import edu.duke.\*;

import java.util.\*;

public class GladLibMap {

private HashMap<String, ArrayList<String>> myMap;

private Random myRandom;

private ArrayList<String> used;

private ArrayList<String> usedcategory;

private static String dataSourceURL = "http://dukelearntoprogram.com/course3/data";

private static String dataSourceDirectory = "/Users/tianjiachen/Documents/Java/week 2/data";

public GladLibMap(){

myMap = new HashMap<String, ArrayList<String>>();

initializeFromSource(dataSourceDirectory);

myRandom = new Random();

}

public GladLibMap(String source){

initializeFromSource(source);

myRandom = new Random();

}

private void initializeFromSource(String source) {

ArrayList<String> arraylist = new ArrayList<String>();

String[] category = {"country", "color", "noun", "name", "adjective", "animal", "timeframe", "verb", "fruit"};

for (int k = 0; k < category.length; k++) {

arraylist = readIt(source+"/"+category[k]+".txt");

myMap.put(category[k], arraylist);

}

used = new ArrayList<String>();

usedcategory = new ArrayList<String>();

}

private String randomFrom(ArrayList<String> source){

int index = myRandom.nextInt(source.size());

return source.get(index);

}

private String getSubstitute(String label) {

if (myMap.containsKey(label)) {

if (!usedcategory.contains(label)) usedcategory.add(label);

return randomFrom(myMap.get(label));

}

else if (label.equals("number")) return ""+myRandom.nextInt(50)+5;

else return "\*\*UNKNOWN\*\*";

}

private String processWord(String w){

int first = w.indexOf("<");

int last = w.indexOf(">",first);

if (first == -1 || last == -1){

return w;

}

String prefix = w.substring(0,first);

String suffix = w.substring(last+1);

String sub = getSubstitute(w.substring(first+1,last));

int index = used.indexOf(sub);

int usedornot = 1;

while (usedornot == 1) {

if (index == -1) {

used.add(sub);

usedornot = 0;

}

else {

sub = getSubstitute(w.substring(first+1,last));

index = used.indexOf(sub);

}

}

return prefix+sub+suffix;

}

private void printOut(String s, int lineWidth){

int charsWritten = 0;

for(String w : s.split("\\s+")){

if (charsWritten + w.length() > lineWidth){

System.out.println();

charsWritten = 0;

}

System.out.print(w+" ");

charsWritten += w.length() + 1;

}

}

private String fromTemplate(String source){

String story = "";

if (source.startsWith("http")) {

URLResource resource = new URLResource(source);

for(String word : resource.words()){

story = story + processWord(word) + " ";

}

}

else {

FileResource resource = new FileResource(source);

for(String word : resource.words()){

story = story + processWord(word) + " ";

}

}

return story;

}

private ArrayList<String> readIt(String source){

ArrayList<String> list = new ArrayList<String>();

if (source.startsWith("http")) {

URLResource resource = new URLResource(source);

for(String line : resource.lines()){

list.add(line);

}

}

else {

FileResource resource = new FileResource(source);

for(String line : resource.lines()){

list.add(line);

}

}

return list;

}

private int totalWordsInMap() {

int sum = 0;

for (String word: myMap.keySet()) {

sum += myMap.get(word).size();

}

return sum;

}

private int totalWordsConsidered() {

int sum = 0;

for (int k = 0; k < usedcategory.size(); k++) {

sum += myMap.get(usedcategory.get(k)).size();

}

return sum;

}

public void makeStory(){

System.out.println("\n");

String story = fromTemplate("data/madtemplate2.txt");

printOut(story, 60);

int number = totalWordsInMap();

System.out.println("\t");

System.out.println("There are "+number+" words to pick from."+"\t");

number = totalWordsConsidered();

System.out.println("There are "+number+" words considered.");

}

}