Task: 03 **Date: 22/05/2024**

Use Case1: ATM Machine

Write a Code to implement an ATM replica. Assign initial balance Rs.10000, Cash available in Machine Rs.50000, PIN 1234. Write a code to execute the same for an infinite number of times as many times as the user wants.

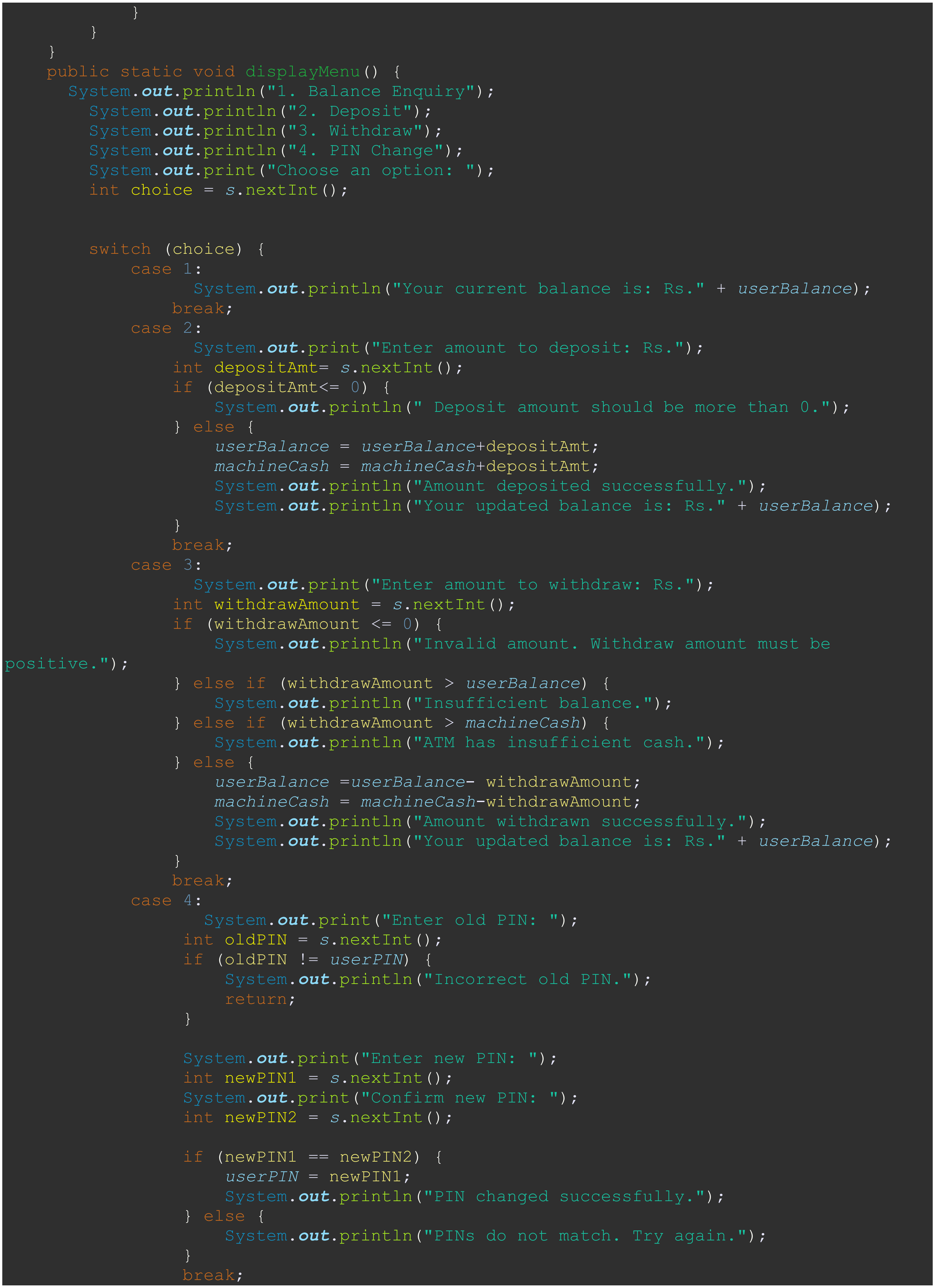
For every iteration ask the user to enter PIN and verify the PIN. Then display the following options:

1. Balance Enquiry
2. Deposit
3. Withdraw
4. PIN change.

- Check whether the amount is valid or not to perform a deposit operation and display the updated balance after the deposit operation.
- Perform withdrawal operations by checking multiple conditions and displaying updated balances.
- Ask the user to enter the old PIN number before changing to a new PIN. If the old PIN is valid then ask new PIN twice to confirm the new PIN.
- After every iteration ask the user to continue. If the user says yes then continue with another iteration else exit from the program.

