



P3-part2: Introducció a la programació - ArrayList i Col·leccions

ARRAYLIST

Exercici 1

L'ajuntament del teu poble organitza un concert i t'ha encarregat portar el registre de les persones que han comprat entrada. Necessites un programa que permeti guardar els noms de les persones que han comprat l'entrada, i després mostri per pantalla el nombre total d'entrades venudes i el nom de cada assistent.

```
1 package part_02;
2
3 import java.util.ArrayList;
4
5 public class Exercici_01_Arraylist {
6
7
8     public static void main(String[] args) {
9         ArrayList<String> assistants = new ArrayList<>();
10        Scanner scanner = new Scanner(System.in);
11        afegirAssistents(assistents, scanner);
12        mostraAssistents(assistents);
13    }
14
15    public static void afegirAssistents(ArrayList<String> assistants, Scanner scanner) {
16        String resposta;
17        do {
18            System.out.print("Afegeix el següent assistent:");
19            String assistant = scanner.nextLine();
20            assistants.add(assistant);
21            System.out.print("Vols afegir més assistents? s/n");
22            resposta = scanner.nextLine();
23        } while (resposta.equals("s"));
24    }
25
26    public static void mostraAssistents(ArrayList<String> assistants) {
27        System.out.println(assistents.size());
28        System.out.println(assistents);
29    }
30
31 }
32
33
34
35
```

Console:

```
Executed: Exercici_01_Arraylist [Java Application] C:\Users\Bemen3\Desktop\Eclipse\plugins\org.eclipse.jdt.ui\org.eclipse.jdt.ui.hotspot.jre.full.win32.x86_64.21.0.5\20241023-1957\jre\bin\java.exe (24 Feb 2025, 8:48:54) [pid: 11096]
Afegeix el següent assistent:nidaa
Vols afegir més assistents?: s/ns
Afegeix el següent assistent:edim
Vols afegir més assistents?: s/ns
Afegeix el següent assistent:caravaca
Vols afegir més assistents?: s/ns
Afegeix el següent assistent:pedro
Vols afegir més assistents?: s/ns
Afegeix el següent assistent:marco
Vols afegir més assistents?: s/n
5
[nidaa, edim, caravaca, pedro, marco]
```

En el mètode main declarem un ArrayList anomenada assistants i dos mètodes, un per afegir assistents i un altre per mostrar-los.

```
public static void main(String[] args) {
    ArrayList<String> assistants = new ArrayList<>();
    Scanner scanner = new Scanner(System.in);
    afegirAssistents(assistents, scanner);
    mostraAssistents(assistents);
}
```

En el mètode per afegir assistents cridem l'Arraylist i l'objecte d'escaner. Declarem una variable de tipus String anomenada resposta.

```
public static void afegirAssistents(ArrayList<String> assistents, Scanner scanner) {  
    String resposta;
```

Amb un bucle do-while imprimim el missatge per afegir el següent assistent i el guardem en una nova variable String anomenada assistent.

```
do {  
    System.out.print("Afegeix el següent assistent:");  
    String assistent = scanner.nextLine();  
    assistents.add(assistent);
```

Preguntem si volem afegir més assistents per seguir executant el bucle o sortir-ne. Mentre responguem que si se seguirà executant el bucle.

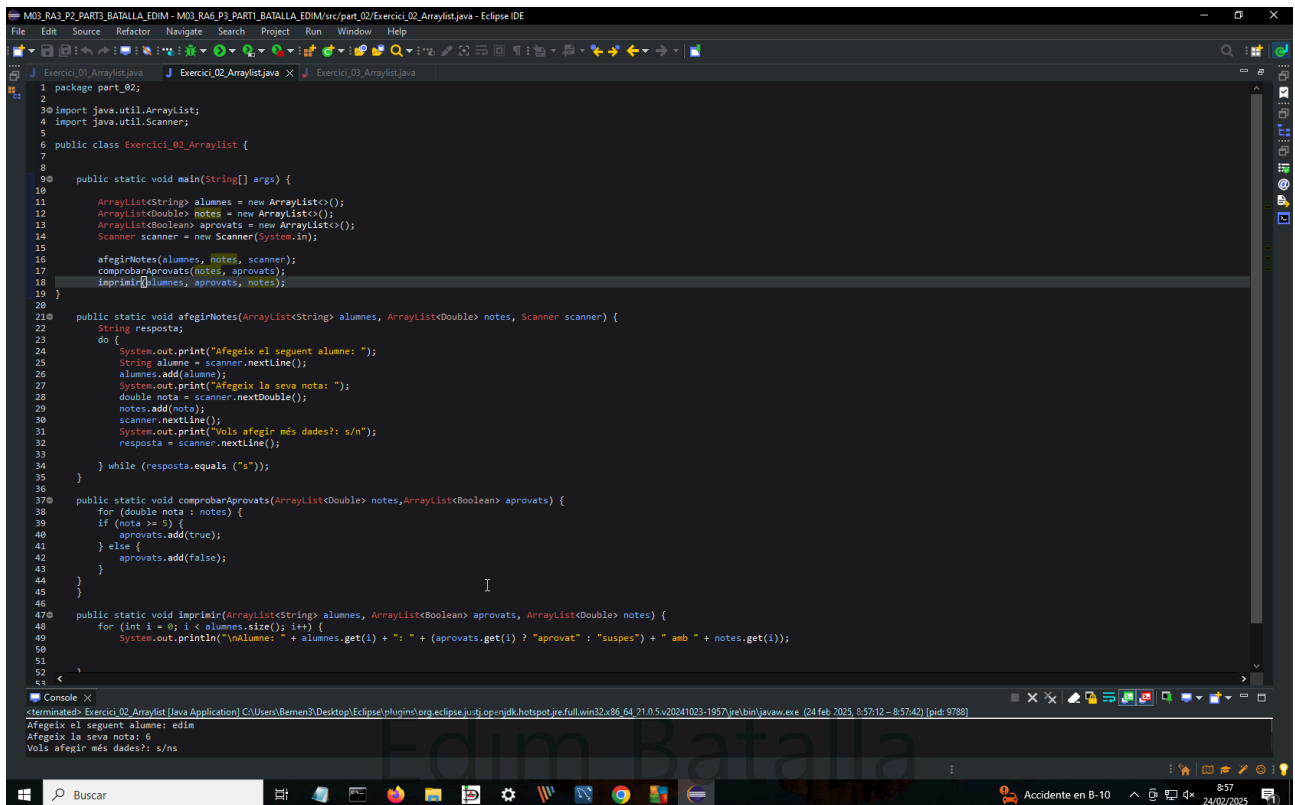
```
System.out.print("Vols afegir més assistents?: s/n");  
resposta = scanner.nextLine();  
} while (resposta.equals ("s"));  
}
```

