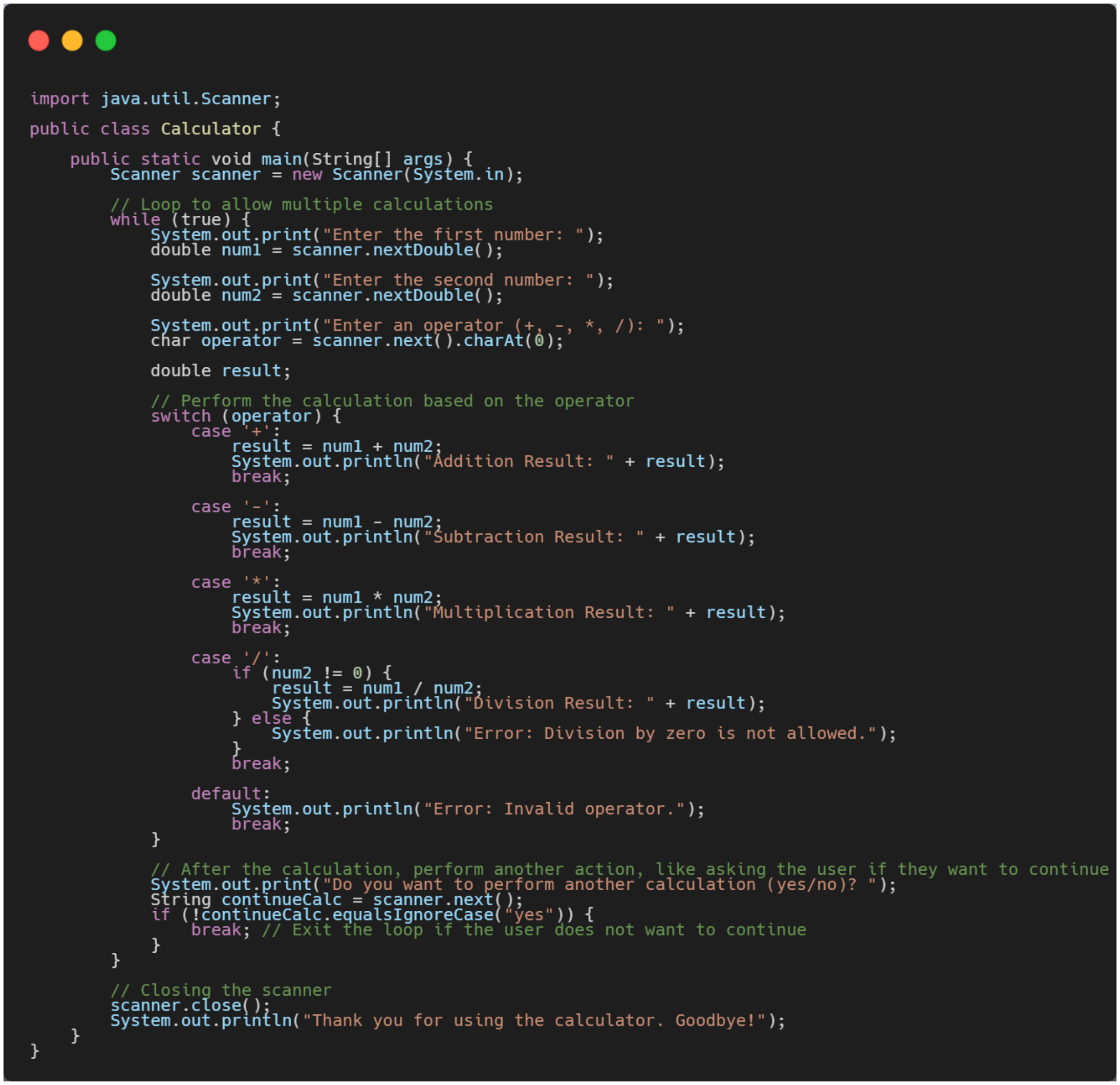
ASSIGNMENT - 3

1. Switch Statement

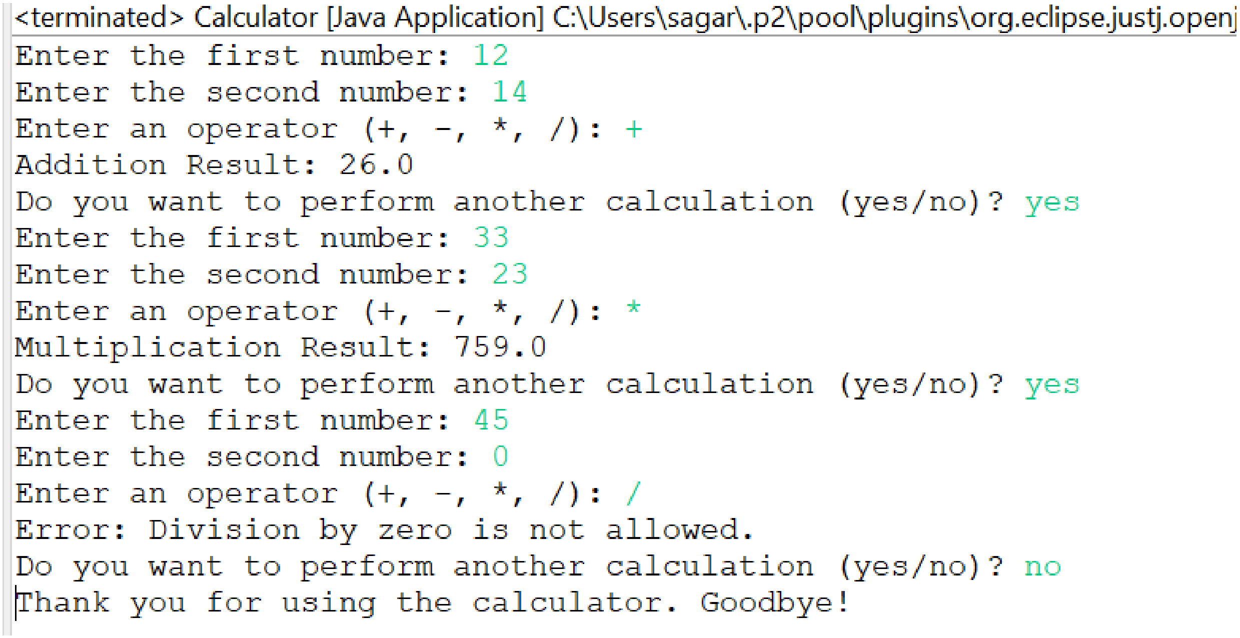
***Q)*Write a program that uses a switch statement to implement a basic calculator. It should take two numbers and an operator as input and perform the corresponding operation*.**