Code

```
package work;

import java.util.Scanner;

public class ATM_machine {
      static int userBalance = 10000;
      static int machineCash = 50000;
    public static int userPIN = 1234;
    static Scanner s = new Scanner(System.in);

    public static void main(String[] args) {
        if (!authenticateUser()) {
            System.out.println("incorrect attempts");
        } else {
            boolean continueMenu = true;
            while (continueMenu) {
                displayMenu();
                continueMenu = askToContinue();
            }
            System.out.println("Thank you for using the ATM.");
        }
    }
    public static boolean authenticateUser() {
        while (true) {
            System.out.println("Enter PIN:");
            int enteredPin = s.nextInt();
            if (enteredPin == userPIN) {
                return true;
            } else {
                System.out.println("Incorrect PIN.");
                System.out.println("enter correct PIN.");
```



```
            }
        }
    }
    public static void displayMenu() {
      System.out.println("1. Balance Enquiry");
        System.out.println("2. Deposit");
        System.out.println("3. Withdraw");
        System.out.println("4. PIN Change");
        System.out.print("Choose an option: ");
        int choice = s.nextInt();


        switch (choice) {
            case 1:
                  System.out.println("Your current balance is: Rs." + userBalance);
                break;
            case 2:
                  System.out.print("Enter amount to deposit: Rs.");
                int depositAmt= s.nextInt();
                if (depositAmt<= 0) {
                    System.out.println(" Deposit amount should be more than 0.");
                } else {
                    userBalance = userBalance+depositAmt;
                    machineCash = machineCash+depositAmt;
                    System.out.println("Amount deposited successfully.");
                    System.out.println("Your updated balance is: Rs." + userBalance);
                }
                break;
            case 3:
                  System.out.print("Enter amount to withdraw: Rs.");
                int withdrawAmount = s.nextInt();
                if (withdrawAmount <= 0) {
                    System.out.println("Invalid amount. Withdraw amount must be 
positive.");
                } else if (withdrawAmount > userBalance) {
                    System.out.println("Insufficient balance.");
                } else if (withdrawAmount > machineCash) {
                    System.out.println("ATM has insufficient cash.");
                } else {
                    userBalance =userBalance- withdrawAmount;
                    machineCash = machineCash-withdrawAmount;
                    System.out.println("Amount withdrawn successfully.");
                    System.out.println("Your updated balance is: Rs." + userBalance);
                }
                break;
            case 4:
                   System.out.print("Enter old PIN: ");
                 int oldPIN = s.nextInt();
                 if (oldPIN != userPIN) {
                     System.out.println("Incorrect old PIN.");
                     return;
                 }

                 System.out.print("Enter new PIN: ");
                 int newPIN1 = s.nextInt();
                 System.out.print("Confirm new PIN: ");
                 int newPIN2 = s.nextInt();

                 if (newPIN1 == newPIN2) {
                     userPIN = newPIN1;
                     System.out.println("PIN changed successfully.");
                 } else {
                     System.out.println("PINs do not match. Try again.");
                 }
                 break;
            default:
                System.out.println("Invalid option.");
        }


    }
    public static boolean askToContinue() {
        System.out.print("Do you want to continue? (yes/no): ");
        s.nextLine();
        String continueOption = s.nextLine();
        return continueOption.equalsIgnoreCase("yes");
    }

}
```

Out put

```
Enter PIN:
1234
1. Balance Enquiry
2. Deposit
3. Withdraw
4. PIN Change
Choose an option: 1
Your current balance is: Rs.10000
Do you want to continue? (yes/no): yes
1. Balance Enquiry
2. Deposit
3. Withdraw
4. PIN Change
Choose an option: 3
Enter amount to withdraw: Rs.12
Amount withdrawn successfully.
Your updated balance is: Rs.9988
Do you want to continue? (yes/no): yes
1. Balance Enquiry
2. Deposit
3. Withdraw
4. PIN Change
Choose an option: 45
Invalid option.
Do you want to continue? (yes/no):
```

Use Case2: Bus Ticket vending Machine

Write a code to implement a Bus ticket vending machine replica. Assign an initial number of tickets sold 0, the amount collected 0. Write a code to execute the same for an infinite number of times as many times as the user wants.

