# EXP6 I/O Operation

**1 Experiment purpose**

Through this training, to master binary I/O operation and text I/O operation.

**2 Development Environment**

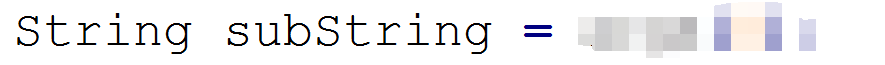
1. Notepad++
2. JDK 1.8+

**3 Content and Steps**

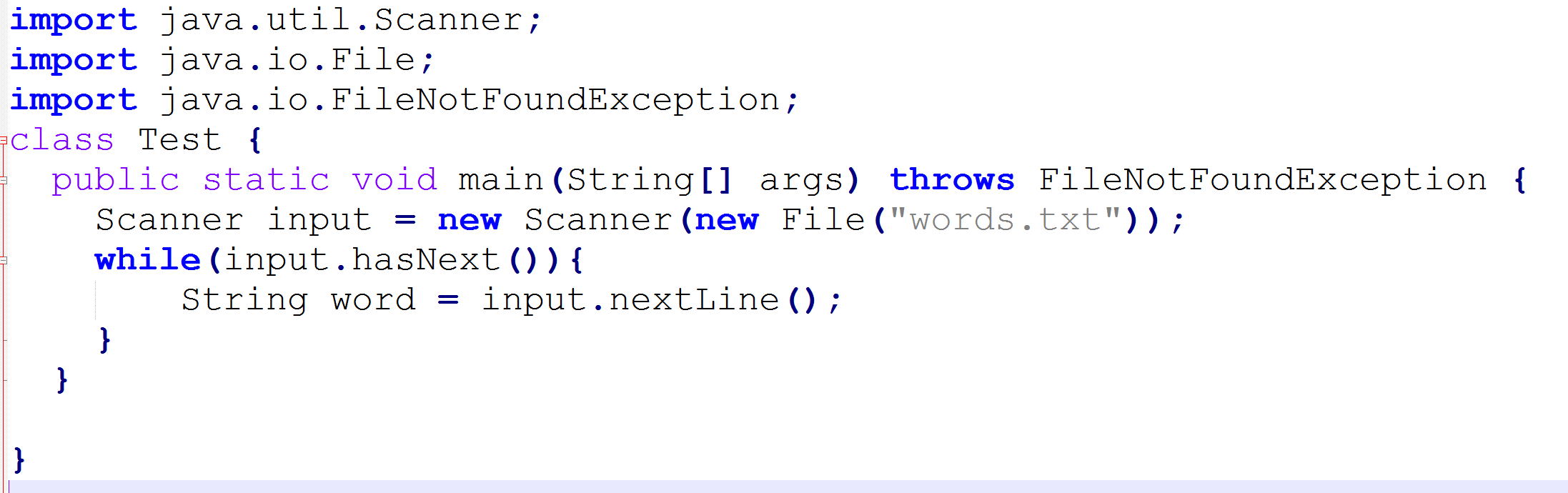
**Problem 1:** Extract all words which contain the sub-string “\*\*” from the text file word.txt, and store them into a new file \*\*\_word.txt.

