ASSIGNMENT-5.4

2303A51595
B-10

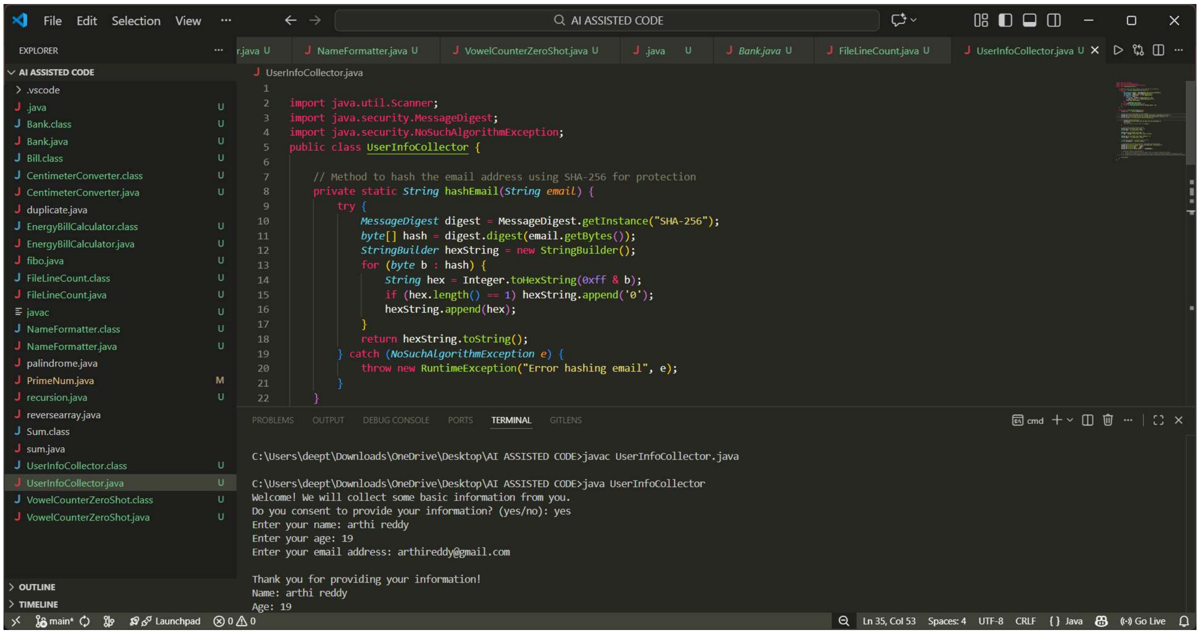
TASK-1

Prompt : Write a Java program to take user name, age, and email from input. Add comments on how to protect user data, like hashing email and not storing data openly.

Code :

```
import java.util.Scanner;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
public class UserInfoCollector {
   private static String hashEmail(String email) {
      try {
         MessageDigest digest = MessageDigest.getInstance("SHA-256");
         byte[] hash = digest.digest(email.getBytes());
         StringBuilder hexString = new StringBuilder();
         for (byte b : hash) {
            String hex = Integer.toHexString(0xff & b);
            if (hex.length() == 1) hexString.append('0');
            hexString.append(hex);
         }
         return hexString.toString();
      } catch (NoSuchAlgorithmException e) {
         throw new RuntimeException("Error hashing email", e);
      }
   }

   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      System.out.println("Welcome! We will collect some basic information from
you.");
      System.out.print("Do you consent to provide your information? (yes/no): ");
      String consent = sc.nextLine().trim().toLowerCase();

      if (!consent.equals("yes")) {
         System.out.println("Thank you! No data will be collected.");
         sc.close();
         return;
      }
      System.out.print("Enter your name: ");
      String name = sc.nextLine().trim();

      System.out.print("Enter your age: ");
      int age = Integer.parseInt(sc.nextLine().trim());

      System.out.print("Enter your email address: ");
      String email = sc.nextLine().trim();
      String hashedEmail = hashEmail(email);
      System.out.println("\nThank you for providing your information!");
      System.out.println("Name: " + name);
      System.out.println("Age: " + age);
      System.out.println("Hashed Email: " + hashedEmail);
      sc.close();
   }
}
```

