

```

1 package Ejercicio06;
2
3 import java.io.BufferedReader;
4 import java.io.FileNotFoundException;
5 import java.io.FileReader;
6 import java.io.IOException;
7 import java.util.ArrayList;
8 import java.util.List;
9 import java.util.Scanner;
10
11 public class U08E06 {
12     //para puntuar los chistes que salen
13     private static int[] puntuacion = new int[3]; //para almacenar la puntuacion del usuario
14     private static int[] posChistes = new int[3]; //para almacenar las posiciones de los chistes del random
15     private static double[] comparacion; //para almacenar el manhattan de los usuarios
16     private static List<Usuario> misUsuarios = new ArrayList<>(); //para almacenar usuarios
17     private static List<Chiste> misChistes = new ArrayList<>(); //para almacenar chistes
18     private static List<Puntos> misPuntos = new ArrayList<>(); //para almacenar puntuaciones
19
20     public static void main(String[] args) {
21
22         //creamos los usuarios
23
24         try {
25             //nos creamos un filereader envuelto en un buffer de lectura
26             BufferedReader br = new BufferedReader(new FileReader("c:\\temp\\usuarios.csv"));
27
28             //leemos la primera linea
29             String linea = br.readLine();
30
31             //mientras hayan lineas que leer
32             while (linea != null) {
33
34                 //construimos un array con los campos de la linea. Separamos los campos con split
35                 String [] fields = linea.split(";");
36
37                 //si hay info de la provincia, la mostramos por pantalla
38                 if(fields.length == 2) {
39                     int id = Integer.parseInt(fields[0]);
40                     String usuari = fields[1];
41                     Usuario temp = new Usuario(id, usuari);
42                     misUsuarios.add(temp);
43                 }
44
45                 //leemos la siguiente linea
46                 linea = br.readLine();
47             }
48
49             br.close();
50
51

```

```

52
53     } catch (FileNotFoundException e) {
54         System.out.println(e.getMessage());
55     } catch (IOException e) {
56         System.out.println(e.getMessage());
57     }
58
59     //creamos los chistes
60
61
62     try {
63         //nos creamos un filereader envuelto en un buffer de lectura
64         BufferedReader br = new BufferedReader(new FileReader("c:\\temp\\chistes.csv"));
65
66         //leemos la primera linea
67         String linea = br.readLine();
68
69         //mientras hayan lineas que leer
70         while (linea != null) {
71
72             //construimos un array con los campos de la linea. Separamos los campos con split
73             String [] fields = linea.split(";");
74
75             //si hay info de la provincia, la mostramos por pantalla
76             if(fields.length == 2) {
77                 int id = Integer.parseInt(fields[0]);
78                 String text = fields[1];
79                 Chiste temp = new Chiste(id, text);
80                 misChistes.add(temp);
81             }
82
83             //leemos la siguiente linea
84             linea = br.readLine();
85         }
86
87         br.close();
88
89
90
91     } catch (FileNotFoundException e) {
92         System.out.println(e.getMessage());
93     } catch (IOException e) {
94         System.out.println(e.getMessage());
95     }
96
97     //creamos los puntos
98
99
100    try {
101        //nos creamos un filereader envuelto en un buffer de lectura
102        BufferedReader br = new BufferedReader(new FileReader("c:\\temp\\puntos.csv"));

```

```

103
104 //leemos la primera linea
105 String linea = br.readLine();
106
107 //mientras hayan lineas que leer
108 while (linea != null) {
109
110     //construimos un array con los campos de la linea. Separamos los campos con split
111     String [] fields = linea.split(";");
112
113     //si hay info de la provincia, la mostramos por pantalla
114     if(fields.length == 3) {
115         int idu = Integer.parseInt(fields[0]);
116         int idc = Integer.parseInt(fields[1]);
117         int punts = Integer.parseInt(fields[2]);
118         Puntos temp = new Puntos(idu, idc, punts);
119         misPuntos.add(temp);
120     }
121
122     //leemos la siguiente linea
123     linea = br.readLine();
124 }
125
126 br.close();
127
128
129
130 } catch (FileNotFoundException e) {
131     System.out.println(e.getMessage());
132 } catch (IOException e) {
133     System.out.println(e.getMessage());
134 }
135
136
137
138 Scanner stdin = new Scanner(System.in);
139
140 //Escogemos 3 chistes al azar y los puntuamos
141 for(int i = 0; i <= 2; i++){
142     puntuacion[i] = 0;
143     int pos = (int)(Math.random() * misChistes.size());
144     posChistes[i] = pos;
145     System.out.println(misChistes.get(pos).getTexto());
146     while (puntuacion[i] <= 0 || puntuacion[i] >= 6){
147         System.out.print("Puntuacion del chiste[1-5]: ");
148         puntuacion[i] = stdin.nextInt();
149     }
150 }
151
152 //algoritmo de comparacion con el resto de usuarios
153

```

```

//inicializamos la nota
//contamos un chiste al azar
//almacenamos la posicion del chiste
//mostramos el chiste

//almacenamos la puntuacion