En el mètode per mostrar els assistents cridem l'Arraylist. Amb el mètode size() imprimim el nombre de assistent juntament amb el contingut de l'Arraylist assistents.

```
public static void mostraAssistents(ArrayList<String> assistents) {  
    System.out.println(assistents.size());  
    System.out.println(assistents);  
}  
}
```

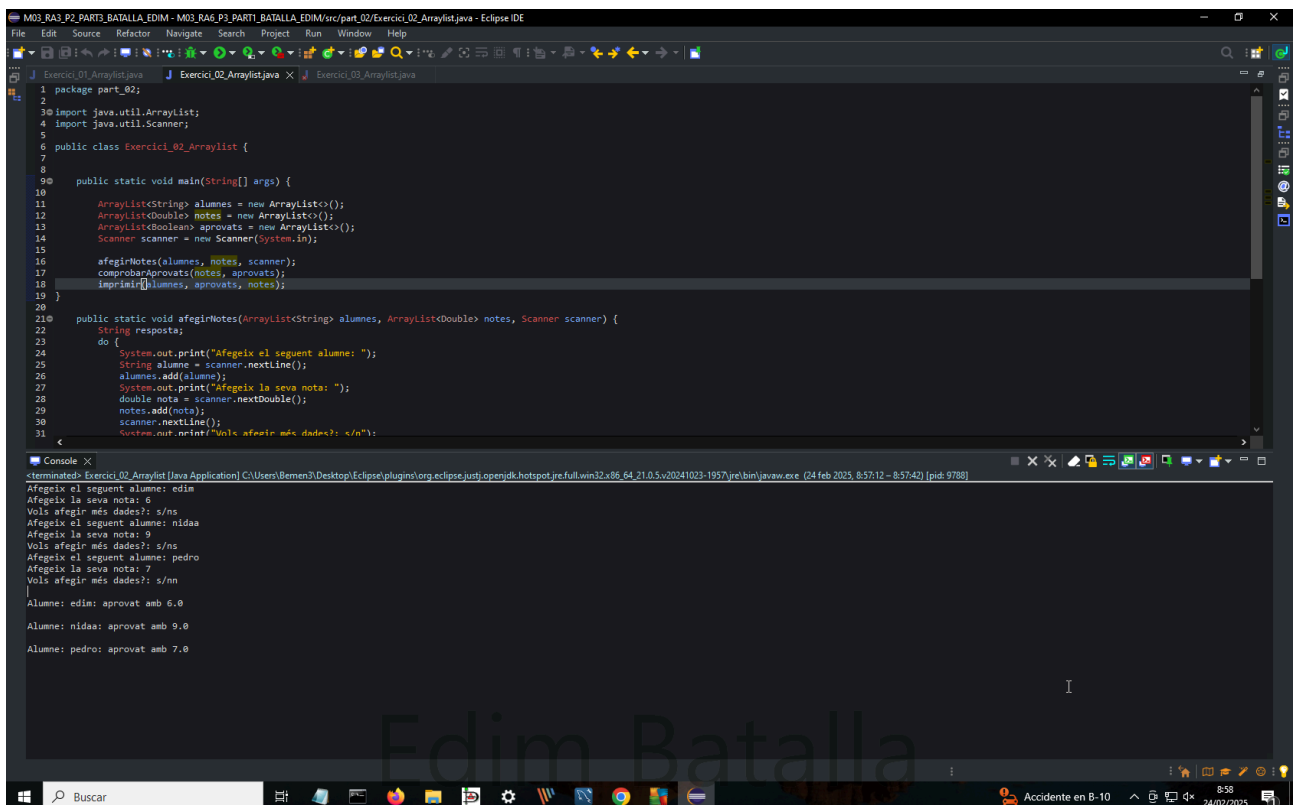
Exercici 2

Ets professor d'informàtica i acabes de corregir un examen. Vols fer un programa que guardi les notes dels alumnes. Després, el programa ha de mostrar totes les notes en ordre i dir quants alumnes han aprovat (nota 5 o més) i quants han suspès.



```
1 package part_02;
2
3 import java.util.ArrayList;
4 import java.util.Scanner;
5
6 public class Exercici_02_Arraylist {
7
8
9
10 public static void main(String[] args) {
11     ArrayList<String> alumnes = new ArrayList<>();
12     ArrayList<Double> notes = new ArrayList<>();
13     ArrayList<Boolean> aprovats = new ArrayList<>();
14     Scanner scanner = new Scanner(System.in);
15
16     afegirNotes(alumnes, notes, scanner);
17     comprovarAprovats(notes, aprovats);
18     imprimir(alumnes, aprovats, notes);
19 }
20
21 public static void afegirNotes(ArrayList<String> alumnes, ArrayList<Double> notes, Scanner scanner) {
22     String resposta;
23     do {
24         System.out.print("Afegeix el següent alumne: ");
25         String alumne = scanner.nextLine();
26         alumnes.add(alumne);
27         System.out.print("Afegeix la seva nota: ");
28         double nota = scanner.nextDouble();
29         notes.add(nota);
30         scanner.nextLine();
31         System.out.print("Vols afegir més dades?: s/n");
32         resposta = scanner.nextLine();
33     } while (resposta.equals("s"));
34 }
35
36 public static void comprovarAprovats(ArrayList<Double> notes, ArrayList<Boolean> aprovats) {
37     for (double nota : notes) {
38         if (nota >= 5) {
39             aprovats.add(true);
40         } else {
41             aprovats.add(false);
42         }
43     }
44 }
45
46 public static void imprimir(ArrayList<String> alumnes, ArrayList<Boolean> aprovats, ArrayList<Double> notes) {
47     for (int i = 0; i < alumnes.size(); i++) {
48         System.out.println("\tAlumne: " + alumnes.get(i) + " : " + (aprovats.get(i) ? "Aprovat" : "Suspes") + " amb " + notes.get(i));
49     }
50 }
51
52 }
53 }
```

Console X
terminated: Exercici_02_Arraylist [Java Application] C:\Users\Bemen3\Desktop\Eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.21.0.5.v20241023-1957\jre\bin\javaw.exe (24 Feb 2025, 8:57:12 - 8:57:42) [pid: 9788]