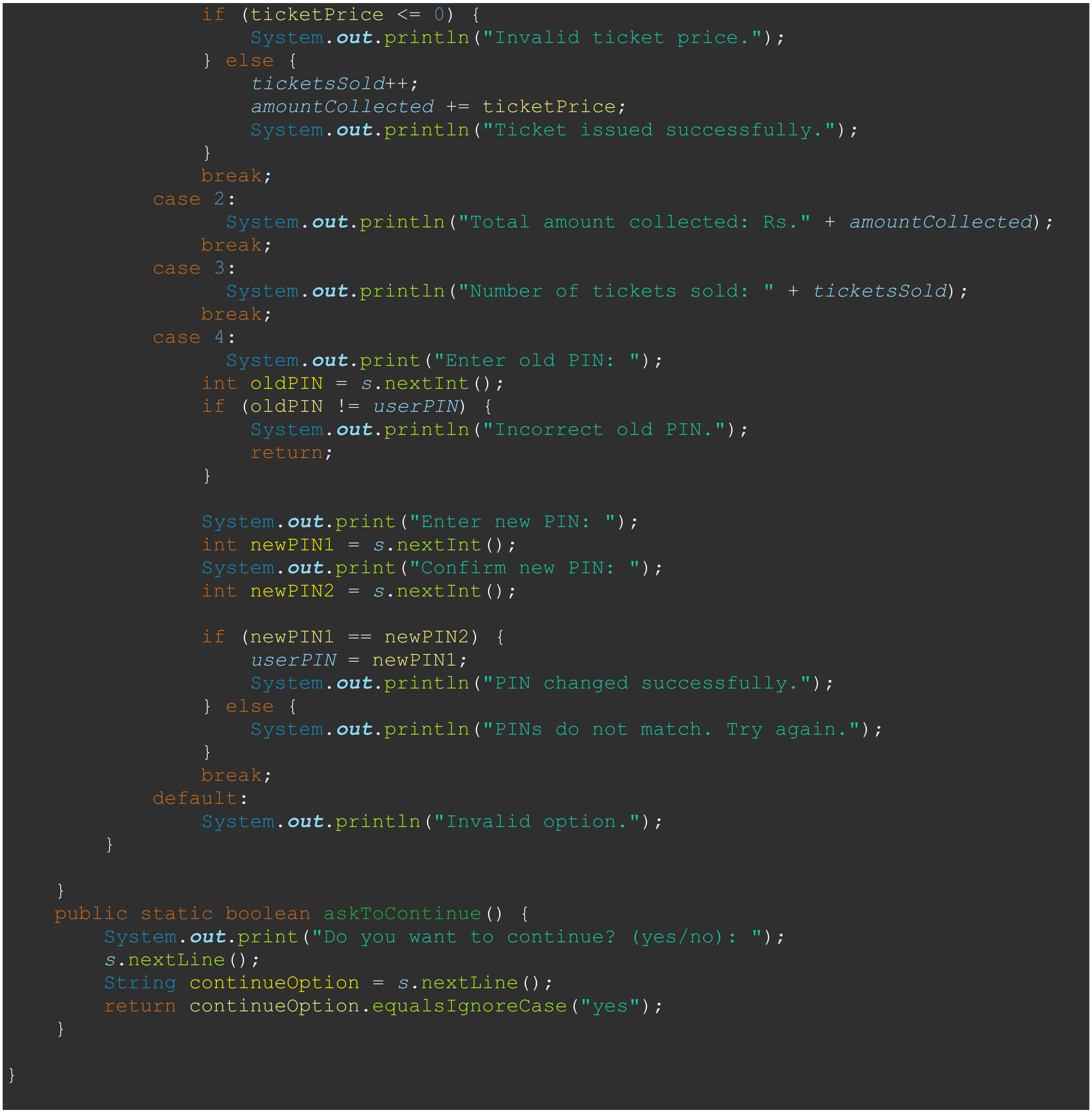
For every iteration ask the user to enter PIN and verify the PIN. Then display the following options:

1. Ticket issue.
2. Balance collected.
3. Number of tickets sold.
4. PIN change.
    - Apply conditions to issue a ticket (include discounts).

- Ask the user to enter the old PIN number before changing to a new PIN. If the old pin is valid then ask new PIN twice to confirm the new PIN.
- After every iteration ask the user to continue. If the user says yes then continue with another iteration else exit from the program.