Ans- *Code*



```
import java.util.Scanner;

public class Calculator {

    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Loop to allow multiple calculations
        while (true) {
            System.out.print("Enter the first number: ");
            double num1 = scanner.nextDouble();

            System.out.print("Enter the second number: ");
            double num2 = scanner.nextDouble();

            System.out.print("Enter an operator (+, -, *, /): ");
            char operator = scanner.next().charAt(0);

            double result;

            // Perform the calculation based on the operator
            switch (operator) {
                case '+':
                    result = num1 + num2;
                    System.out.println("Addition Result: " + result);
                    break;

                case '-':
                    result = num1 - num2;
                    System.out.println("Subtraction Result: " + result);
                    break;

                case '*':
                    result = num1 * num2;
                    System.out.println("Multiplication Result: " + result);
                    break;

                case '/':
                    if (num2 != 0) {
                        result = num1 / num2;
                        System.out.println("Division Result: " + result);
                    } else {
                        System.out.println("Error: Division by zero is not allowed.");
                    }
                    break;

                default:
                    System.out.println("Error: Invalid operator.");
                    break;
            }

            // After the calculation, perform another action, like asking the user if they want to continue
            System.out.print("Do you want to perform another calculation (yes/no)? ");
            String continueCalc = scanner.next();
            if (!continueCalc.equalsIgnoreCase("yes")) {
                break; // Exit the loop if the user does not want to continue
            }
        }

        // Closing the scanner
        scanner.close();
        System.out.println("Thank you for using the calculator. Goodbye!");
    }
}
```

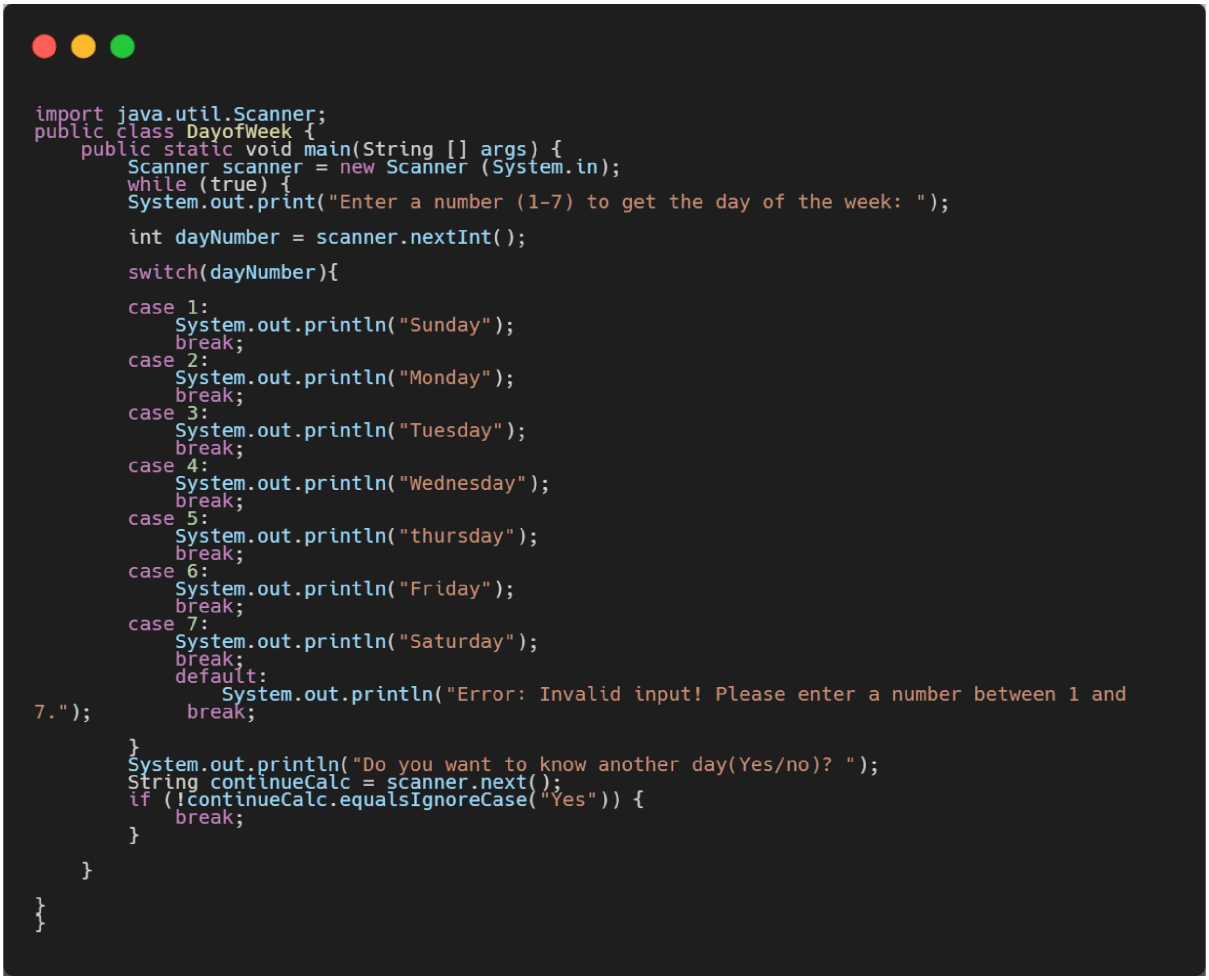
Output:



```
<terminated> Calculator [Java Application] C:\Users\sagar\.p2\pool\plugins\org.eclipse.justj.openj
Enter the first number: 12
Enter the second number: 14
Enter an operator (+, -, *, /): +
Addition Result: 26.0
Do you want to perform another calculation (yes/no)? yes
Enter the first number: 33
Enter the second number: 23
Enter an operator (+, -, *, /): *
Multiplication Result: 759.0
Do you want to perform another calculation (yes/no)? yes
Enter the first number: 45
Enter the second number: 0
Enter an operator (+, -, *, /): /
Error: Division by zero is not allowed.
Do you want to perform another calculation (yes/no)? no
Thank you for using the calculator. Goodbye!
```