Afegeix el següent alumne: edim
Afegeix la seva nota: 6
Vols afegir més dades?: s/ns



```
1 package part_02;
2
3 import java.util.ArrayList;
4 import java.util.Scanner;
5
6 public class Exercici_02_Arraylist {
7
8
9
10 public static void main(String[] args) {
11     ArrayList<String> alumnes = new ArrayList<>();
12     ArrayList<Double> notes = new ArrayList<>();
13     ArrayList<Boolean> aprovats = new ArrayList<>();
14     Scanner scanner = new Scanner(System.in);
15
16     afegirNotes(alumnes, notes, scanner);
17     comprovarAprovats(notes, aprovats);
18     imprimir(alumnes, aprovats, notes);
19 }
20
21 public static void afegirNotes(ArrayList<String> alumnes, ArrayList<Double> notes, Scanner scanner) {
22     String resposta;
23     do {
24         System.out.print("Afegeix el següent alumne: ");
25         String alumne = scanner.nextLine();
26         alumnes.add(alumne);
27         System.out.print("Afegeix la seva nota: ");
28         double nota = scanner.nextDouble();
29         notes.add(nota);
30         scanner.nextLine();
31         System.out.print("Vols afegir més dades?: s/n");
32         resposta = scanner.nextLine();
33     } while (resposta.equals("s"));
34 }
35
36 public static void comprovarAprovats(ArrayList<Double> notes, ArrayList<Boolean> aprovats) {
37     for (double nota : notes) {
38         if (nota >= 5) {
39             aprovats.add(true);
40         } else {
41             aprovats.add(false);
42         }
43     }
44 }
45
46 public static void imprimir(ArrayList<String> alumnes, ArrayList<Boolean> aprovats, ArrayList<Double> notes) {
47     for (int i = 0; i < alumnes.size(); i++) {
48         System.out.println("\tAlumne: " + alumnes.get(i) + " : " + (aprovats.get(i) ? "Aprovat" : "Suspes") + " amb " + notes.get(i));
49     }
50 }
51
52 }
53 }
```

Console X
terminated: Exercici_02_Arraylist [Java Application] C:\Users\Bemen3\Desktop\Eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.21.0.5.v20241023-1957\jre\bin\javaw.exe (24 Feb 2025, 8:57:12 - 8:57:42) [pid: 9788]
Afegeix el següent alumne: edim
Afegeix la seva nota: 6
Vols afegir més dades?: s/ns
Afegeix el següent alumne: nidas
Afegeix la seva nota: 9
Vols afegir més dades?: s/ns
Afegeix el següent alumne: pedro
Afegeix la seva nota: 7
Vols afegir més dades?: s/nn
Alumne: edim: aprovat amb 6.0
Alumne: nidas: aprovat amb 9.0
Alumne: pedro: aprovat amb 7.0