Output :



```

import java.util.Scanner;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
public class UserInfoCollector {

    // Method to hash the email address using SHA-256 for protection
    private static String hashEmail(String email) {
        try {
            MessageDigest digest = MessageDigest.getInstance("SHA-256");
            byte[] hash = digest.digest(email.getBytes());
            StringBuilder hexString = new StringBuilder();
            for (byte b : hash) {
                String hex = Integer.toHexString(0xff & b);
                if (hex.length() == 1) hexString.append('0');
                hexString.append(hex);
            }
            return hexString.toString();
        } catch (NoSuchAlgorithmException e) {
            throw new RuntimeException("Error hashing email", e);
        }
    }
```

```
C:\Users\deept\Downloads\OneDrive\Desktop\AI ASSISTED CODE>javac UserInfoCollector.java

C:\Users\deept\Downloads\OneDrive\Desktop\AI ASSISTED CODE>java UserInfoCollector
Welcome! We will collect some basic information from you.
Do you consent to provide your information? (yes/no): yes
Enter your name: arthi reddy
Enter your age: 19
Enter your email address: arthireddy@gmail.com

Thank you for providing your information!
Name: arthi reddy
Age: 19
```

Analysis :

In this task, we collect user details like name, age, and email. User data is private, so it should not be stored openly. Email can be protected by hashing. Collecting only required data helps in keeping user information safe. Developers are responsible for protecting user privacy.

TASK-2

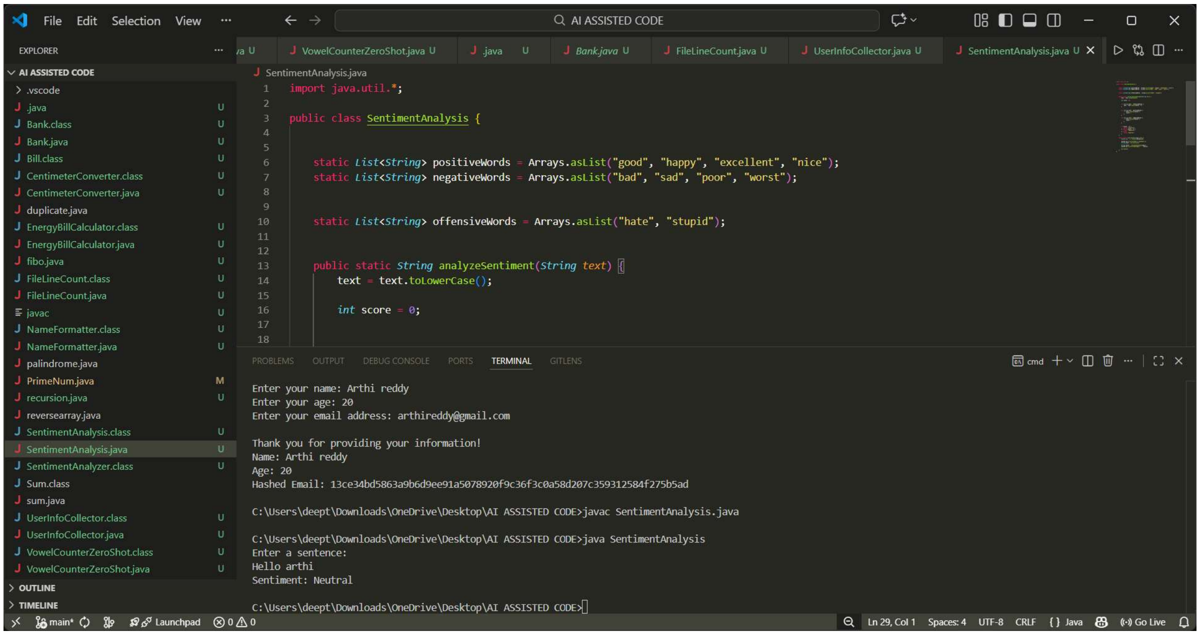
Prompt : Generate a Java program for sentiment analysis (positive, negative, neutral).