```

```

154 //creamos el array donde iremos almacenando los distintos calculos
155 comparacion = new double[misUsuarios.size()];
156
157 //calculamos el manhattam vs los usuarios
158 for(int i = 0; i < comparacion.length; i++){
159     comparacion[i] = manhattam(i);
160 }
161
162 /*//para ver los calculos
163 for(int i = 0; i < comparacion.length; i++){
164     System.out.println(comparacion[i]);
165 }*/
166
167 //obtenemos la menor distancia
168 double menor=comparacion[0];
169 for(int i = 0; i < comparacion.length; i++){
170     if (comparacion[i] < menor) menor = comparacion[i];
171 }
172
173 //System.out.println(menor);
174
175 //System.out.println();
176
177 System.out.println("Usuarios parecidos: ");
178
179 //imprimimos usuarios parecidos
180 for(int i = 0; i < comparacion.length; i++){
181     if (comparacion[i] == menor){
182         System.out.println(misUsuarios.get(i).getNombre());
183         otrosChistes(i);
184     }
185 }
186
187
188
189 }
190
191 public static double manhattam(int usuario){
192     //aumentamos 1 el usuario para que coincida la posicion
193     usuario++;
194
195     double nota1=0;
196     double nota2=0;
197     double nota3=0;
198
199     //corremos el arrayList de puntos para el chiste 1.
200     for(int i = 0; i < misPuntos.size();i++){
201         if(misPuntos.get(i).getIdUsuario() == usuario && misPuntos.get(i).getIdChiste() == posChistes[0]) nota1 =
            misPuntos.get(i).getPuntos();
202     }
203

```

```

204     //corremos el arrayList de puntos para el chiste 2
205     for(int i = 0; i < misPuntos.size();i++){
206         if(misPuntos.get(i).getIdUsuario() == usuario && misPuntos.get(i).getIdChiste() == posChistes[1]) nota2 =
            misPuntos.get(i).getPuntos();
207     }
208
209     //corremos el arrayList de puntos para el chiste 3
210     for(int i = 0; i < misPuntos.size();i++){
211         if(misPuntos.get(i).getIdUsuario() == usuario && misPuntos.get(i).getIdChiste() == posChistes[2]) nota3 =
            misPuntos.get(i).getPuntos();
212     }
213
214     double calculo = (Math.abs(nota1-puntuacion[0] + Math.abs(nota2-puntuacion[1]) + Math.abs(nota3-puntuacion[2]))/3);
215
216
217
218
219     return calculo;
220 }
221
222
223 public static void otrosChistes(int usu){
224
225     usu++;
226
227     for(int i = 0; i < misPuntos.size(); i++){
228         if (misPuntos.get(i).getIdUsuario() == usu){
229             int chiste = misPuntos.get(i).getIdChiste();
230             for(int z = 0; z < misChistes.size();z++){
231                 if (misChistes.get(z).getId() == chiste && posChistes[0] != z && posChistes[1] != z && posChistes[2] != z){
232                     System.out.println (misChistes.get(z).getTexto());
233                     System.out.println("Puntos otorgados: " + misPuntos.get(i).getPuntos());
234                 }
235             }
236         }
237     }
238 }
239
240
241
242
243 } //fin clase U08E06
244
245 class Usuario{
246
247     //variables miembro
248     int Id;
249     String nombre;
250
251     //constructor
252     public Usuario(int id, String nom){

```

```

253         this.Id = id;
254         this.nombre = nom;
255     }
256
257     //getters
258     public int getId() { return this.Id;}
259     public String getNombre() { return this.nombre;}
260
261     //setters
262     public void setId(int id) { this.Id = id;}
263     public void setNombre(String nom) { this.nombre = nom;}
264
265 } //fin clase Usuario
266
267 class Chiste{
268
269     //variables miembro
270     int Id;
271     String texto;
272
273     public Chiste (int id, String chiste){
274         this.Id = id;
275         this.texto = chiste;
276     }
277
278     //getters
279     public int getId() { return this.Id;}
280     public String getTexto() { return this.texto;}
281
282     //setters
283     public void setId(int id) { this.Id = id;}
284     public void setNombre(String chiste) { this.texto = chiste;}
285
286 }//fin clase chiste
287
288 class Puntos{
289
290     //variables miembro
291     int IdUsuario;
292     int IdChiste;
293     int puntos;
294
295     public Puntos(int idUser, int idText, int punts){
296         this.IdUsuario = idUser;
297         this.IdChiste = idText;
298         this.puntos = punts;
299     }
300
301     //getters
302     public int getIdUsuario() { return this.IdUsuario;}
303     public int getIdChiste() { return this.IdChiste;}

```

```
304     public int getPuntos() { return this.puntos;}
305
306     //setters
307     public void setIdUsuario(int IdUser) { this.IdUsuario = IdUser;}
308     public void setIdChiste(int idtext) { this.IdChiste = idtext;}
309     public void setPuntos(int punts) { this.puntos = punts;}
310
311 }//fin clase Puntos
312
313
314
315
```