
  + wallet[2] charges 500, 1000, 1500 → Total = **3000 (within limit)**

### ****Which Card Exceeds the Limit?****

* **Jane Doe’s card (wallet[1])** has a **credit limit of $3000**, but the loop attempts to charge **$9000**, causing it to exceed its limit.