```
import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;
import java.util.ArrayList;
import java.util.List;
import java.util.Scanner;
import java.io.BufferedReader;
import java.io.FileReader;

public class Main {

    public static void main(String[] args) {
        //file paths for newly created CSV files with the prefacture
numbers and the status change dates inside
        //For different files changing the file path is necessary
        String file1Path = "/Users/grantjones/Downloads/Clinica UAndes
Work/insuracneComparator/src/resources/05-05-2023 Fecha and Pre
Factura.csv";
        String file2Path = "/Users/grantjones/Downloads/Clinica UAndes
Work/insuracneComparator/src/resources/11-05-2023 Fecha and Pre
Factura .csv";
        //Calls read file method on file paths
        insuranceComparator<String, String> file1Map =
readFile(file1Path);
        insuranceComparator<String, String> file2Map =
readFile(file2Path);


        //Calls method to find same prefacture numbers with differing
date status
      List<String> differingKeys = findDifferingKeys(file1Map,
file2Map);

      String csvFile = "/Users/grantjones/Downloads/Clinica UAndes
Work/insuracneComparator/src/resources/Prefacture, Name, Episode, and
Email.csv";
      readCSVFile(csvFile);

         if (differingKeys.isEmpty()) {
            //hashtable is empty
            System.out.println("No differences found");
        } else {
            System.out.println();
            System.out.println("Prefacture Numbers of people whos
status has changed");
            System.out.println();

            for (String key : differingKeys) {
                //Prints the prefacture number of people whos date
status has changes
```

```
                System.out.println(key);
            }
        }
    }
    //method to read the attached CSV files
    private static insuranceComparator<String, String> readFile(String
filePath) {
        insuranceComparator<String, String> map = new
insuranceComparator<>();

        try (FileInputStream fis = new FileInputStream(new
File(filePath))) {
            Scanner scanner = new Scanner(fis);

            // Skip the header row if needed
            if (scanner.hasNextLine()) {
                scanner.nextLine();
            }

            // Read each line and extract key-value pairs
            while (scanner.hasNextLine()) {
                String line = scanner.nextLine();

                // Split the line into two fields
                String[] fields = line.split(",");

                // Ensure that the line has two fields
                if (fields.length == 2) {
                    String key = fields[0].trim();
                    String value = fields[1].trim();

                    map.put(key, value);
                }
            }

            scanner.close();
        } catch (IOException e) {
            e.printStackTrace();
        }

        return map;
    }

    //method to find the prefacutre numbers with differing date status
    private static List<String>
findDifferingKeys(insuranceComparator<String, String> map1,

insuranceComparator<String, String> map2) {
        //Initializes an arraylist to pass the prefacture numbers
needed to be returned to
```

```
        List<String> differingKeys = new ArrayList<>();
        //Creates an array of Entry<K,V> objects using the hashtable
of the data from 11-05-2023
        Entry<String, String>[] hashTable2 = map2.getHashTable();
        //iterates through the data
        for (int i = 0; i < map2.getSize(); i++) {
            //Creates an Entry<K,V> object for each piece of data in
the file
            Entry<String, String> entry = hashTable2[i];
            if (entry != null) {
                //Prefacture number of object from 11-05-2023
                String key = entry.getPreNum();
                //corresponding date status
                String value1 = entry.getFecha();
                //date status with the same prefacture number from
05-05-2023 file
                String value2 = map1.get(key);

                //compares the values of the dates and if they are the
same the loop continues
                //If they are not the same it adds them to the
arraylist
                if (value2 != null && !value1.equals(value2)) {
                    differingKeys.add(key);
                }
            }
        }
        //returns the arraylist of prefacture numbers of people to be
emailed
        return differingKeys;
    }

    public static void readCSVFile(String csvFile) {
        String line;

        try (BufferedReader br = new BufferedReader(new
FileReader(csvFile))) {
            while ((line = br.readLine()) != null) {
                // Split the line by the CSV delimiter (usually comma)
                String[] data = line.split(",");

                // Process the data for each row
                for (String value : data) {
                    System.out.print(value + "  \t");
                }
                System.out.println();
                System.out.println();
            }
        } catch (IOException e) {
            e.printStackTrace();
```

```
        }
    }


    // Helper method to pad a string with spaces on the right
    private static String padRight(String s, int n) {
        return String.format("%-" + n + "s", s);
    }


}
```