**Steps:**

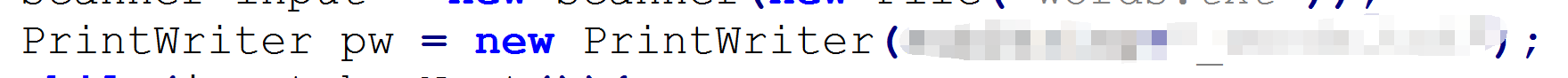
1. Get the sub-string from parameters



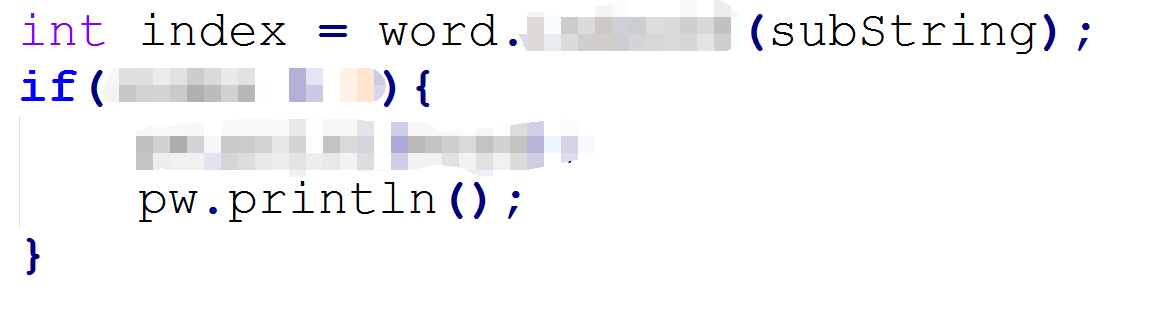
2. Read words from the text file



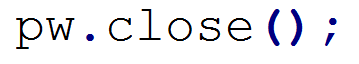
3. Create a PrintWriter Object for output.



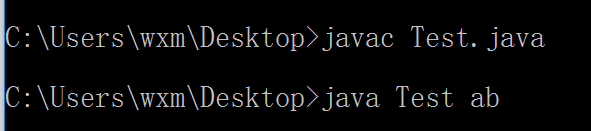
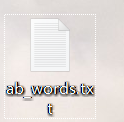
4. Judge and write the valid words to the file.

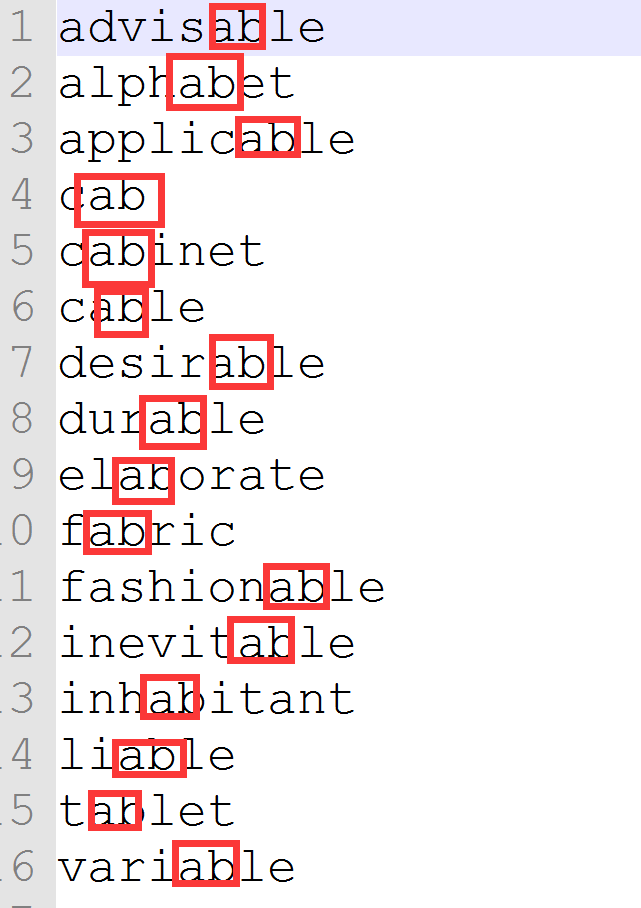


5. Close PrintWriter



6. Test the program:



**Problem 2:** Randomly generate circles and rectangles and sort by area.

1. Define the GeometricObject class which implements the Comparable interface.

**public** **abstract** **class** GeometricObject **implements** Comparable<GeometricObject>{

**public** **abstract** **double** getArea();

@Override

**public** **int** compareTo(GeometricObject g) {

**//comparison**

}

}

2. Define the classes Circle and Rectangle.

**public** **class** Circle **extends** GeometricObject {

**private** **double** r;

**public** Circle(**double** r){

//initialization

}

@Override

**public** **double** getArea() {

//calculate area

}

**public** String toString(){

**//return information;**

}

}

**public** **class** Rectangle **extends** GeometricObject {

**private** **double** w;

**private** **double** h;

**public** Rectangle(**double** w, **double** h){

**//**initialization

}

@Override

**public** **double** getArea() {

**//calculate area**

}

**public** String toString(){

**//return information;**

}

}

3. Generate N geometric objects randomly.



**public** **static** **void** generate(String path, **int** num){

**try** {

PrintWriter pw = **new** PrintWriter(path);