En el mètode main declarem tres ArrayLists, una amb el nom dels alumnes (String) una amb les notes (double) i una per determinar si amb la nota està aprovat o no (Booleà)

```
public static void main(String[] args) {  
  
ArrayList<String> alumnes = new ArrayList<>();  
ArrayList<Double> notes = new ArrayList<>();  
ArrayList<Boolean> aprovats = new ArrayList<>();  
Scanner scanner = new Scanner(System.in);
```

També declarem els mètodes per afegir notes, comprovar si les notes arriben a l'aprovat o no i per imprimir les notes per pantalla.

```
afegirNotes(alumnes, notes, scanner);  
comprobarAprovats(notes, aprovats);  
imprimir(alumnes, aprovats, notes);  
}
```

En el mètode d'afegir notes cridem els ArrayList amb els noms dels alumnes i el de les notes juntament amb l'objecte del escàner.

```
public static void afegirNotes(ArrayList<String> alumnes, ArrayList<Double>  
notes, Scanner scanner) {  
String resposta;
```

En un bucle do-while demanem el nom de l'alumne i la nota corresponent. Un cop guardat tant el nom com la nota en els ArrayLists corresponents amb la funció .add preguntem a l'usuari si vol afegir més alumnes i mes notes. El bucle es recorre mentre l'usuari respongui afirmativament.

```
do {  
System.out.print("Afegeix el següent alumne: ");  
String alumne = scanner.nextLine();  
alumnes.add(alumne);  
System.out.print("Afegeix la seva nota: ");  
double nota = scanner.nextDouble();  
notes.add(nota);  
scanner.nextLine();  
System.out.print("Vols afegir més dades?: s/n");  
resposta = scanner.nextLine();  
} while (resposta.equals ("s"));  
}
```

En el segon mètode cridem els ArrayList de notes i el d'aprovats

```
public static void comprobarAprovats(ArrayList<Double>  
notes, ArrayList<Boolean> aprovats) {
```

Amb un bucle for agafem cada element i comprovem si la nota és major o igual a 5, aprovat ho declarem com a true. O menor, suspens o declarem com a false.

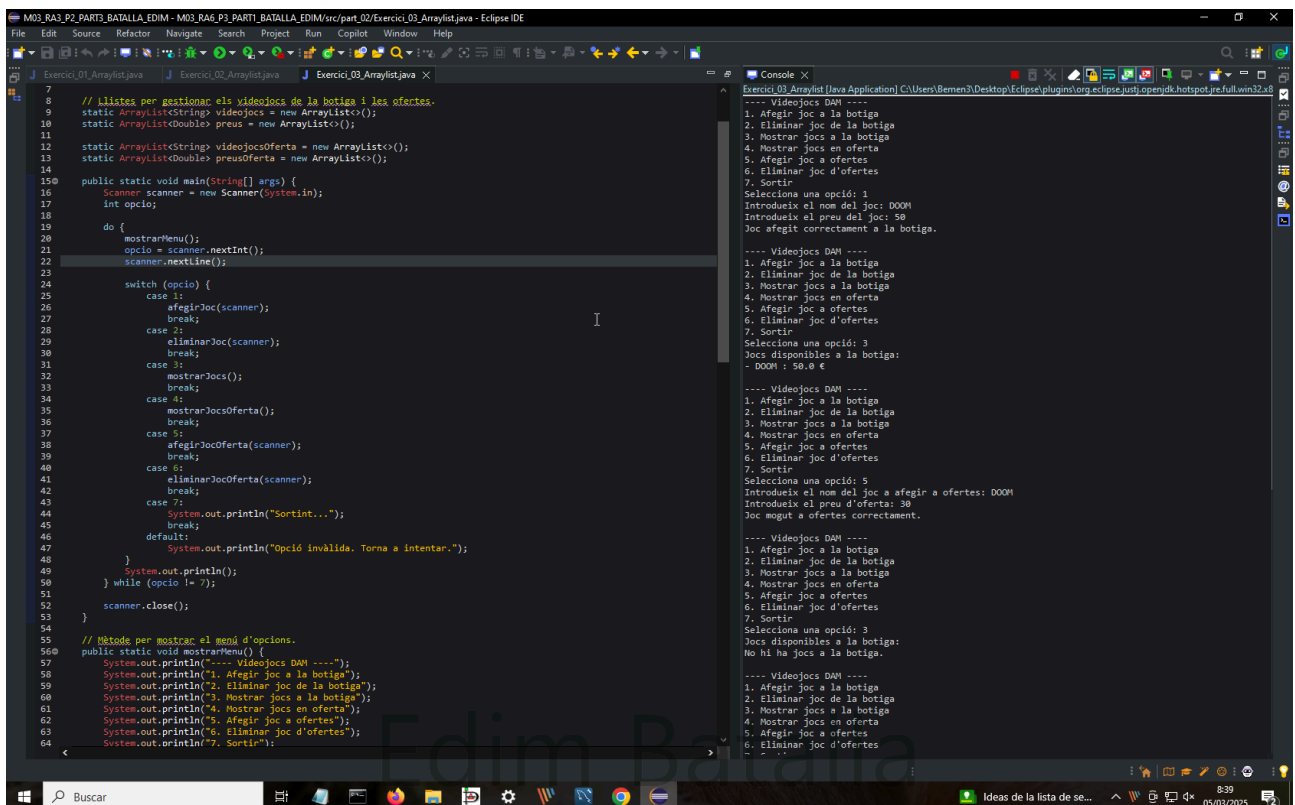
```
for (double nota : notes) {  
if (nota >= 5) {  
aprovats.add(true);  
} else {  
aprovats.add(false);  
}  
}  
}
```