Code :

```
import java.util.*;
public class SentimentAnalysis {

   static List<String> positiveWords = Arrays.asList("good", "happy", "excellent",
"nice");
   static List<String> negativeWords = Arrays.asList("bad", "sad", "poor", "worst");

   static List<String> offensiveWords = Arrays.asList("hate", "stupid");

   public static String analyzeSentiment(String text) {
      text = text.toLowerCase();

      int score = 0;

      for (String word : offensiveWords) {
         text = text.replace(word, "");
      }


      for (String word : positiveWords) {
         if (text.contains(word)) {
            score++;
         }
      }

      for (String word : negativeWords) {
         if (text.contains(word)) {
            score--;
         }
      }
```

```
        if (score > 0) {
            return "Positive";
        } else if (score < 0) {
            return "Negative";
        } else {
            return "Neutral";
        }
    }

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

        System.out.println("Enter a sentence:");
        String input = sc.nextLine();

        String result = analyzeSentiment(input);
        System.out.println("Sentiment: " + result);

        sc.close();
    }
}
```

Output :



```
import java.util.*;

public class SentimentAnalysis {


    static List<String> positiveWords = Arrays.asList("good", "happy", "excellent", "nice");
    static List<String> negativeWords = Arrays.asList("bad", "sad", "poor", "worst");


    static List<String> offensiveWords = Arrays.asList("hate", "stupid");


    public static String analyzeSentiment(String text) {
        text = text.toLowerCase();

        int score = 0;


```

```
Enter your name: Arthi reddy
Enter your age: 20
Enter your email address: arthireddy@gmail.com

Thank you for providing your information!
Name: Arthi reddy
Age: 20
Hashed Email: 13ce34bd5863a9b6d9ee91a5078920f9c36f3c0a58d207c359312584f275b5ad

C:\Users\deept\Downloads\OneDrive\Desktop\AI ASSISTED CODE>javac SentimentAnalysis.java

C:\Users\deept\Downloads\OneDrive\Desktop\AI ASSISTED CODE>java SentimentAnalysis
Enter a sentence:
Hello arthi
Sentiment: Neutral

C:\Users\deept\Downloads\OneDrive\Desktop\AI ASSISTED CODE>
```

Analysis :

This program finds whether the given text is positive, negative, or neutral. Bias can occur because of offensive words or unbalanced data. To reduce this, offensive words are removed and both positive and negative words are treated equally.

TASK-3

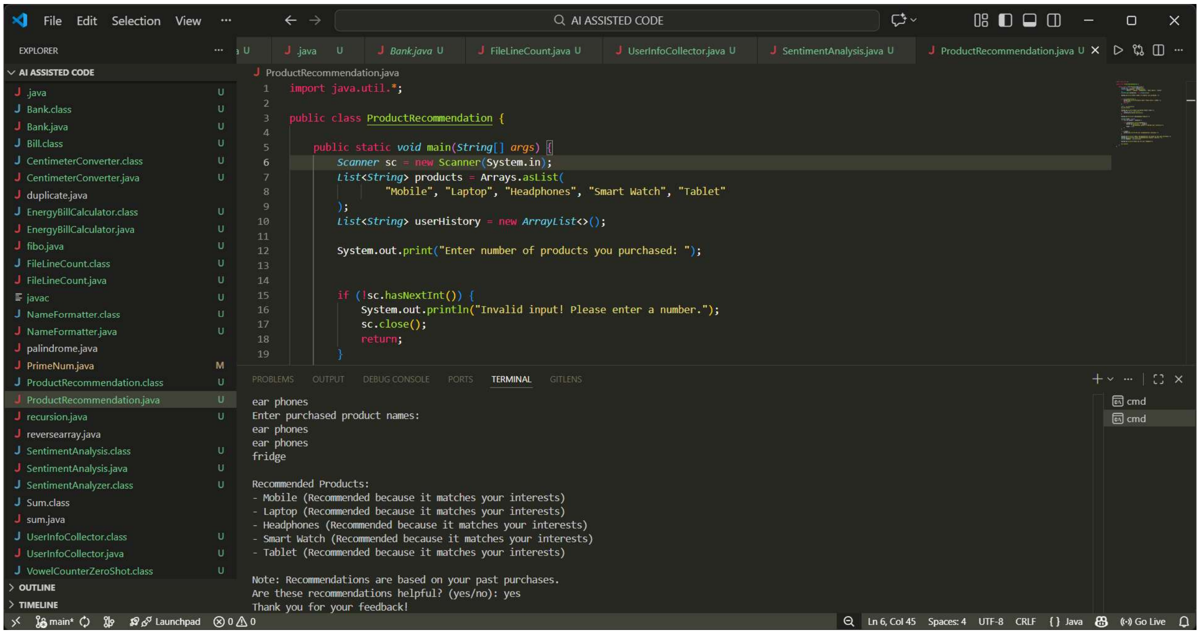
Prompt : Write a Java program that recommends products based on user purchase history.