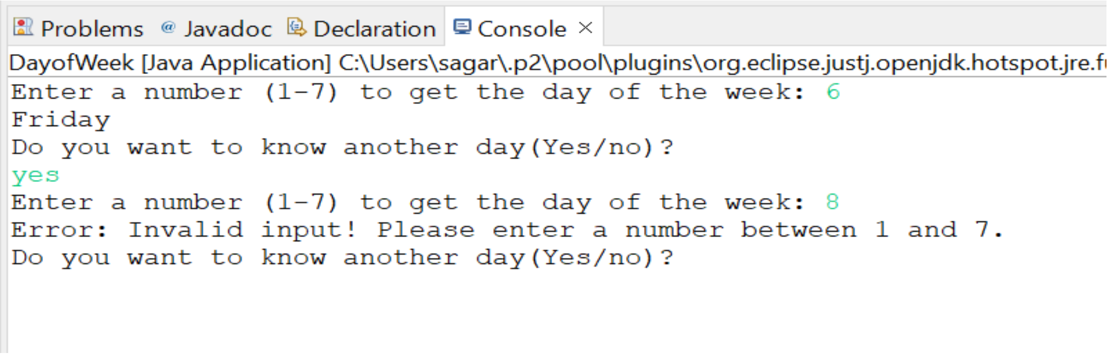
Q2) Create a program that accepts a number representing a day of the week (1-7) and uses a switch statement to print the corresponding day's name. Include handling for invalid inputs

Ans -



```
import java.util.Scanner;
public class DayofWeek {
    public static void main(String [] args) {
        Scanner scanner = new Scanner (System.in);
        while (true) {
        System.out.print("Enter a number (1-7) to get the day of the week: ");

        int dayNumber = scanner.nextInt();

        switch(dayNumber){

        case 1:
            System.out.println("Sunday");
            break;
        case 2:
            System.out.println("Monday");
            break;
        case 3:
            System.out.println("Tuesday");
            break;
        case 4:
            System.out.println("Wednesday");
            break;
        case 5:
            System.out.println("thursday");
            break;
        case 6:
            System.out.println("Friday");
            break;
        case 7:
            System.out.println("Saturday");
            break;
            default:
                System.out.println("Error: Invalid input! Please enter a number between 1 and
7.");        break;

        }
        System.out.println("Do you want to know another day(Yes/no)? ");
        String continueCalc = scanner.next();
        if (!continueCalc.equalsIgnoreCase("Yes")) {
            break;
        }

    }

}
}
```

OUTPUT:



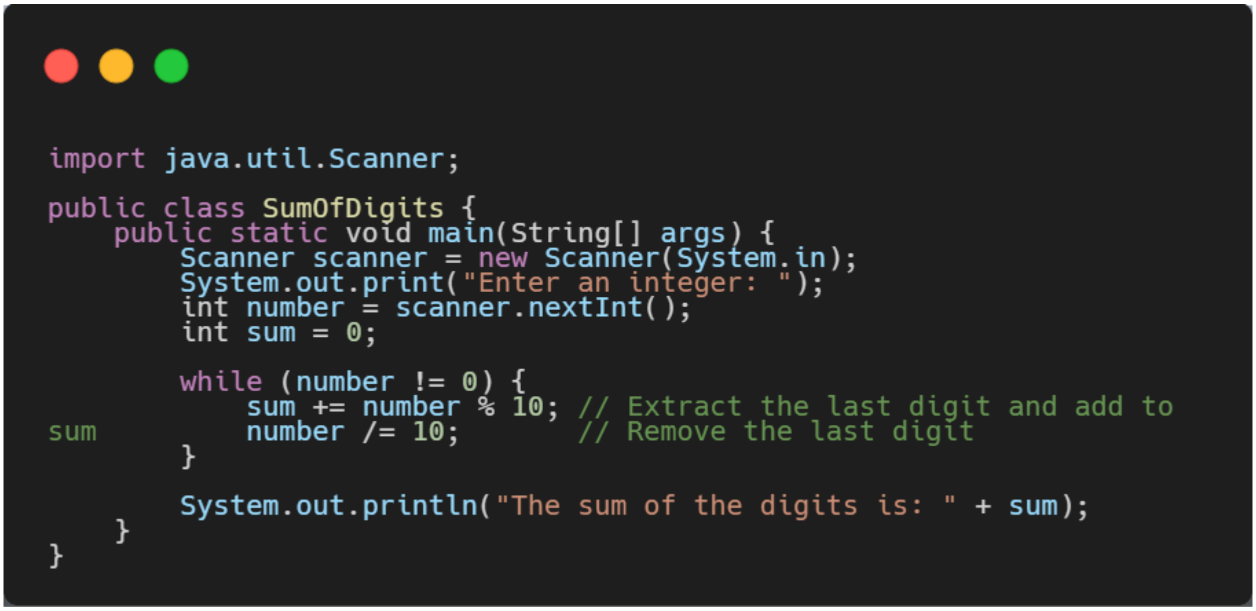
DayofWeek [Java Application] C:\Users\sagar\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.f

```
Enter a number (1-7) to get the day of the week: 6
Friday
Do you want to know another day(Yes/no)?
yes
Enter a number (1-7) to get the day of the week: 8
Error: Invalid input! Please enter a number between 1 and 7.
Do you want to know another day(Yes/no)?
```