En el mètode per imprimir cridem els tres ArrayList. Recorrem l'arraylist amb els noms dels alumnes i per cada alumne mostrem per pantalla si ha aprovat o no i amb quina nota.

```
public static void imprimir(ArrayList<String> alumnes, ArrayList<Boolean>
aprovats, ArrayList<Double> notes) {
for (int i = 0; i < alumnes.size(); i++) {
System.out.println("\nAlumne: " + alumnes.get(i) + ": " + (aprovats.get(i) ?
"aprovat" : "suspes") + " amb " + notes.get(i));
}
}
}
```

Exercici 3

Treballes en una botiga de videojocs i vols fer un programa per gestionar les noves ofertes. Cada vegada que un joc estigui d'oferta, el seu nom s'afegirà a la llista. El programa ha de permetre afegir títols, eliminar un joc de la llista quan acabi l'oferta i mostrar quins jocs estan actualment en promoció.



```
7 // Llistes per gestionar els videojocs de la botiga i les ofertes.
8 static ArrayList<String> videojocs = new ArrayList<>();
9 static ArrayList<Double> preus = new ArrayList<>();
10 static ArrayList<String> videojocsOfertes = new ArrayList<>();
11 static ArrayList<Double> preusOfertes = new ArrayList<>();
12
13 public static void main(String[] args) {
14     Scanner scanner = new Scanner(System.in);
15     int opcio;
16     do {
17         mostrarMenu();
18         opcio = scanner.nextInt();
19         scanner.nextLine();
20         switch (opcio) {
21             case 1:
22                 afegirJoc(scanner);
23                 break;
24             case 2:
25                 eliminarJoc(scanner);
26                 break;
27             case 3:
28                 mostrarJocs();
29                 break;
30             case 4:
31                 mostrarJocsOfertes();
32                 break;
33             case 5:
34                 afegirJocOferta(scanner);
35                 break;
36             case 6:
37                 eliminarJocOferta(scanner);
38                 break;
39             case 7:
40                 System.out.println("Sortint...");
41                 break;
42             default:
43                 System.out.println("Opció invàlida. Torna a intentar.");
44         }
45         System.out.println();
46     } while (opcio != 7);
47     scanner.close();
48 }
49
50 // Mètode per mostrar el menú d'opcions.
51 public static void mostrarMenu() {
52     System.out.println("---- Videojocs DAN ----");
53     System.out.println("1. Afegir joc a la botiga");
54     System.out.println("2. Eliminar joc de la botiga");
55     System.out.println("3. Mostrar jocs a la botiga");
56     System.out.println("4. Mostrar jocs en oferta");
57     System.out.println("5. Afegir joc a ofertes");
58     System.out.println("6. Eliminar joc d'ofertes");
59     System.out.println("7. Sortir");
60 }
61
62 // Mètode per afegir un joc a la botiga.
63 public static void afegirJoc(Scanner scanner) {
64     System.out.println("Introdueix el nom del joc: ");
65     String nom = scanner.nextLine();
66     System.out.println("Introdueix el preu del joc: ");
67     double preu = scanner.nextDouble();
68     videojocs.add(nom);
69     preus.add(preu);
70     System.out.println("Joc afegit correctament a la botiga.");
71 }
```

```
----- Videojocs DAN -----
1. Afegir joc a la botiga
2. Eliminar joc de la botiga
3. Mostrar jocs a la botiga
4. Mostrar jocs en oferta
5. Afegir joc a ofertes
6. Eliminar joc d'ofertes
7. Sortir
Selecciona una opció: 1
Introdueix el nom del joc: DOOM
Introdueix el preu del joc: 50
Joc afegit correctament a la botiga.

----- Videojocs DAN -----
1. Afegir joc a la botiga
2. Eliminar joc de la botiga
3. Mostrar jocs a la botiga
4. Mostrar jocs en oferta
5. Afegir joc a ofertes
6. Eliminar joc d'ofertes
7. Sortir
Selecciona una opció: 3
Jocs disponibles a la botiga:
- DOOM : 50.0 €

----- Videojocs DAN -----
1. Afegir joc a la botiga
2. Eliminar joc de la botiga
3. Mostrar jocs a la botiga
4. Mostrar jocs en oferta
5. Afegir joc a ofertes
6. Eliminar joc d'ofertes
7. Sortir
Selecciona una opció: 5
Introdueix el nom del joc a afegir a ofertes: DOOM
Introdueix el preu d'oferta: 30
Joc mogut a ofertes correctament.

----- Videojocs DAN -----
1. Afegir joc a la botiga
2. Eliminar joc de la botiga
3. Mostrar jocs a la botiga
4. Mostrar jocs en oferta
5. Afegir joc a ofertes
6. Eliminar joc d'ofertes
7. Sortir
Selecciona una opció: 3
Jocs disponibles a la botiga:
No hi ha jocs a la botiga.

----- Videojocs DAN -----
1. Afegir joc a la botiga
2. Eliminar joc de la botiga
3. Mostrar jocs a la botiga
4. Mostrar jocs en oferta
5. Afegir joc a ofertes
6. Eliminar joc d'ofertes
7. Sortir
Selecciona una opció: 3
Jocs disponibles a la botiga:
No hi ha jocs a la botiga.
```