Code :

```
import java.util.*;

public class ProductRecommendation {

  public static void main(String[] args) {
    Scanner sc = new Scanner(System.in);
    List<String> products = Arrays.asList(
        "Mobile", "Laptop", "Headphones", "Smart Watch", "Tablet"
    );
    List<String> userHistory = new ArrayList<>();

    System.out.print("Enter number of products you purchased: ");


    if (!sc.hasNextInt()) {
      System.out.println("Invalid input! Please enter a number.");
      sc.close();
      return;
    }

    int n = sc.nextInt();
    sc.nextLine();

    System.out.println("Enter purchased product names:");
    for (int i = 0; i < n; i++) {
      userHistory.add(sc.nextLine());
```

```java
    }

    System.out.println("\nRecommended Products:");

    boolean found = false;
    for (String product : products) {
    if (!found) {
      System.out.println("No new recommendations available.");
    }

    System.out.println("\nNote: Recommendations are based on your past purchases.");
    System.out.print("Are these recommendations helpful? (yes/no): ");
    String feedback = sc.nextLine();

    System.out.println("Thank you for your feedback!");

    sc.close();
  }
}
```

Output :



```java
import java.util.*;

public class ProductRecommendation {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        List<String> products = Arrays.asList(
                "Mobile", "Laptop", "Headphones", "Smart Watch", "Tablet"
        );
        List<String> userHistory = new ArrayList<>();

        System.out.print("Enter number of products you purchased: ");


        if (!sc.hasNextInt()) {
            System.out.println("Invalid input! Please enter a number.");
            sc.close();
            return;
        }
```

```
ear phones
Enter purchased product names:
ear phones
ear phones
fridge

Recommended Products:
- Mobile (Recommended because it matches your interests)
- Laptop (Recommended because it matches your interests)
- Headphones (Recommended because it matches your interests)
- Smart Watch (Recommended because it matches your interests)
- Tablet (Recommended because it matches your interests)

Note: Recommendations are based on your past purchases.
Are these recommendations helpful? (yes/no): yes
Thank you for your feedback!
```

Analysis :

The program recommends products based on the user's previous purchases. It avoids repeating already bought products and clearly explains why recommendations are shown, making it fair and transparent.