2. While Loop

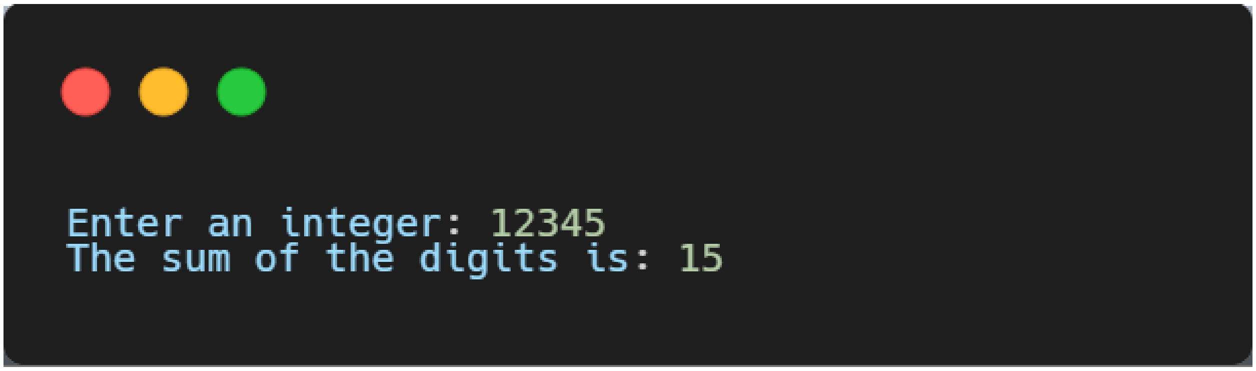
Write a program that calculates the sum of digits of a given integer using a while loop.

Ans -



```
import java.util.Scanner;

public class SumOfDigits {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter an integer: ");
        int number = scanner.nextInt();
        int sum = 0;

        while (number != 0) {
            sum += number % 10; // Extract the last digit and add to
sum         number /= 10;       // Remove the last digit
        }

        System.out.println("The sum of the digits is: " + sum);
    }
}
```

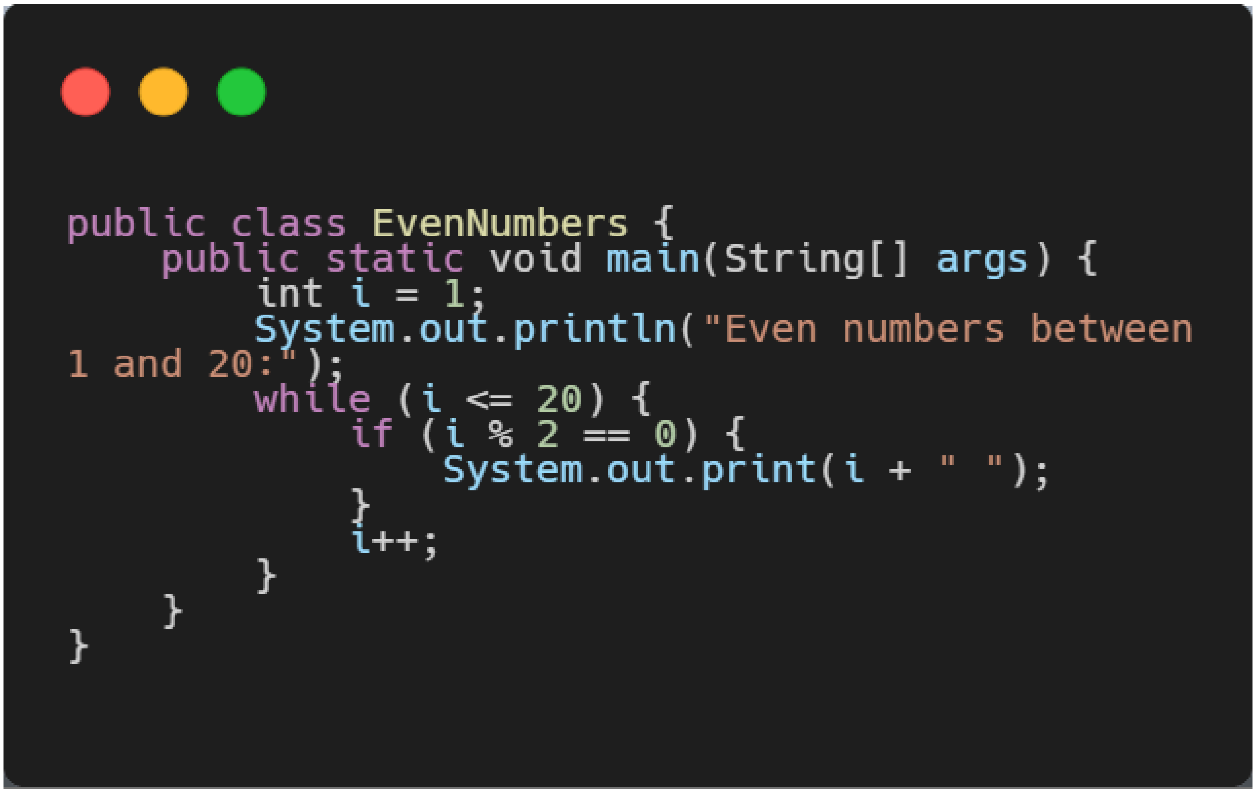
OUTPUT:



```
Enter an integer: 12345
The sum of the digits is: 15
```