The screenshot displays the Eclipse IDE interface. The left pane shows the source code for `Exercici_02_Arraylist.java`, which implements a game logic. The right pane shows the console output of the application.

Source Code (Exercici_02_Arraylist.java):

```
67 // Mètode per afegir un joc a la botiga.
68 public static void afegirJoc(Scanner scanner) {
69     System.out.print("Introdueix el nom del joc: ");
70     String nomJoc = scanner.nextLine();
71     System.out.print("Introdueix el preu del joc: ");
72     double preu = scanner.nextDouble();
73     scanner.nextLine(); // Consumim el salt de línia
74     videojocs.add(nomJoc);
75     preus.add(preu);
76     System.out.println("Joc afegit correctament a la botiga.");
77 }
78
79 // Mètode per eliminar un joc de la botiga.
80 public static void eliminarJoc(Scanner scanner) {
81     System.out.print("Introdueix el nom del joc a eliminar de la botiga: ");
82     String jocEliminar = scanner.nextLine();
83     int index = videojocs.indexOf(jocEliminar);
84     if (index != -1) {
85         videojocs.remove(index);
86         preus.remove(index);
87         System.out.println("Joc eliminat correctament de la botiga.");
88     } else {
89         System.out.println("El joc no existeix a la botiga.");
90     }
91 }
92
93 // Mètode per mostrar tots els jocs de la botiga.
94 public static void mostrarJocs() {
95     System.out.println("Jocs disponibles a la botiga:");
96     if (videojocs.isEmpty()) {
97         System.out.println("No hi ha jocs a la botiga.");
98     } else {
99         for (int i = 0; i < videojocs.size(); i++) {
100             System.out.println(i + " " + videojocs.get(i) + " : " + preus.get(i) + " €");
101         }
102     }
103 }
104
105 // Mètode per mostrar els jocs en oferta.
106 public static void mostrarJocsOferta() {
107     System.out.println("Jocs en oferta:");
108     if (videojocsOferta.isEmpty()) {
109         System.out.println("No hi ha jocs en oferta.");
110     } else {
111         for (int i = 0; i < videojocsOferta.size(); i++) {
112             System.out.println(i + " " + videojocsOferta.get(i) + " : " + preusOferta.get(i) + " €");
113         }
114     }
115 }
116
117 // Mètode per afegir un joc a la llista d'ofertes: es mou de la botiga a ofertes.
118 public static void afegirJocOferta(Scanner scanner) {
119     System.out.print("Introdueix el nom del joc a afegir a ofertes: ");
120     String jocOferta = scanner.nextLine();
121     int indexOferta = videojocs.indexOf(jocOferta);
122     if (indexOferta != -1) {
123         System.out.print("Introdueix el preu d'oferta: ");
124     }
```

Console Output:

```
Exercici_02_Arraylist [Java Application] C:\Users\Bernen\Desktop\Eclipse\plugins\org.eclipse.jdt.ui\openjdk.hotspot.jre.full.win32.x86_64.jdk
---- Videojocs DAM ----
1. Afegir joc a la botiga
2. Eliminar joc de la botiga
3. Mostrar jocs a la botiga
4. Mostrar jocs en oferta
5. Afegir joc a ofertes
6. Eliminar joc d'ofertes
7. Sortir
Selecció una opció: 3
Jocs disponibles a la botiga:
No hi ha jocs a la botiga.

---- Videojocs DAM ----
1. Afegir joc a la botiga
2. Eliminar joc de la botiga
3. Mostrar jocs a la botiga
4. Mostrar jocs en oferta
5. Afegir joc a ofertes
6. Eliminar joc d'ofertes
7. Sortir
Selecció una opció: 4
Jocs en oferta:
- DDMH : 30.0 €

---- Videojocs DAM ----
1. Afegir joc a la botiga
2. Eliminar joc de la botiga
3. Mostrar jocs a la botiga
4. Mostrar jocs en oferta
5. Afegir joc a ofertes
6. Eliminar joc d'ofertes
7. Sortir
Selecció una opció: 6
Introdueix el nom del joc a eliminar de ofertes: DDMH
Joc eliminat de ofertes correctament.

---- Videojocs DAM ----
1. Afegir joc a la botiga
2. Eliminar joc de la botiga
3. Mostrar jocs a la botiga
4. Mostrar jocs en oferta
5. Afegir joc a ofertes
6. Eliminar joc d'ofertes
7. Sortir
Selecció una opció: 3
Jocs disponibles a la botiga:
No hi ha jocs a la botiga.

---- Videojocs DAM ----
1. Afegir joc a la botiga
2. Eliminar joc de la botiga
3. Mostrar jocs a la botiga
4. Mostrar jocs en oferta
5. Afegir joc a ofertes
6. Eliminar joc d'ofertes
7. Sortir
Selecció una opció: 6
Introdueix el nom del joc a afegir a ofertes:
Introdueix el preu d'oferta:
```

```

111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
10
```

