Great  
Excellent  
Awesome  
Demolish  
Blasphemous  
Average  
Defeat  
  
Defeat  
Average  
Demolish

package SetsMaps;  
  
import java.io.IOException;  
import java.nio.file.Files;  
import java.nio.file.Paths;  
import java.util.Set;  
import java.util.TreeSet;  
  
public class WordProcessor {  
  
 */\*\*  
 \* Reads words from the file and returns a set of unique words (case-insensitive).  
 \*/* public Set<String> getUniqueWords(String filePath) throws IOException {  
 Set<String> words = new TreeSet<>(String.*CASE\_INSENSITIVE\_ORDER*);  
  
 Files.*lines*(Paths.*get*(filePath))  
 .forEach(line -> {  
 String[] tokens = line.split("\\s+");  
 for (String word : tokens) {  
 if (!word.trim().isEmpty()) {  
 words.add(word.toLowerCase());  
 }  
 }  
 });  
  
 return words;  
 }  
  
  
 */\*\*  
 \* Displays words in ascending alphabetical order.  
 \*/* public void displayAscending(Set<String> words) {  
 System.*out*.println("Words in Ascending Order:");  
 words.forEach(System.*out*::println);  
 }  
  
 */\*\*  
 \* Displays words in descending alphabetical order.  
 \*/* public void displayDescending(Set<String> words) {  
 System.*out*.println("Words in Descending Order:");  
 words.stream()  
 .sorted((a, b) -> b.compareTo(a))  
 .forEach(System.*out*::println);  
 }  
  
 public static void main(String[] args) {  
 WordProcessor wp = new WordProcessor();  
 String filePath = "collection\_of\_words.txt"; // File should be in the same folder as this file  
  
 try {  
 Set<String> words = wp.getUniqueWords(filePath);  
 wp.displayAscending(words);  
 System.*out*.println();  
 wp.displayDescending(words);  
 } catch (IOException e) {  
 System.*out*.println("Error: " + e.getMessage());  
 }  
 }  
}