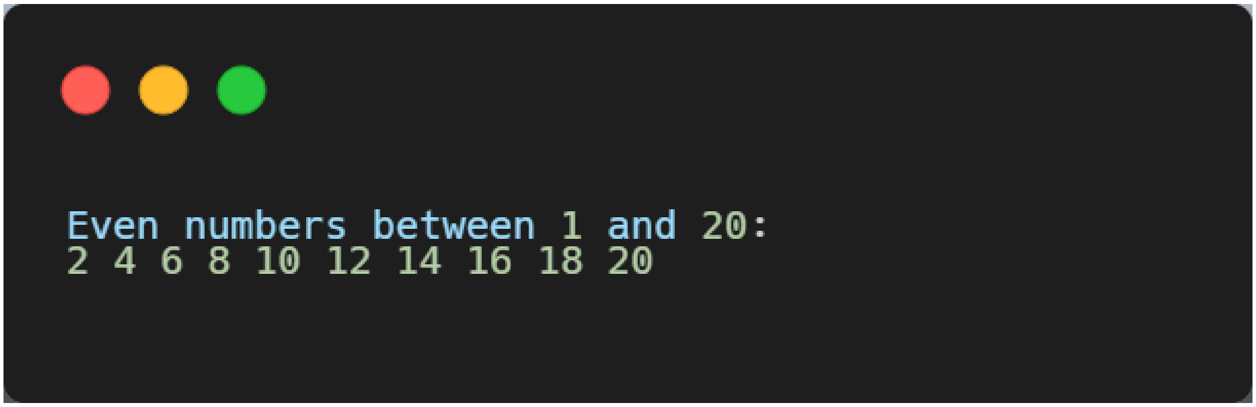
Develop a program to print all even numbers between 1 and 20 using a while loop

ANS -



```java
public class EvenNumbers {
    public static void main(String[] args) {
        int i = 1;
        System.out.println("Even numbers between 1 and 20:");
        while (i <= 20) {
            if (i % 2 == 0) {
                System.out.print(i + " ");
            }
            i++;
        }
    }
}
```

OUTPUT:

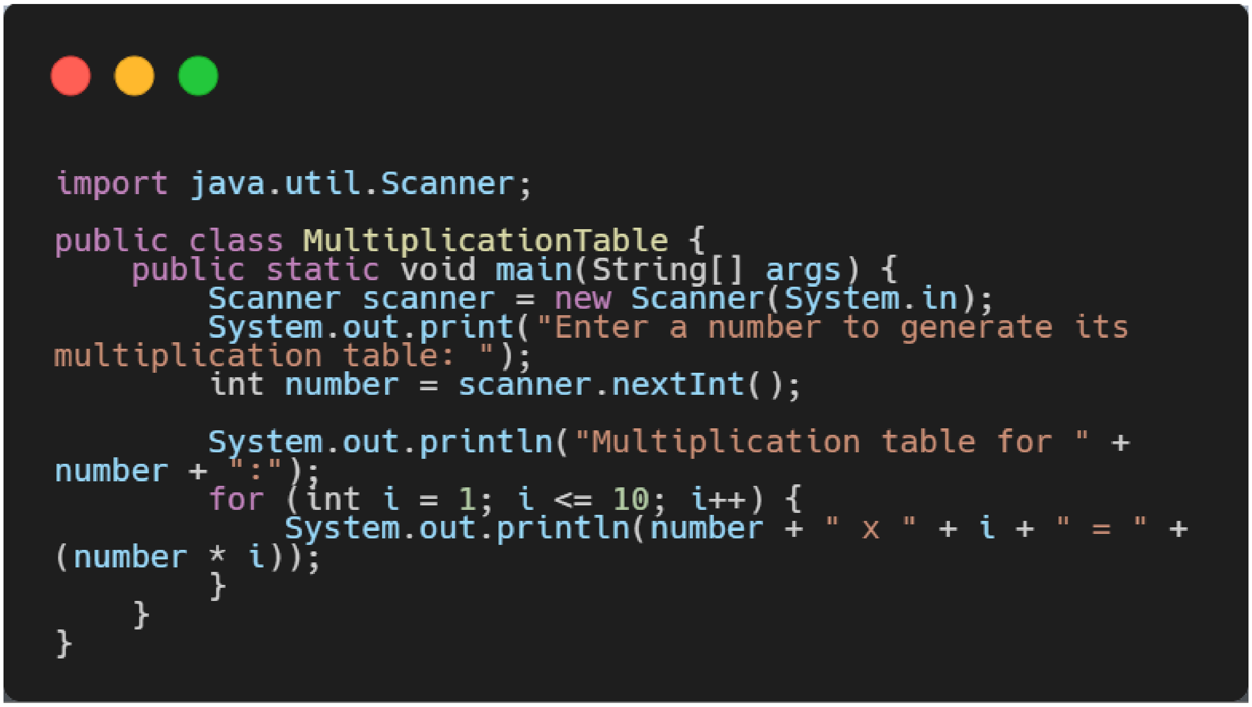


```
Even numbers between 1 and 20:
2 4 6 8 10 12 14 16 18 20
```

3. For Loop

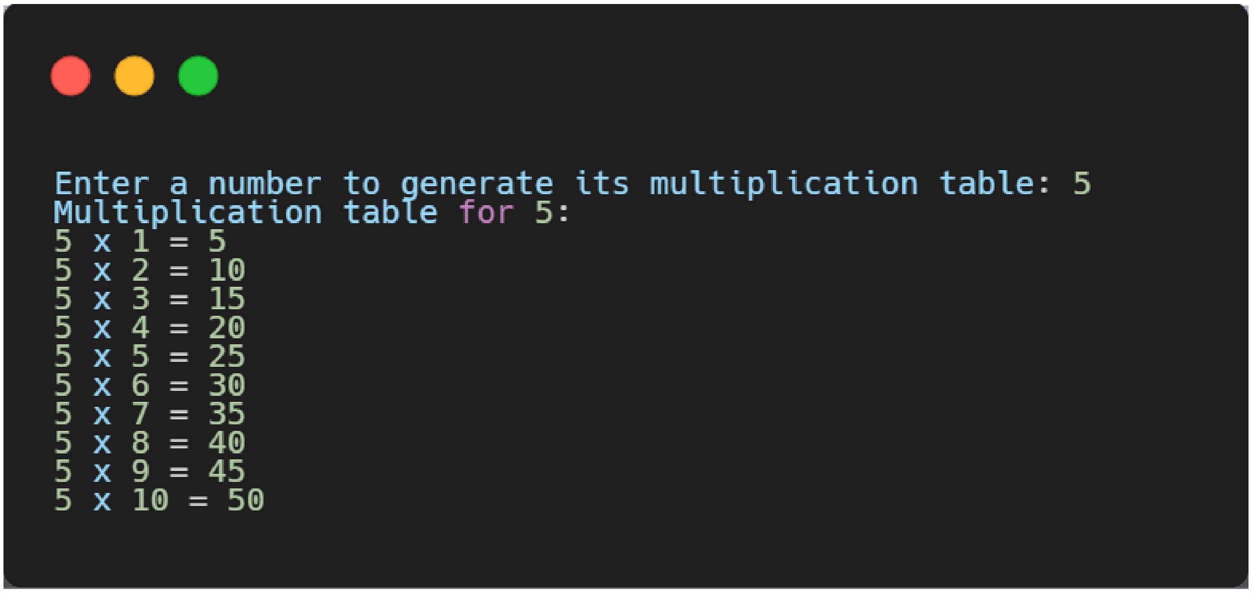
Write a program to generate the multiplication table of a number up to 10 using a for loop.