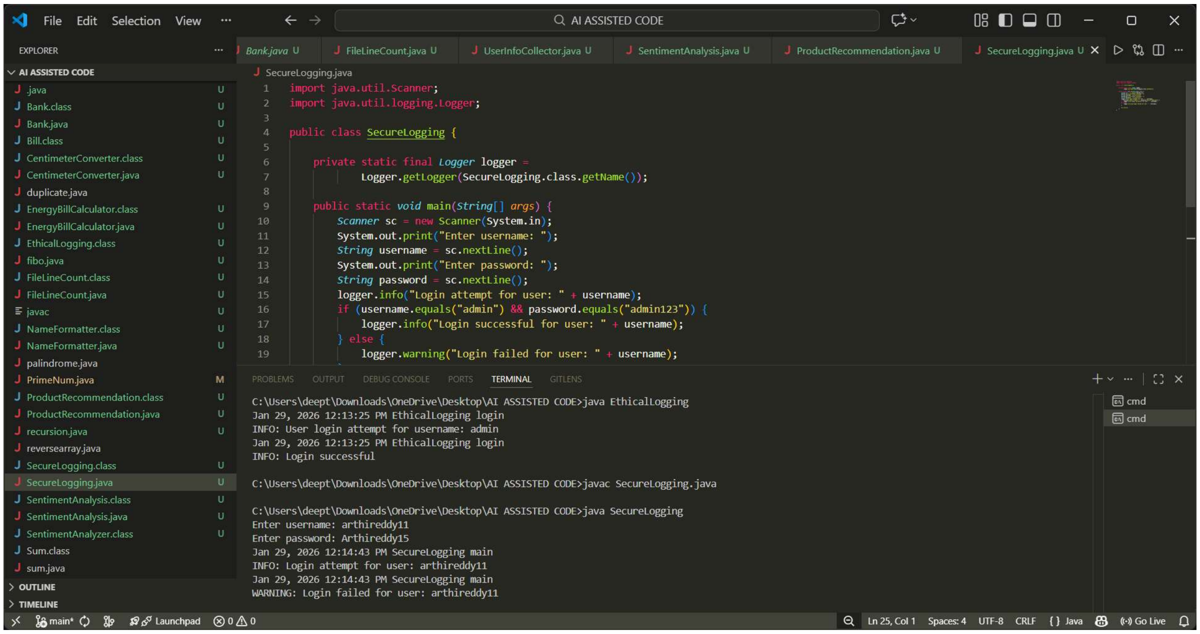
TASK-4

Prompt : Write a Java program with logging functionality for an application. Make sure the logs do not store sensitive data like passwords or email IDs and add comments explaining safe logging.

Code :

```
import java.util.Scanner;
import java.util.logging.Logger;

public class SecureLogging {

    private static final Logger logger =
            Logger.getLogger(SecureLogging.class.getName());

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter username: ");
        String username = sc.nextLine();
        System.out.print("Enter password: ");
        String password = sc.nextLine();
        logger.info("Login attempt for user: " + username);
        if (username.equals("admin") &&
password.equals("admin123")) {
            logger.info("Login successful for user: " + username);
        } else {
            logger.warning("Login failed for user: " + username);
        }

        sc.close();
    }
}
```

Output :



SecureLogging.java

```java
import java.util.Scanner;
import java.util.logging.Logger;

public class SecureLogging {

    private static final Logger logger =
            Logger.getLogger(SecureLogging.class.getName());

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter username: ");
        String username = sc.nextLine();
        System.out.print("Enter password: ");
        String password = sc.nextLine();
        logger.info("Login attempt for user: " + username);
        if (username.equals("admin") && password.equals("admin123")) {
            logger.info("Login successful for user: " + username);
        } else {
            logger.warning("Login failed for user: " + username);
```

```
C:\Users\deept\Downloads\OneDrive\Desktop\AI ASSISTED CODE>java EthicalLogging
Jan 29, 2026 12:13:25 PM EthicalLogging login
INFO: User login attempt for username: admin
Jan 29, 2026 12:13:25 PM EthicalLogging login
INFO: Login successful

C:\Users\deept\Downloads\OneDrive\Desktop\AI ASSISTED CODE>javac SecureLogging.java

C:\Users\deept\Downloads\OneDrive\Desktop\AI ASSISTED CODE>java SecureLogging
Enter username: arthireddy11
Enter password: Arthireddy15
Jan 29, 2026 12:14:43 PM SecureLogging main
INFO: Login attempt for user: arthireddy11
Jan 29, 2026 12:14:43 PM SecureLogging main
WARNING: Login failed for user: arthireddy11
```

Analysis :

This program demonstrates ethical logging by recording only necessary application events and avoiding sensitive data such as passwords. It helps in debugging while ensuring user privacy and data security

