/\*\*\*\*

\* Name: Manik Singh

\* Student Number: T00715263

\* Assignment Number: 4

\* Due Date: October 24, 2023

\* Program Description: This file has the ValidNumbers class that reads data from

\* numbers.txt text file then verifies it and stores it in valid.txt or

\* invalid.txt based on the data.

\*\*\*\*/

import java.io.\*;

import java.util.Scanner;

public class ValidNumbers {

public static void main(String[] args) throws IOException /\* IOException which is a checked exception gets thrown to the operating system. \*/

{

//variables declaration

String nextNumber = "";

String areaCode;

String prefix;

String lineNumber;

int firstHyphen = 0;

int secondHyphen = 0;

// Create a Scanner to read from the input file

File file = new File("numbers.txt");

Scanner input = new Scanner(file);

// Create PrintWriter object for valid and invalid numbers

PrintWriter validNumbers = new PrintWriter("valid.txt");

PrintWriter invalidNumbers = new PrintWriter("invalid.txt");

//Using a while loop to read data from input file till the end of file

while (input.hasNext()) {

try { //try block initialisation

nextNumber = input.nextLine();

firstHyphen = nextNumber.indexOf("-");

//check for hyphens and throwing HyphenException

if (firstHyphen < 0) {

throw new HyphenException(nextNumber + " has missing hyphens.\n");

}

secondHyphen = nextNumber.indexOf("-",firstHyphen + 1);

if (secondHyphen < 0) {

throw new HyphenException(nextNumber + " has missing hyphens.\n");

}

/\* Storing different parts of numbers using substring and then trimming white space \*/

areaCode = nextNumber.substring(0, firstHyphen);

areaCode = areaCode.trim();

prefix = nextNumber.substring(firstHyphen + 1, secondHyphen);

prefix = prefix.trim();

lineNumber = nextNumber.substring(secondHyphen + 1);

lineNumber = lineNumber.trim();

checkArea(areaCode); // Check the area code using checkArea method

checkPrefix(prefix); // Check the prefix using prefixArea method

if(lineNumber.length() != 4) /\* Check the length of line code and throwing LineException \*/

{

throw new LineException("Line error: " + lineNumber);

}

//Print statements for valid file

validNumbers.println("Testing this Code: " + nextNumber);

validNumbers.println("Code is valid");

validNumbers.println("");

}

//catch the exceptions thrown in the try block

catch (HyphenException h)

{

invalidNumbers.println("Testing this Code: " + nextNumber);

invalidNumbers.println("Hyphen error: " + h.getMessage()+ "\n");

}

catch (NumberFormatException nfe) /\* NumberFormatException is already existing java exception class. So we will just catch it. \*/

{

invalidNumbers.println("Testing this Code: " + nextNumber);

invalidNumbers.println("Number format error. Number provided " + nextNumber + " contains letters. Please add numbers only \n");

}

catch (AreaException a)

{

invalidNumbers.println("Testing this Code: " + nextNumber);

invalidNumbers.println("Area code error: " + a.getMessage()+ "\n");

}

catch (PrefixException p)

{

invalidNumbers.println("Testing this Code: " + nextNumber);

invalidNumbers.println("Prefix error: " + p.getMessage() + "\n");

}

catch (LineException l)

{

invalidNumbers.println("Testing this Code: " + nextNumber);

invalidNumbers.println("Line error: " + l.getMessage() + "\n");

}

}

// Print statement for terminal

System.out.println("Numbers verified and sent to Valid and Invalid Files.");

// Close the input, valid, and invalid files

input.close();

validNumbers.close();

invalidNumbers.close();

}

// Helper method to check if the area code is valid

public static void checkArea(String areaCode) throws AreaException

{

if (areaCode.length() != 3 || !isValidAreaCode(areaCode)) {

throw new AreaException(areaCode);

}

}

// Helper method to check if the prefix code is valid

public static void checkPrefix(String prefix) throws PrefixException

{

if (prefix.length() != 3 || !isValidPrefix(prefix)) {

throw new PrefixException(prefix);

}

}

public static boolean isValidAreaCode(String areaCode) {

int[] validAreaCodes = {236, 250, 672, 778};

int areaNumber = Integer.parseInt(areaCode);

for (int i = 0; i < validAreaCodes.length; i++) {

if (areaNumber == validAreaCodes[i]) {

return true;

}

}

return false;

}

public static boolean isValidPrefix(String prefix) {

int[] validPrefixes = {214, 220, 252, 289, 299, 314, 377, 828, 471, 371};

int prefixNumber = Integer.parseInt(prefix);

for (int i = 0; i < validPrefixes.length; i++) {

if (prefixNumber == validPrefixes[i]) {

return true;

}

}

return false;

}

}

/\*\*\*\*

\* Name: Manik Singh

\* Student Number: T00715263

\* Assignment Number: 4

\* Due Date: October 24, 2023

\* Program Description: This file contains the HyphenException class which is a

\* child class of the Exception class. It will throw a custom exception if the

\* input has provided invalid hyphens.