Hem declarat quatre arraylist com statics per facilitar el cridar-les desde tots els mètodes. Dues arraylist son per guardar els noms dels videojocs i els videojocs en oferta, del tipus string. I les altres dues son del tipus Double per guardar els preus normals i en oferta.

```
public class Exercici_03_Arraylist {  
static ArrayList<String> videojocs = new ArrayList<>();  
static ArrayList<Double> preus = new ArrayList<>();  
static ArrayList<String> videojocsOferta = new ArrayList<>();  
static ArrayList<Double> preusOferta = new ArrayList<>();
```

En el mètode main cridem el mètode que contè el menú i amb un switch case i un bucle do-while que es repetirà fins que l'usuari esculli l'opció de sortir declarem cada una de les opcions del menú que executarà la resta dels mètodes del codi.

```
public static void main(String[] args) {  
Scanner scanner = new Scanner(System.in);  
int opcio;  
do {  
mostrarMenu();  
opcio = scanner.nextInt();  
scanner.nextLine();  
switch (opcio) {  
  
case 1:  
afegirJoc(scanner);  
break;  
case 2:  
eliminarJoc(scanner);  
break;  
case 3:  
mostrarJocs();  
break;  
case 4:  
mostrarJocsOferta();  
break;  
case 5:  
afegirJocOferta(scanner);  
break;  
case 6:  
eliminarJocOferta(scanner);  
break;  
case 7:  
System.out.println("Sortint...");  
break;  
default:  
System.out.println("Opció invàlida. Torna a intentar.");  
}  
System.out.println();  
} while (opcio != 7);  
scanner.close();  
}
```

El primer mètode és el menú ens si mateix, mostra les diferents opcions imprimint línies de text perquè l'usuari esculli entre les diferents opcions

```
public static void mostrarMenu() {  
System.out.println("---- Videojocs DAM ----");  
System.out.println("1. Afegir joc a la botiga");  
System.out.println("2. Eliminar joc de la botiga");  
System.out.println("3. Mostrar jocs a la botiga");  
System.out.println("4. Mostrar jocs en oferta");  
System.out.println("5. Afegir joc a ofertes");  
System.out.println("6. Eliminar joc d'ofertes");  
System.out.println("7. Sortir");  
System.out.print("Selecciona una opció: ");  
}
```

Amb l'opció 1 del menú cridem el mètode per afegir un joc, primer es pregunta el nom del joc que es guarda en una variable String i després el seu preu que es guarda en una variable double.

```
public static void afegirJoc(Scanner scanner) {  
System.out.print("Introdueix el nom del joc: ");  
String nomJoc = scanner.nextLine();  
System.out.print("Introdueix el preu del joc: ");  
double preu = scanner.nextDouble();  
scanner.nextLine();
```

En les dues següents línies afegim els valors que acabem de demanar i guardats en les variables per afegir-los als arraylists. Acabem informant al usuari de que el joc ha sigut introduït correctament.

```
videojocs.add(nomJoc);  
preus.add(preu);  
System.out.println("Joc afegit correctament a la botiga.");  
}
```

Amb l'opció dos cridem el mètode per eliminar jocs, demanem al usuari el nom del joc que vol eliminar, el nom es guarda en una variable tipus string.

```
public static void eliminarJoc(Scanner scanner) {  
System.out.print("Introdueix el nom del joc a eliminar de la botiga: ");  
String jocEliminar = scanner.nextLine();
```

Busquem la posició del index que coincideix amb el nom del joc introduït, si el joc no es troba es retorna un -1.

```
int index = videojocs.indexOf(jocEliminar);
```

Si el index es diferent a -1 vol dir que existeix i esborrem el joc i el preu dels arraylist corresponents.

```
if (index != -1) {  
videojocs.remove(index);  
preus.remove(index);
```

Informem si el joc ha sigut eliminat o no s'ha trobat:

```
System.out.println("Joc eliminat correctament de la botiga.");  
} else {  
System.out.println("El joc no existeix a la botiga.");  
}  
}
```


La tercera opció del menu crida el mètode per mostrar el llistat dels jocs disponibles. Comprovem si l'arraylist es buida i si no ho és la recorrem amb un bucle for per imprimir cada joc de la llista amb el seu preu.

```
public static void mostrarJocs() {  
System.out.println("Jocs disponibles a la botiga:");  
if (videojocs.isEmpty()){  
System.out.println("No hi ha jocs a la botiga.");  
} else {  
for (int i = 0; i < videojocs.size(); i++) {  
System.out.println("- " + videojocs.get(i) + " : " + preus.get(i) + " €");  
}  
}  
}
```

