

Smart.java

```
1 package prof;
2
3 import java.nio.charset.Charset;
10
11
12 public class Smart {
13
14
15     private ArrayList<Objekat> listaobjekata;
16
17     public Smart() {
18         this.listaobjekata = new ArrayList<Objekat>();
19     }
20
21     public ArrayList<Objekat> getListaobjekata() {
22         return listaobjekata;
23     }
24
25     public void setListaobjekata(ArrayList<Objekat> listaobjekata) {
26         this.listaobjekata = listaobjekata;
27     }
28
29     @Override
30     public String toString() {
31         return "Smart [listaobjekata=" + listaobjekata + "]";
32     }
33
34
35
36
37     public void lstprofesora() {
38         loadProf("listaprofesora.txt");
39         System.out.printf("%20s %25s, %2s, %20s \n", "Name Profesora",
"Current Location", "Br.", "Kako se profa oseca");
40         for (int i = 0; i < this.listaobjekata.size(); i++) {
41             Objekat objekat = this.listaobjekata.get(i);
42             System.out.println(objekat);
43         }
44
45     }
46
47
48
49
50     public boolean dodavanjeProfesora(Objekat prof) {
51
52         for (int i = 0; i < this.listaobjekata.size(); i++) {
53             Objekat profalista = this.listaobjekata.get(i);
54             if(profalista.getName().equalsIgnoreCase(prof.getName())) {
55                 return false;

```

Smart.java

```

56     }
57 }
58 this.listaobjekata.add(prof);
59 saveProf("listaprofesora.txt");
60 return true;
61 }
62
63
64 public Objekat izmenaProfesora(Objekat objekat) {
65     for (int i = 0; i < this.listaobjekata.size(); i++) {
66         Objekat objekat2 = this.listaobjekata.get(i);
67         String Name = objekat2.getName();
68         if (Name.equalsIgnoreCase(objekat.getName())) {
69             Objekat kojisemenja = this.listaobjekata.set(i, objekat);
70             saveProf("listaprofesora.txt");
71             return kojisemenja;
72         }
73     }
74     return null;
75 }
76
77
78
79 public void saveProf(String path) {
80
81     ArrayList<String>lines = new ArrayList<String>();
82     for(int i = 0; i< this.listaobjekata.size(); i++) {
83         Objekat objekat = this.listaobjekata.get(i);
84         String Name = objekat.getName();
85         String Location = objekat.getLokacija();
86         int number = objekat.getNumberlocation();
87         String prof0secaj = objekat.getProfa0secaj();
88
89         String line = Name + "/" + Location + "/" + number + "/" + prof0secaj;
90         lines.add(line);
91
92         try {
93             Files.write(Paths.get(path), lines,
94                 Charset.defaultCharset(), StandardOpenOption.CREATE,
95                 StandardOpenOption.TRUNCATE_EXISTING, StandardOpenOption.WRITE);
96         } catch (java.io.IOException e) {
97             System.out.println("Datoteka " + path + " nije
98                 pronadjena.");
99         }
100     }
101
102     public Objekat profBrisanjeIme(Objekat idX) {

```

```

103
104     int index = -1;
105     for (int i = 0; i < this.listaobjekata.size(); i++) {
106         if (this.listaobjekata.get(i).getName().equalsIgnoreCase
107             (idX.getName())) {
108             index = i;
109         }
110     }
111     if (index != -1) {
112         Objekat profesor = this.listaobjekata.remove(index);
113         System.out.println("Brisanje po imenu je uspesnu uradjeno");
114         saveProf("listaprofesora.txt");
115         return profesor;
116     }
117     return null;
118 }
119
120 public void loadProf(String path) {
121     this.listaobjekata = new ArrayList<Objekat>();
122     List<String> lines;
123     try {
124         lines = Files.readAllLines(Paths.get(path),
125             Charset.defaultCharset());
126         for (String line: lines) {
127             String[] attributes = line.split("/");
128             String Name = attributes[0];
129             String Location = attributes[1];
130             int number = Integer.parseInt(attributes[2]);
131             String profOsecaj = attributes[3];
132
133             Objekat objekat = new Objekat(Name, Location, number,
134                 profOsecaj);
135             this.listaobjekata.add(objekat);
136         }
137     } catch (java.io.IOException e) {
138         System.out.println("Datoteka " + path + " nije pronađena.");
139     }
140 }
141
142 }
143

```