**// generate num geometric objects**

pw.close();

} **catch** (IOException e) {

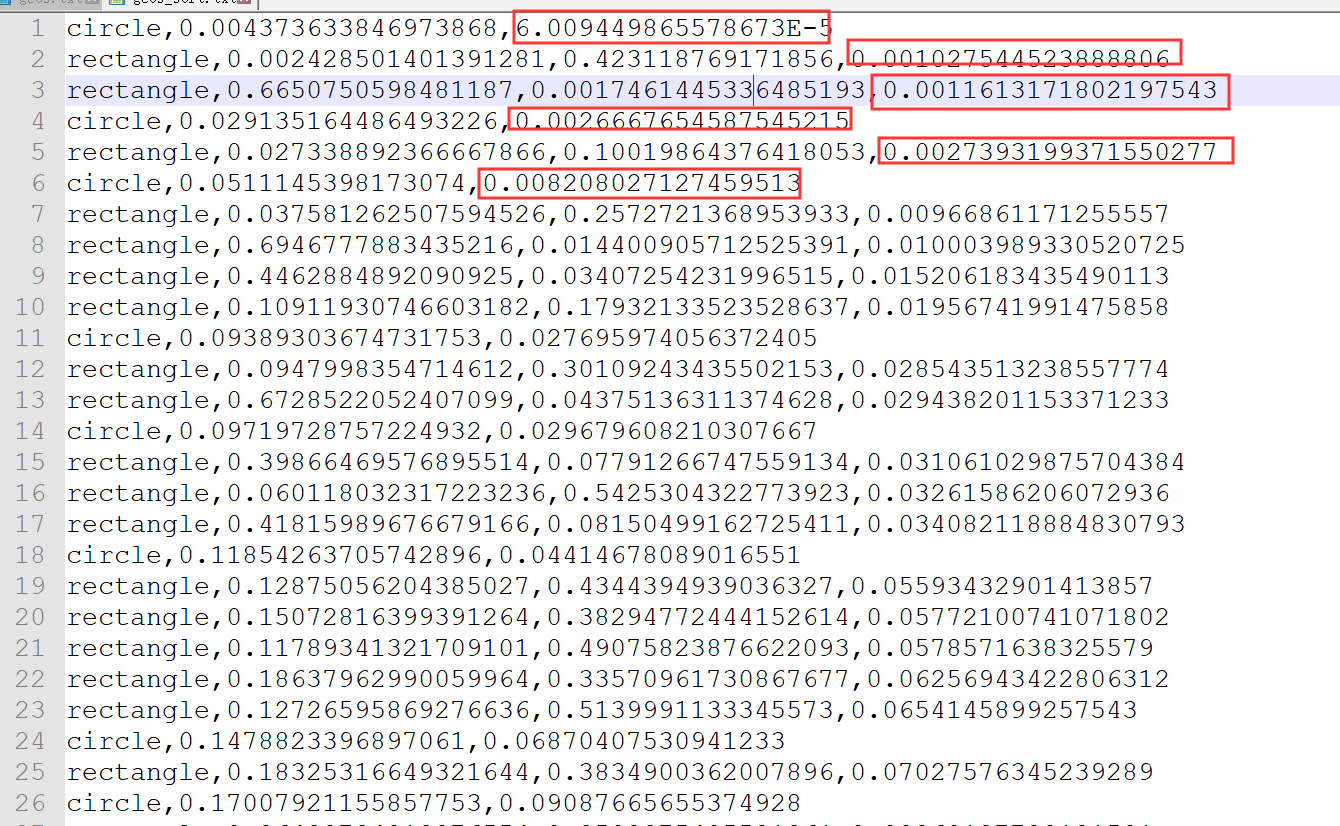
// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

4. Sort and export geometric objects to a file.



**public** **static** **void** sortByArea(String srcPath, String sortPath){

List<GeometricObject> list = **new** ArrayList<GeometricObject>();

**try** {

String line = "";

Scanner input = **new** Scanner(**new** FileInputStream(srcPath));

**// get geometric objects from file**

Collections.*sort*(list);

PrintWriter pw = **new** PrintWriter(sortPath);

// output sorted geometric object to file

pw.close();

input.close();

} **catch** (IOException e1) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

}