\*\*\*\*/

public class HyphenException extends Exception

{

public HyphenException()

{

super(" The number format provided is incorrect. Hyphen/s is/are missing ");

}

public HyphenException(String message)

{

super(message + " number format provided is incorrect. Hyphen/s is/are missing ");

}

}

/\*\*\*\*

\* Name: Manik Singh

\* Student Number: T00715263

\* Assignment Number: 4

\* Due Date: October 24, 2023

\* Program Description: This file contains the LineException class which is a

\* child class of the Exception class. It will throw a custom exception if the

\* input has provided invalid line code.

\*\*\*\*/

public class LineException extends Exception

{

public LineException()

{

super(" Line code is invalid ");

}

public LineException(String message)

{

super(message + " Line code is invalid ");

}

}

/\*\*\*\*

\* Name: Manik Singh

\* Student Number: T00715263

\* Assignment Number: 4

\* Due Date: October 24, 2023

\* Program Description: This file contains the PrefixException class which is a

\* child class of the Exception class. It will throw a custom exception if the

\* input has provided invalid prefix code.

\*\*\*\*/

public class PrefixException extends Exception

{

public PrefixException()

{

super(" Prefix code is invalid ");

}

public PrefixException(String message)

{

super(message + " Prefix code is invalid ");

}

}

/\*\*\*\*

\* Name: Manik Singh

\* Student Number: T00715263

\* Assignment Number: 4

\* Due Date: October 24, 2023

\* Program Description: This file contains the AreaException class which is a

\* child class of the Exception class. It will throw a custom exception if the

\* input has provided invalid area code.

\*\*\*\*/

public class AreaException extends Exception

{

public AreaException()

{

super(" Area code is invalid ");

}

public AreaException(String message)

{

super(message + " Area code is invalid ");

}

}

**Input File “numbers.txt”**

236-214-6733

250-001-2673

538-748-3779

5t6-767-6383

672-220-3454

778-yty-7738

672- 252-7733

236 -252-2324

23-6363-3720

250-2200-748

672-252-477

250-252-3838

6378-368-3839

672-7478-4773

23 6-252-2324

45-7889-7765

6377938762

672-6672782

5627892-253

**Valid File “valid.txt”**

Testing this Number: 236-214-6733

Code is valid

Testing this Number: 672-220-3454

Code is valid

Testing this Number: 672- 252-7733

Code is valid

Testing this Number: 236 -252-2324

Code is valid

Testing this Number: 250-252-3838

Code is valid

**Invalid File “invalid.txt”**

Testing this Number: 250-001-2673

Prefix error: 001 Prefix code is invalid

Testing this Number: 538-748-3779

Area code error: 538 Area code is invalid

Testing this Number: 5t6-767-6383

Number format error. Number provided 5t6-767-6383 contains letters. Please add numbers only

Testing this Number: 778-yty-7738

Number format error. Number provided 778-yty-7738 contains letters. Please add numbers only

Testing this Number: 23-6363-3720

Area code error: 23 Area code is invalid

Testing this Number: 250-2200-748

Prefix error: 2200 Prefix code is invalid

Testing this Number: 672-252-477

Line error: Line error: 477 Line code is invalid

Testing this Number: 6378-368-3839

Area code error: 6378 Area code is invalid

Testing this Number: 672-7478-4773

Prefix error: 7478 Prefix code is invalid

Testing this Number: 23 6-252-2324

Area code error: 23 6 Area code is invalid

Testing this Number: 45-7889-7765

Area code error: 45 Area code is invalid

Testing this Number: 6377938762

Hyphen error: 6377938762 has missing hyphen.

number format provided is incorrect. Hyphen/s is/are missing

Testing this Number: 672-6672782

Hyphen error: 672-6672782 has missing hyphen.

number format provided is incorrect. Hyphen/s is/are missing

Testing this Number: 5627892-253

Hyphen error: 5627892-253 has missing hyphen.

number format provided is incorrect. Hyphen/s is/are missing