ANS -



```java
import java.util.Scanner;

public class MultiplicationTable {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number to generate its
multiplication table: ");
        int number = scanner.nextInt();

        System.out.println("Multiplication table for " +
number + ":");
        for (int i = 1; i <= 10; i++) {
            System.out.println(number + " x " + i + " = " +
(number * i));
        }
    }
}
```

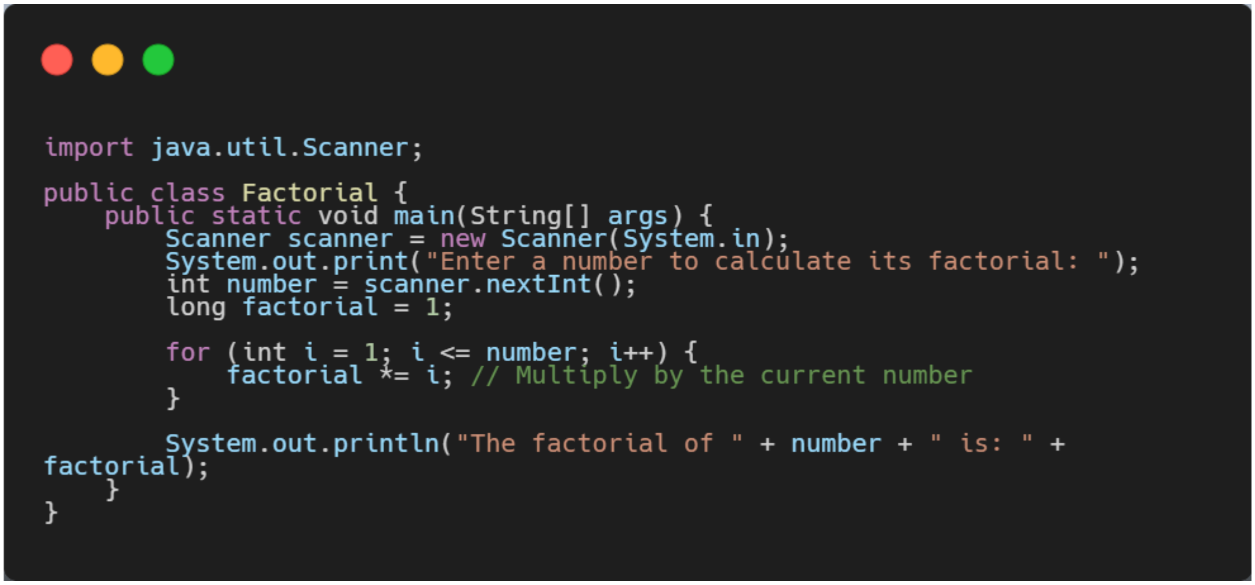
OUTPUT:



```
Enter a number to generate its multiplication table: 5
Multiplication table for 5:
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50
```

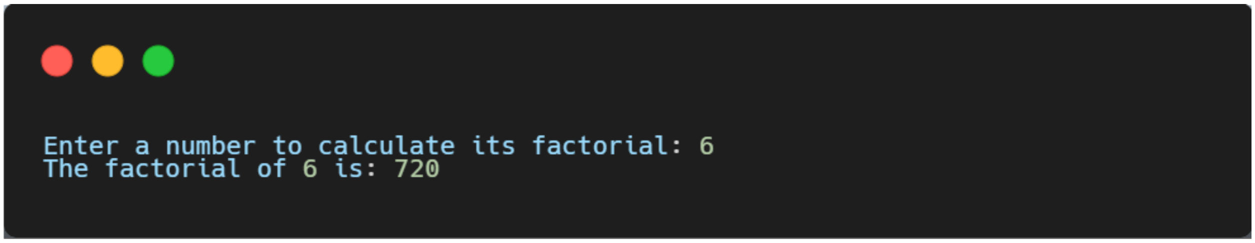
Implement a program to calculate the factorial of a given number using a for loop.

