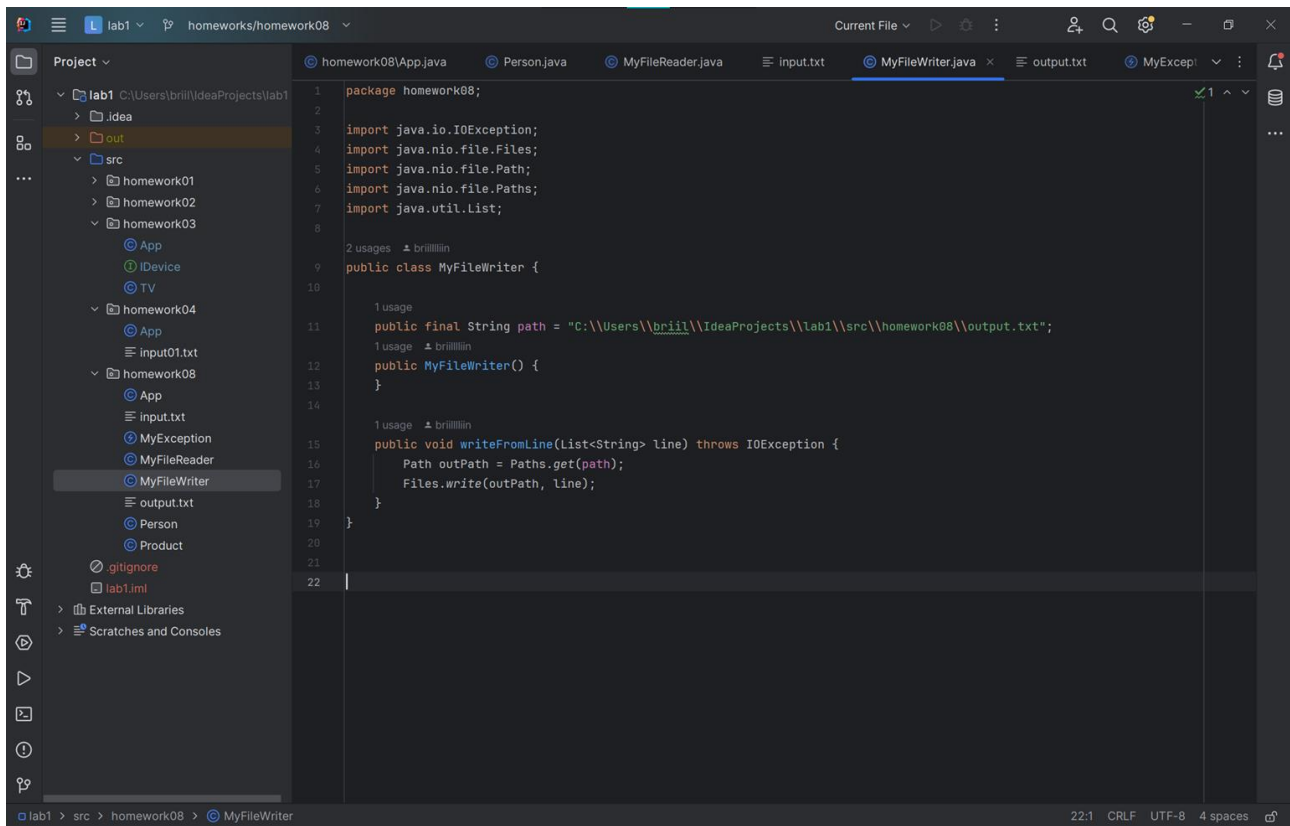


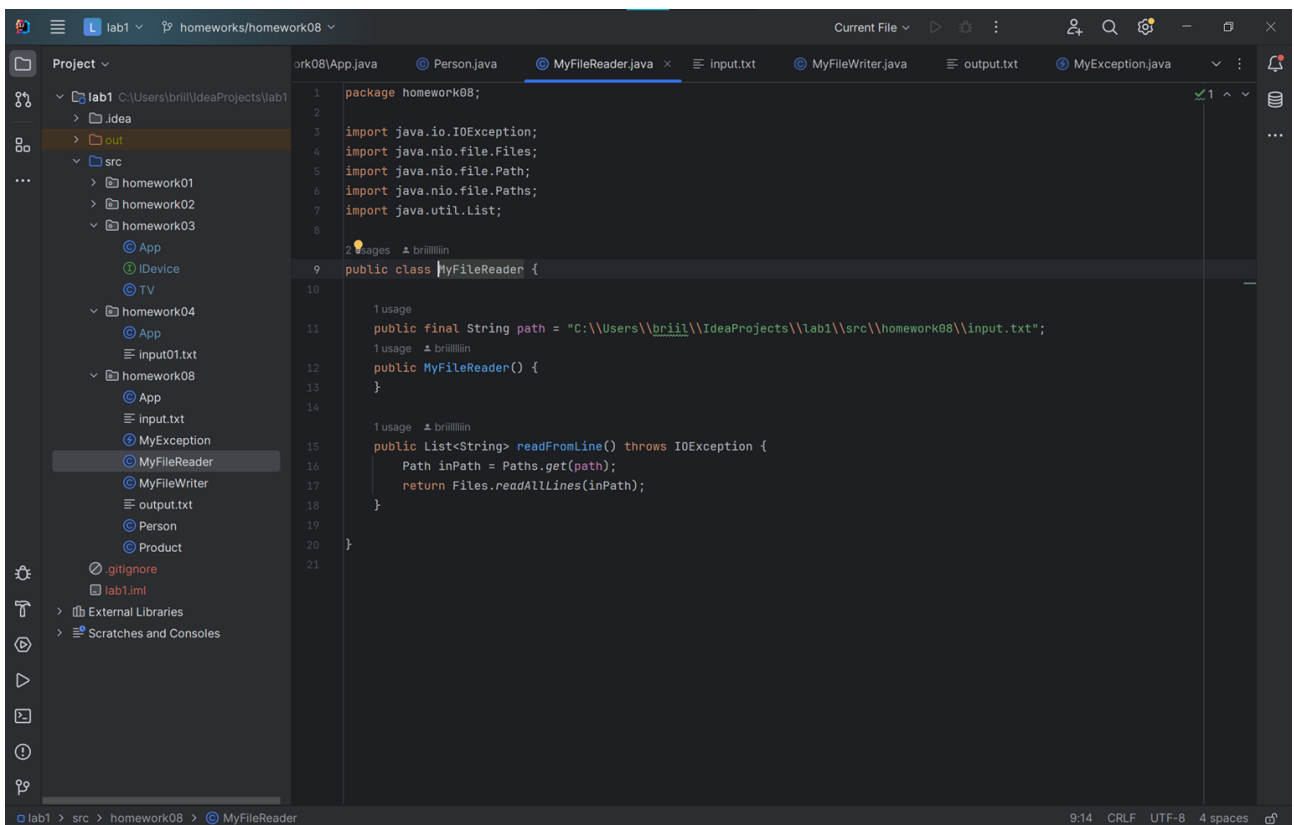
Классы MyFileReader и MyFileWriter, которые отвечают за чтение из файла и запись в файл



