import edu.duke.\*;

public class CommonWords

{

public String[] getCommon(){

FileResource resource = new FileResource("data/common.txt");

String[] common = new String[20];

int index = 0;

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

common[index] = s;

index += 1;

}

return common;

}

public int indexOf(String[] list, String word) {

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

if (list[k].equals(word)) {

return k;

}

}

return -1;

}

public void countWords(FileResource resource, String[] common, int[] counts){

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

word = word.toLowerCase();

int index = indexOf(common,word);

if (index != -1) {

counts[index] += 1;

}

}

}

void countShakespeare(){

//String[] plays = {"caesar.txt", "errors.txt", "hamlet.txt",

// "likeit.txt", "macbeth.txt", "romeo.txt"};

String[] plays = {"small.txt"};

String[] common = getCommon();

int[] counts = new int[common.length];

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

FileResource resource = new FileResource("data/" + plays[k]);

countWords(resource,common,counts);

System.out.println("done with " + plays[k]);

}

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

System.out.println(common[k] + "\t" + counts[k]);

}

}

}

import java.util.\*;

import edu.duke.\*;

public class CountWords {

StorageResource myWords;

public CountWords() {

myWords = new StorageResource();

}

public int getCount(){

return myWords.size();

}

public String getRandomWord(){

Random rand = new Random();

int choice = rand.nextInt(myWords.size());

for(String s : myWords.data()){

if (choice == 0) {

return s;

}

choice = choice - 1;

}

return "\*\*\* NEVER HAPPENS \*\*\*";

}

public void readWords(String source){

myWords.clear();

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

URLResource resource = new URLResource(source);

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

myWords.add(word.toLowerCase());

}

}

else {

FileResource resource = new FileResource(source);

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

myWords.add(word.toLowerCase());

}

}

}

public boolean contains(String[] list, int size, String word){

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

if (list[k].equals(word)){

return true;

}

}

return false;

}

public int countDifferentArray(){

int diffCount = 0;

String[] words = new String[getCount()];

for(String s : myWords.data()){

if (! contains(words,diffCount,s)){

words[diffCount] = s;

diffCount++;

}

}

return diffCount;

}

public int countDifferentArrayList(){

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

for(String s : myWords.data()){

if (! words.contains(s)) {

words.add(s);

}

}

return words.size();

}

public void tester(){

readWords("data/confucius.txt");

//readWords("http://dukelearntoprogram.com/data/confucius.txt");

System.out.println("number of words read: "+getCount());

int carray = countDifferentArray();

int clist = countDifferentArrayList();

System.out.println(carray+" "+clist);

}

public String getRandomWord(String[] words) {

Random rand = new Random();

int index = rand.nextInt(words.length);

return words[index];

}

public void randomTester(){

readWords("data/confucius.txt");

System.out.println("starting");

int RAND\_SIZE = 100000;

for(int k=0; k < RAND\_SIZE; k++){

String word = getRandomWord();

if (word.indexOf("\*\*\* NEVER") != -1){

System.out.println(word);

}

}

System.out.println("done with randoms");

String[] words = new String[myWords.size()];

int index = 0;

for(String s : myWords.data()){

words[index] = s;

index += 1;

}

System.out.println("starting array");

for(int k=0; k < RAND\_SIZE; k++){

String word = getRandomWord(words);

}System.out.println("done with randoms");

}

}

/\*\*

\* Simulate rolling two six-sided die, keep statistics

\*

\* @author Duke Software Team

\* @version 1.0

\*/

import java.util.Random;

public class DiceRolling

{

public void simulate(int rolls){

Random rand = new Random();

int [] counts = new int [13];

for(int k=0; k < rolls; k++){

int d1 = rand.nextInt(6) + 1;

int d2 = rand.nextInt(6) + 1;

System.out.println("roll is " + d1 + "+" + d2 + "=" + (d1+d2));

counts[d1+d2] += 1;

}

for (int k=2; k <=12; k++) {

System.out.println(k + "'s=\t" + counts[k] + "\t" + 100.0 \* counts[k]/rolls);

}

}

public void simpleSimulate(int rolls){

Random rand = new Random();

int twos = 0;

int twelves = 0;

for(int k=0; k < rolls; k++){

int d1 = rand.nextInt(6) + 1;

int d2 = rand.nextInt(6) + 1;

if (d1 + d2 == 2){

twos += 1;

}

else if (d1 + d2 == 12){

twelves += 1;

}

}

System.out.println("2's=\t" + twos + "\t" + 100.0 \* twos/rolls);

System.out.println("12's=\t"+twelves+"\t"+100.0\*twelves/rolls);

}

}