ANS -



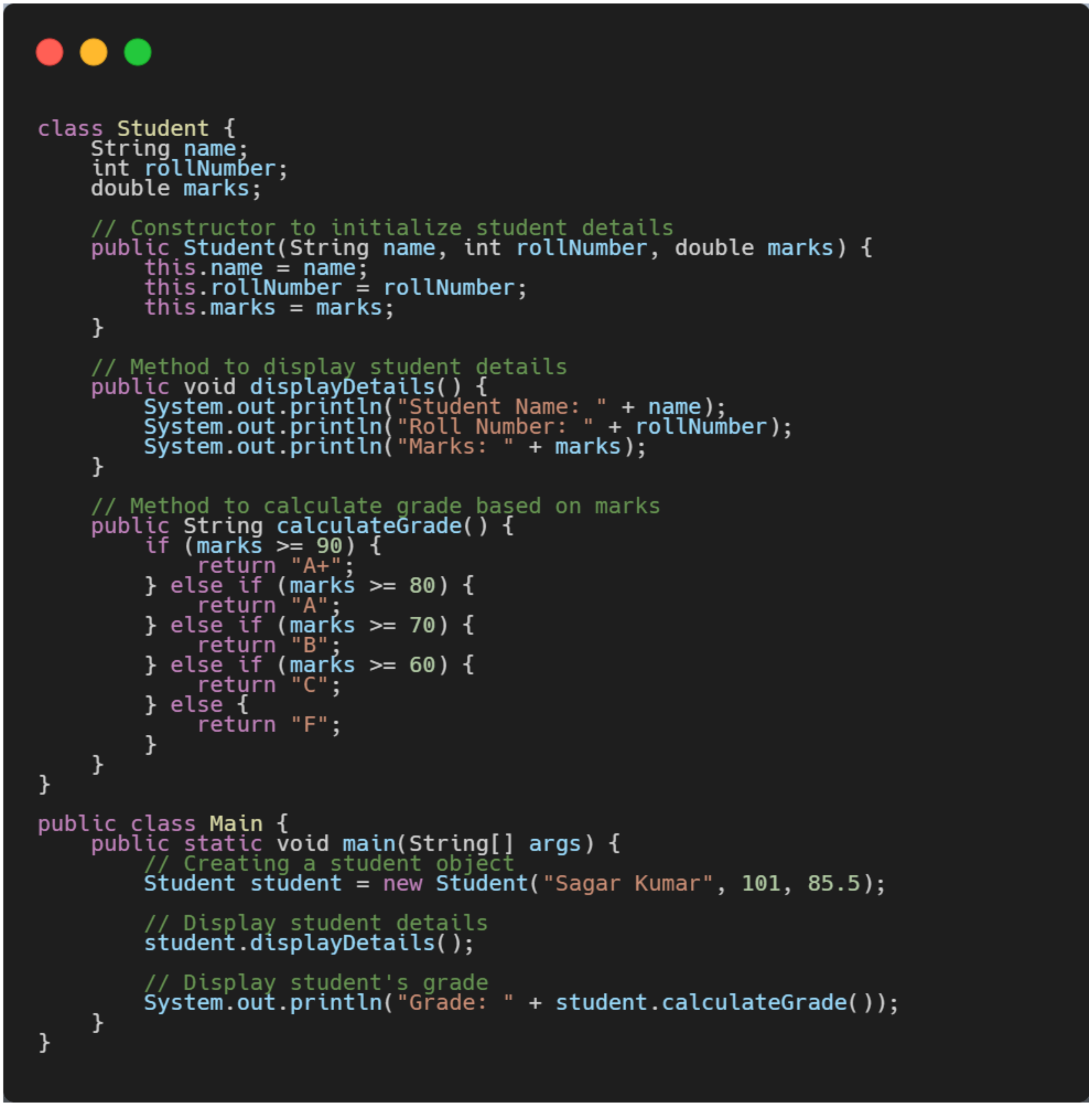
```java
import java.util.Scanner;

public class Factorial {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a number to calculate its factorial: ");
        int number = scanner.nextInt();
        long factorial = 1;

        for (int i = 1; i <= number; i++) {
            factorial *= i; // Multiply by the current number
        }

        System.out.println("The factorial of " + number + " is: " +
factorial);
    }
}
```

OUTPUT:



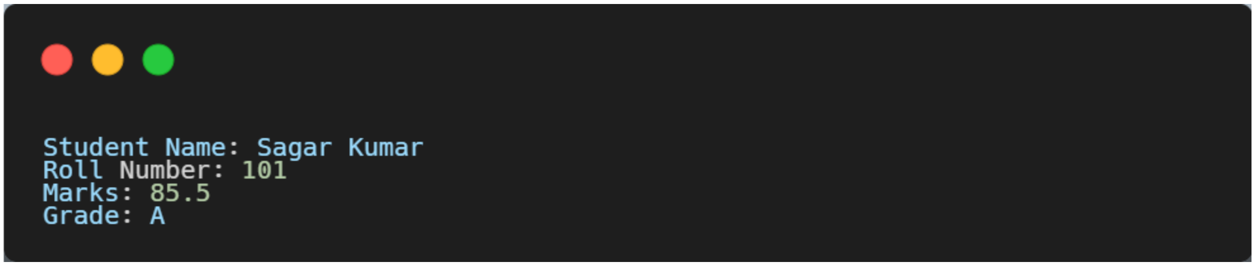
```
Enter a number to calculate its factorial: 6
The factorial of 6 is: 720
```

Create a class Student with attributes like name, rollNumber, and marks. Include methods to display the student's details and calculate their grade based on their marks.
ANS -



```java
class Student {
    String name;
    int rollNumber;
    double marks;

    // Constructor to initialize student details
    public Student(String name, int rollNumber, double marks) {
        this.name = name;
        this.rollNumber = rollNumber;
        this.marks = marks;
    }

    // Method to display student details
    public void displayDetails() {
        System.out.println("Student Name: " + name);
        System.out.println("Roll Number: " + rollNumber);
        System.out.println("Marks: " + marks);
    }

    // Method to calculate grade based on marks
    public String calculateGrade() {
        if (marks >= 90) {
            return "A+";
        } else if (marks >= 80) {
            return "A";
        } else if (marks >= 70) {
            return "B";
        } else if (marks >= 60) {
            return "C";
        } else {
            return "F";
        }
    }
}

public class Main {
    public static void main(String[] args) {
        // Creating a student object
        Student student = new Student("Sagar Kumar", 101, 85.5);

        // Display student details
        student.displayDetails();

        // Display student's grade
        System.out.println("Grade: " + student.calculateGrade());
    }
}
```