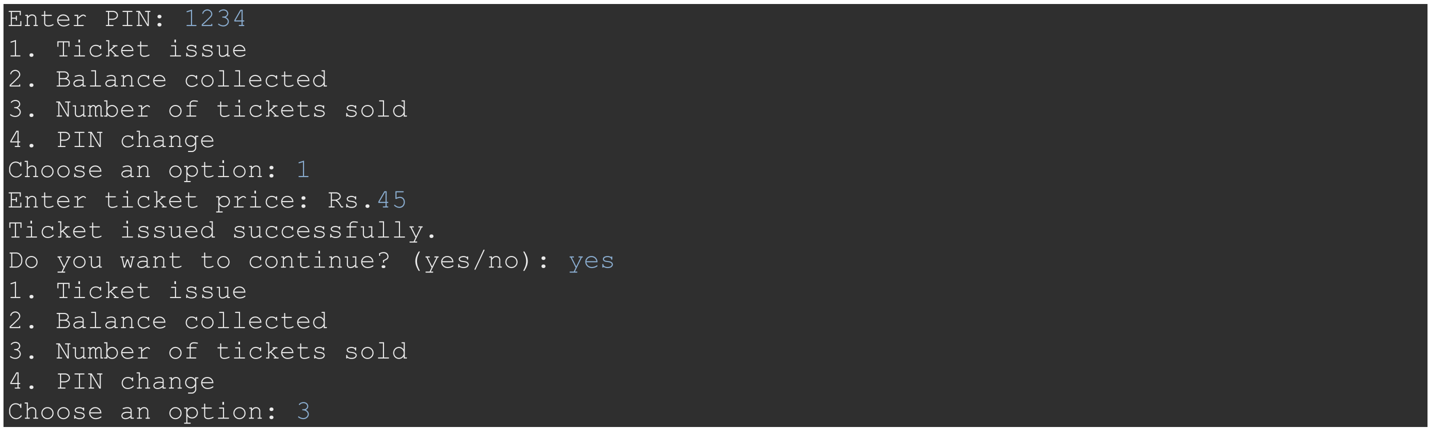
Code

```
package work;

import java.util.Scanner;

public class ticket {
    public static int ticketsSold = 0;
    public static double amountCollected = 0;
    public static int userPIN = 1234;
    public static Scanner s = new Scanner(System.in);

    public static void main(String[] args) {
      if (!authenticateUser()) {
            System.out.println("incorrect attempts");
        } else {
            boolean continueMenu = true;
            while (continueMenu) {
                displayMenu();
                continueMenu = askToContinue();
            }
            System.out.println("Thank you for using .");
        }
    }

    public static boolean authenticateUser() {
        final int MAX_ATTEMPTS = 3;
        int attempts = 0;

        while (attempts < MAX_ATTEMPTS) {
            System.out.print("Enter PIN: ");
            int enteredPIN = s.nextInt();

            if (verifyPIN(enteredPIN)) {
                return true;
            } else {
                System.out.println("Incorrect PIN. Try again.");
                attempts++;
            }
        }
        return false;
    }

    public static boolean verifyPIN(int enteredPIN) {
        return enteredPIN == userPIN;
    }

    public static void displayMenu() {
        System.out.println("1. Ticket issue");
        System.out.println("2. Balance collected");
        System.out.println("3. Number of tickets sold");
        System.out.println("4. PIN change");
        System.out.print("Choose an option: ");
        int choice = s.nextInt();

        switch (choice) {
            case 1:
                  System.out.print("Enter ticket price: Rs.");
                double ticketPrice = s.nextDouble();
```



```
                if (ticketPrice <= 0) {
                    System.out.println("Invalid ticket price.");
                } else {
                    ticketsSold++;
                    amountCollected += ticketPrice;
                    System.out.println("Ticket issued successfully.");
                }
                break;
            case 2:
                  System.out.println("Total amount collected: Rs." + amountCollected);
                break;
            case 3:
                  System.out.println("Number of tickets sold: " + ticketsSold);
                break;
            case 4:
                  System.out.print("Enter old PIN: ");
                int oldPIN = s.nextInt();
                if (oldPIN != userPIN) {
                    System.out.println("Incorrect old PIN.");
                    return;
                }

                System.out.print("Enter new PIN: ");
                int newPIN1 = s.nextInt();
                System.out.print("Confirm new PIN: ");
                int newPIN2 = s.nextInt();

                if (newPIN1 == newPIN2) {
                    userPIN = newPIN1;
                    System.out.println("PIN changed successfully.");
                } else {
                    System.out.println("PINs do not match. Try again.");
                }
                break;
            default:
                System.out.println("Invalid option.");
        }

    }
    public static boolean askToContinue() {
        System.out.print("Do you want to continue? (yes/no): ");
        s.nextLine();
        String continueOption = s.nextLine();
        return continueOption.equalsIgnoreCase("yes");
    }

}
```

Output



```
Enter PIN: 1234
1. Ticket issue
2. Balance collected
3. Number of tickets sold
4. PIN change
Choose an option: 1
Enter ticket price: Rs.45
Ticket issued successfully.
Do you want to continue? (yes/no): yes
1. Ticket issue
2. Balance collected
3. Number of tickets sold
4. PIN change
Choose an option: 3
```

```
Number of tickets sold: 1
Do you want to continue? (yes/no):
```