

Score: 3.500 (=100.0%)

Id: 46127

Potrebno je napraviti klasu `JavaProjectFileVisitor` koja nasljeđuje `SimpleFileVisitor<Path>`. Klasa mora imati navedene public metode:

```
public JavaProjectFileVisitor(Set<String> extensionFilter);
public Map<String,Integer> getNumFilesPerExtension();
```

Kroz konstruktor klasa dobiva skup koji sadrži popis ekstenzija. Metoda `getNumFilesPerExtension` se izvodi isključivo za datoteke koje odgovaraju filteru ekstenzija (tj. njihove ekstenzije se nalaze u skupu). Ključevi u mapi su ekstenzije datoteka (npr. `.txt`, `.java`, `.html`, ...) dok su vrijednosti broj datoteka s tom ekstenzijom. Možete pretpostaviti da ne postoje dvije datoteke istog imena. Ispisati broj datoteka samo onih ekstenzija koje se pojavljuju u direktoriju (pogledati primjer).

Pokretanjem koda nad direktorijem koji je prikazan u nastavku očekujemo prikazane rezultate:

```
JavaProjectFileVisitor visitor = new JavaProjectFileVisitor(Set.of(".txt", ".java", ".pdf"));
File f = new File("folder1");
Files.walkFileTree(f.toPath(), visitor);
Map<String, Integer> map = visitor.getNumFilesPerExtension();
for(String key : map.keySet()){
    System.out.println("Extension: " + key + " - " + map.get(key) + " times.");
}
```

Ispis:

```
Extension: .java - 2 times.
Extension: .txt - 2 times.
```

Izgled direktorija:

```
+--folder1
|  +--a.txt (10 B)
|  +--folder2
|  |  +--b1.java (11 B)
|  |  +--b2.java (15 B)
|  |  +--b3.txt (100 B)
|  +--folder3
```

Napomena: Klasu `JavaProjectFileVisitor` napisati bez modifikatora vidljivosti i kopirati u prostor za rješenje bez navođenja paketa kojem pripada. Na vrhu možete dodavati *importe* iz standardnih Javinih biblioteka. Importe na klase iz predavanja, auditornih i pripreme obrisati prije predaje rješenja. Uočite da programski kod koji predajete **ne smije ništa ispisivati**, nego samo vraća tražene podatke.

Student's answer:

```
1 import java.io.IOException;
2 import java.nio.file.FileVisitResult;
3 import java.nio.file.Path;
4 import java.nio.file.SimpleFileVisitor;
5 import java.nio.file.attribute.BasicFileAttributes;
6 import java.util.HashMap;
7 import java.util.HashSet;
8 import java.util.Map;
9 import java.util.Set;
10
11 class JavaProjectFileVisitor extends SimpleFileVisitor<Path> {
12     Set<String> extensionFilter = new HashSet<>();
13     Map<String,Integer> map = new HashMap<>();
14
15
16     public JavaProjectFileVisitor(Set<String> extensionFilter) {
17         this.extensionFilter = extensionFilter;
18     }
19
20     public Map<String,Integer> getNumFilesPerExtension() {
21         Map<String,Integer> out = new HashMap<>();
22         for (String ext : map.keySet()) {
23             if(extensionFilter.contains(ext)) {
24                 out.put(ext, map.get(ext));
25             }
26         }
27         return out;
28     }
29
30     @Override
31     public FileVisitResult visitFile(Path file, BasicFileAttributes attrs) throws
32         String filename = file.getFileName().toString();
33         int index = filename.lastIndexOf(".");
34         if (index != -1) {
35             String ext = filename.substring(index);
36             int count = map.getOrDefault(ext, 0);
37             map.put(ext, count + 1);
38         }
39         return FileVisitResult.CONTINUE;
40     }
41 }
42
43
44
45
46
47
48
49
50
51
52
53
```

Correct answer:

```
1 This test does not have "show solutions" option enabled.
```

Hint: Correct. Well done!

Exam results obtained during submission evaluation:

#	stdin	stdout	expected	stderr	percentage	hint	mode	correct?
1	instanceof	OK	OK	EdgarLibrary v0.11.1	25.00	Correct. Well done!	check elements order : false, case sensitive : false, ignore whitespace : true	true
2	numFilesPerExtension	OK	OK	EdgarLibrary v0.11.1	60.00	Correct. Well done!	check elements order : false, case sensitive : false, ignore whitespace : true	true
3	numFilesPerExtension2	OK	OK	EdgarLibrary v0.11.1	50.00	Correct. Well done!	check elements order : false, case sensitive : false, ignore whitespace : true	true

Student's result

Correct result

Rerun student's code

Edgar: On-Line Exam Web Application © Developed with ❤️ @FER