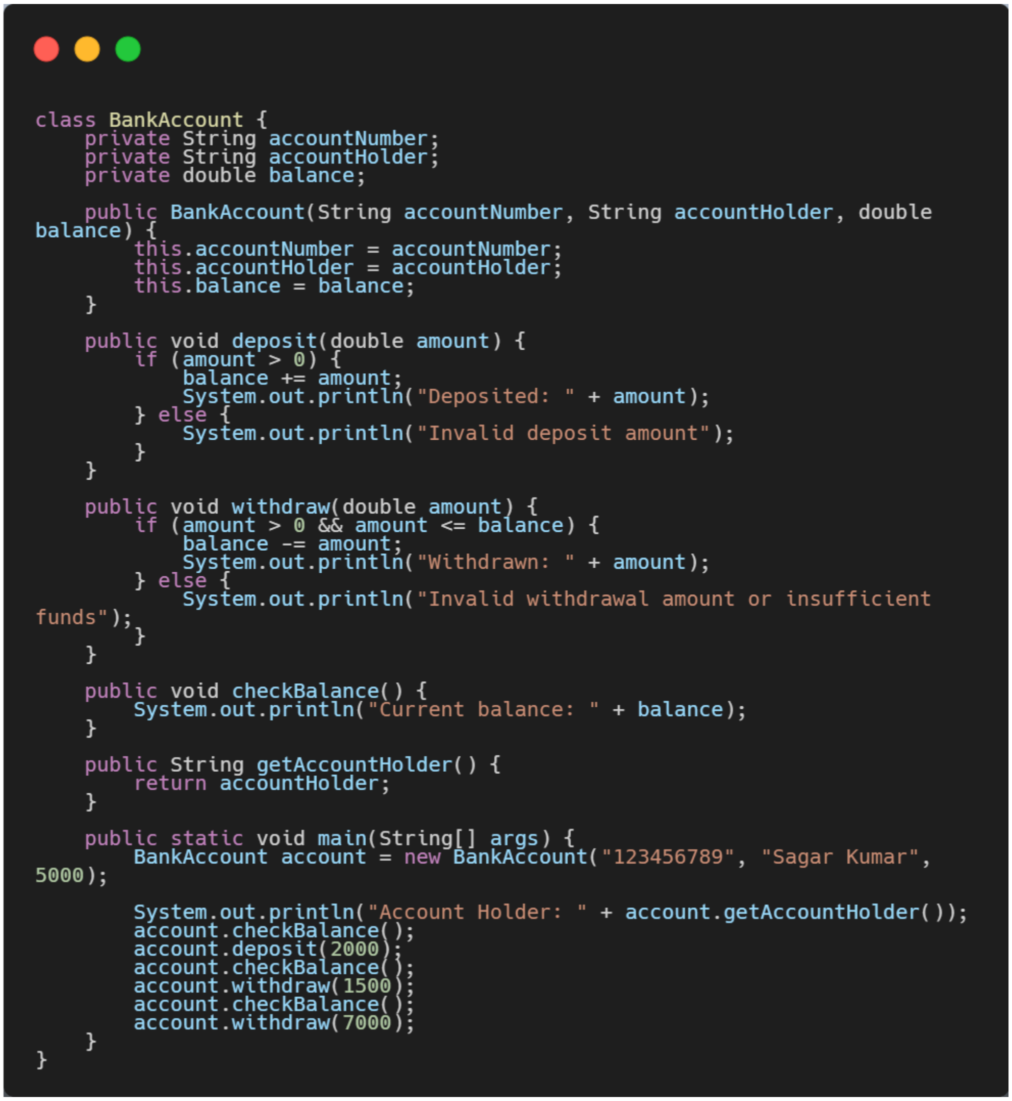
OUTPUT:



```
Student Name: Sagar Kumar
Roll Number: 101
Marks: 85.5
Grade: A
```

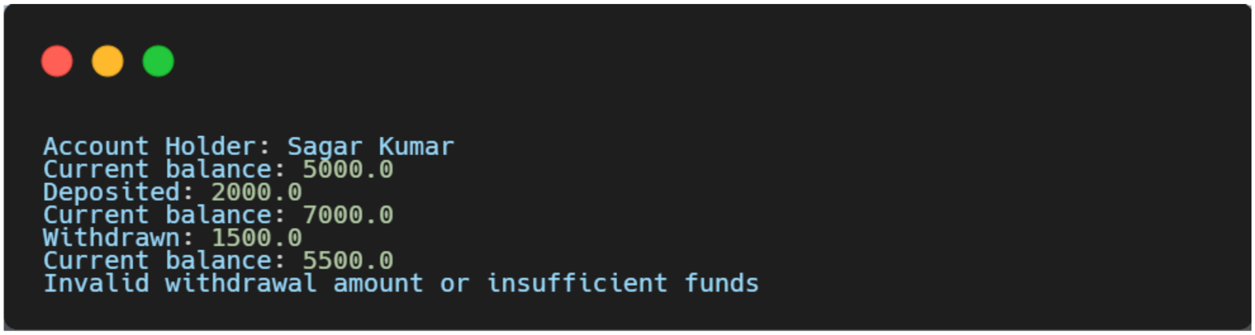
Define a class BankAccount with attributes like accountNumber, accountHolder, and balance. Add methods for depositing, withdrawing, and checking the balance. Demonstrate these functionalities in a program.

ANS-



```java
class BankAccount {
    private String accountNumber;
    private String accountHolder;
    private double balance;

    public BankAccount(String accountNumber, String accountHolder, double
balance) {
        this.accountNumber = accountNumber;
        this.accountHolder = accountHolder;
        this.balance = balance;
    }

    public void deposit(double amount) {
        if (amount > 0) {
            balance += amount;
            System.out.println("Deposited: " + amount);
        } else {
            System.out.println("Invalid deposit amount");
        }
    }

    public void withdraw(double amount) {
        if (amount > 0 && amount <= balance) {
            balance -= amount;
            System.out.println("Withdrawn: " + amount);
        } else {
            System.out.println("Invalid withdrawal amount or insufficient
funds");
        }
    }

    public void checkBalance() {
        System.out.println("Current balance: " + balance);
    }

    public String getAccountHolder() {
        return accountHolder;
    }

    public static void main(String[] args) {
        BankAccount account = new BankAccount("123456789", "Sagar Kumar",
5000);

        System.out.println("Account Holder: " + account.getAccountHolder());
        account.checkBalance();
        account.deposit(2000);
        account.checkBalance();
        account.withdraw(1500);
        account.checkBalance();
        account.withdraw(7000);
    }
}
```

OUTPUT:



```
Account Holder: Sagar Kumar
Current balance: 5000.0
Deposited: 2000.0
Current balance: 7000.0
Withdrawn: 1500.0
Current balance: 5500.0
Invalid withdrawal amount or insufficient funds
```