The screenshot shows the IntelliJ IDEA IDE with the 'MyFileWriter.java' file open. The left sidebar displays the project structure for 'lab1', with 'homework08' expanded to show 'MyFileWriter'. The main editor area contains the following Java code:

```
1 package homework08;
2
3 import java.io.IOException;
4 import java.nio.file.Files;
5 import java.nio.file.Path;
6 import java.nio.file.Paths;
7 import java.util.List;
8
9 public class MyFileWriter {
10
11     1 usage ± brilliIn
12     public final String path = "C:\\Users\\brii\\IdeaProjects\\lab1\\src\\homework08\\output.txt";
13     1 usage ± brilliIn
14     public MyFileWriter() {
15     }
16
17     1 usage ± brilliIn
18     public void writeFromLine(List<String> line) throws IOException {
19         Path outputPath = Paths.get(path);
20         Files.write(outputPath, line);
21     }
22 }
```

The status bar at the bottom indicates the file is at 'lab1 > src > homework08 > MyFileWriter' with a line count of 22, CRLF line endings, UTF-8 encoding, and 4 spaces per tab.



The screenshot shows the IntelliJ IDEA IDE with the 'MyFileReader.java' file open. The left sidebar displays the project structure for 'lab1', with 'homework08' expanded to show 'MyFileReader'. The main editor area contains the following Java code:

```
1 package homework08;
2
3 import java.io.IOException;
4 import java.nio.file.Files;
5 import java.nio.file.Path;
6 import java.nio.file.Paths;
7 import java.util.List;
8
9 public class MyFileReader {
10
11     1 usage ± brilliIn
12     public final String path = "C:\\Users\\brii\\IdeaProjects\\lab1\\src\\homework08\\input.txt";
13     1 usage ± brilliIn
14     public MyFileReader() {
15     }
16
17     1 usage ± brilliIn
18     public List<String> readFromLine() throws IOException {
19         Path inPath = Paths.get(path);
20         return Files.readAllLines(inPath);
21     }
22 }
```

The status bar at the bottom indicates the file is at 'lab1 > src > homework08 > MyFileReader' with a line count of 21, CRLF line endings, UTF-8 encoding, and 4 spaces per tab.

## Основные изменения в App классе

```
7  
8 public class App {  
9  
10  
11     public static void main(String[] args) throws MyException, IOException {  
12         List<Person> personList = new ArrayList<>();  
13         List<Product> productList = new ArrayList<>();  
14         List<String> output = new ArrayList<>();  
15  
16         MyFileReader in = new MyFileReader();  
17         MyFileWriter out = new MyFileWriter();  
18  
19         List<String> lines = in.readFromLine();  
20         int count = 0;  
21  
22         for (String l : lines) {  
23  
24             if (l.equals("")) {  
25                 count++;  
26             }  
27  
28             if (count == 0 && !l.equals("")) {  
29  
30                 String[] person = l.split( regex: " ");  
31  
32                 String name = person[0];  
33                 if (name.isEmpty()) {  
34                     throw new MyException("Имя не может быть пустым");  
35                 }  
36                 String total = person[1];  
37                 int sum = Integer.parseInt(total.trim());  
38                 if (sum < 0) {  
39                     throw new MyException("Деньги не могут быть отрицательными");  
40                 }  
41                 personList.add(Person.createPerson(name, sum));  
42             }  
43         }  
44     }  
45 }
```

```
60  
61  
62  
63         if (count == 2 && !l.equals("")) {  
64  
65             if (l.equals("END")) {  
66                 for (Person person : personList) {  
67                     output.addAll(person.getProducts());  
68                 }  
69                 break;  
70             }  
71  
72             String[] data = l.split( regex: ";" );  
73  
74             Person currPerson = Person.getPerson(data[0], personList);  
75             Product currProduct = Product.getProduct(data[1], productList);  
76  
77             if (currPerson != null && currProduct != null) {  
78                 output.add(currPerson.buy(currProduct));  
79             }  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89         out.writeFromLine(output);  
90     }  
91 }
```