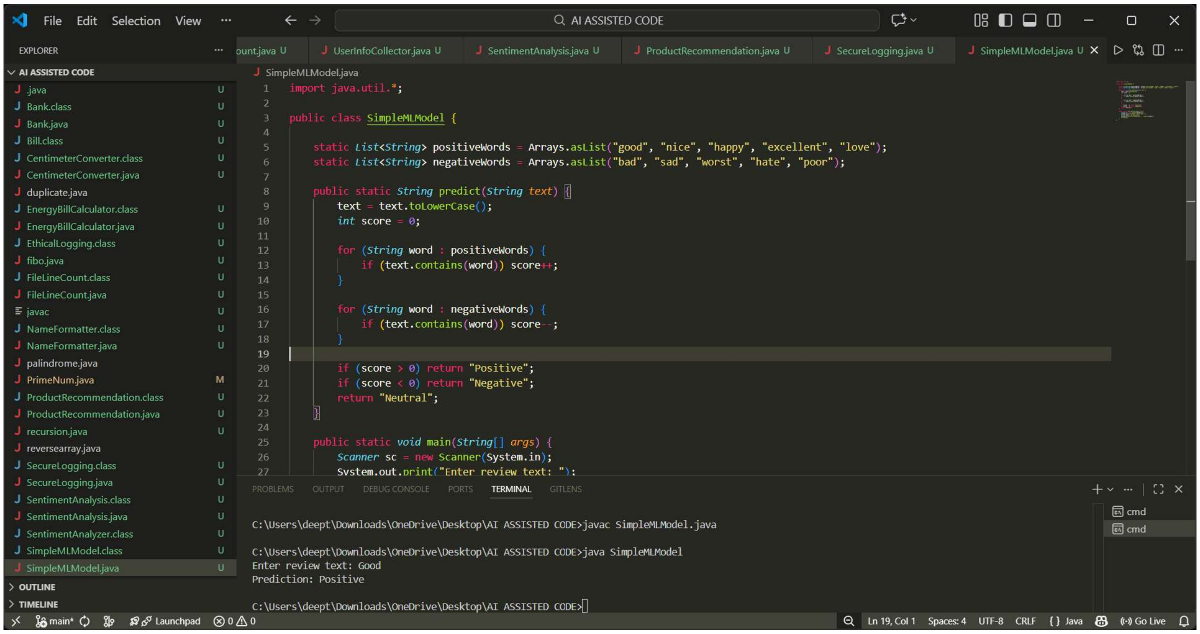
TASK-5

Prompt : Write a Java program that takes user input and predicts sentiment like a simple machine learning model.

Code :

```java
import java.util.*;

public class SimpleMLModel {

    static List<String> positiveWords = Arrays.asList("good", "nice", "happy", 
"excellent", "love");
    static List<String> negativeWords = Arrays.asList("bad", "sad", "worst", "hate", 
"poor");

    public static String predict(String text) {
        text = text.toLowerCase();
        int score = 0;

        for (String word : positiveWords) {
            if (text.contains(word)) score++;
        }

        for (String word : negativeWords) {
            if (text.contains(word)) score--;
        }

        if (score > 0) return "Positive";
        if (score < 0) return "Negative";
        return "Neutral";
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter review text: ");
        String input = sc.nextLine();
        System.out.println("Prediction: " + predict(input));
        sc.close();
```

Output :



SimpleMLModel.java

```java
import java.util.*;

public class SimpleMLModel {

    static List<String> positiveWords = Arrays.asList("good", "nice", "happy", "excellent", "love");
    static List<String> negativeWords = Arrays.asList("bad", "sad", "worst", "hate", "poor");

    public static String predict(String text) {
        text = text.toLowerCase();
        int score = 0;

        for (String word : positiveWords) {
            if (text.contains(word)) score++;
        }

        for (String word : negativeWords) {
            if (text.contains(word)) score--;
        }

        if (score > 0) return "Positive";
        if (score < 0) return "Negative";
        return "Neutral";
    }

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print("Enter review text: ");
```

```
C:\Users\deept\Downloads\OneDrive\Desktop\AI ASSISTED CODE>javac SimpleMLModel.java

C:\Users\deept\Downloads\OneDrive\Desktop\AI ASSISTED CODE>java SimpleMLModel
Enter review text: Good
Prediction: Positive

C:\Users\deept\Downloads\OneDrive\Desktop\AI ASSISTED CODE>
```

Analysis :

This program takes user input and predicts sentiment using a simple model. Since the model is basic and may not be fully accurate, the results should be checked by humans and used responsibly.