L'opció quatre crida el mètode per mostrar els jocs en oferta. Comprovem si l'arraylist està buida i si no ho està la recorrem amb un bucle for i mostrem el nom i el preu dels jocs en oferta.

```
public static void mostrarJocsOferta() {  
System.out.println("Jocs en oferta:");  
if (videojocsOferta.isEmpty()){  
System.out.println("No hi ha jocs en oferta.");  
} else {  
for (int i = 0; i < videojocsOferta.size(); i++) {  
System.out.println("- " + videojocsOferta.get(i) + " : " + preusOferta.get(i)  
+ " €");  
}  
}  
}
```

Amb l'opció cinc cridem el mètode per afegir un joc al llistat d'ofertes. Demanem a l'usuari el nom del joc .

```
public static void afegirJocOferta(Scanner scanner) {  
System.out.print("Introdueix el nom del joc a afegir a ofertes: ");  
String jocOferta = scanner.nextLine();
```

Comprovem si està dintre del arraylist dels jocs sense oferta.

```
int indexOferta = videojocs.indexOf(jocOferta);  
if (indexOferta != -1) {
```

Si el troba demanem el preu del joc en oferta i afegim tant el nom com el preu en els seus arrays corresponents i els eliminem dels arraylists de jocs i preus sense ofertes.

```
System.out.print("Introdueix el preu d'oferta: ");  
double preuOfertaJoc = scanner.nextDouble();  
scanner.nextLine();  
videojocsOferta.add(jocOferta);  
preusOferta.add(preuOfertaJoc);  
videojocs.remove(indexOferta);  
preus.remove(indexOferta);
```

Informem de si l'acció s'ha realitzat correctament o no s'ha trobat el nom del joc.

```
System.out.println("Joc mogut a ofertes correctament.");  
} else {  
System.out.println("El joc no existeix a la botiga.");  
}  
}
```

L'última opció que crida un mètode és la número 6, en aquest cas eliminarem el joc del llistat d'ofertes. Demanem al usuari el nom del joc, busquem si el nom del joc està en l'array list i sí és així procedim a eliminar-ne tant el nom com el preu d'oferta.

```
public static void eliminarJocOferta(Scanner scanner) {
    System.out.print("Introdueix el nom del joc a eliminar de ofertes: ");
    String jocEliminarOferta = scanner.nextLine();
    int indexOfertaEliminar = videojocsOferta.indexOf(jocEliminarOferta);
    if (indexOfertaEliminar != -1) {
        videojocsOferta.remove(indexOfertaEliminar);
        preusOferta.remove(indexOfertaEliminar);
        System.out.println("Joc eliminat de ofertes correctament.");
    } else {
        System.out.println("El joc no existeix a la llista d'ofertes.");
    }
}
}
```

Exercici 4

Ets encarregat d'un restaurant petit amb 5 taules. Quan arriba un client, el seu nom s'apunta a la taula que ocupen (per exemple, a la taula 3 s'asseu "Anna"). Has de fer un programa que permeti:

- Assignar clients a cada taula.
- Canviar el nom d'una taula si arriben altres clients. (5 taules màxim, alliberar taula per afegir nou clients)
- Mostrar quins clients hi ha asseguts a cada taula.

```
1 package part_02;
2
3 import java.util.ArrayList;
4 import java.util.Scanner;
5
6 public class Exercici_04_Arraylist {
7
8     // ArrayList per gestionar les 5 taules (inicialment buides)
9     static ArrayList<String> taules = new ArrayList<>(5);
10
11     public static void main(String[] args) {
12         Scanner scanner = new Scanner(System.in);
13         int opcio;
14
15         // Inicialitzem l'ArrayList amb 5 posicions null
16         for (int i = 0; i < 5; i++) {
17             taules.add(null);
18         }
19
20         do {
21             mostrarMenu();
22             opcio = scanner.nextInt();
23             scanner.nextLine();
24
25             switch (opcio) {
26                 case 1:
27                     assignarClient(scanner);
28                     break;
29                 case 2:
30                     canviarClient(scanner);
31                     break;
32                 case 3:
33                     alliberarTaula(scanner);
34                     break;
35                 case 4:
36                     mostrarTaules();
37                     break;
38                 case 5:
39                     System.out.println("Sortint...");
40                     break;
41                 default:
42                     System.out.println("Opció invàlida. Torna a intentar.");
43             }
44             System.out.println();
45         } while (opcio != 5);
46     }
47     scanner.close();
48 }
```

```
---- Restaurant ----
1. Assignar client a una taula
2. Canviar el nom d'una taula (nou client)
3. Alliberar una taula
4. Mostrar les taules
5. Sortir
Selecciona una opció: 1
Introdueix el número de taula (1-5): 1
Introdueix el nom del client: Edim
Client assignat correctament a la taula 1.

---- Restaurant ----
1. Assignar client a una taula
2. Canviar el nom d'una taula (nou client)
3. Alliberar una taula
4. Mostrar les taules
5. Sortir
Selecciona una opció: 1
Introdueix el número de taula (1-5): 2
Introdueix el nom del client: Nidaa
Client assignat correctament a la taula 2.

---- Restaurant ----
1. Assignar client a una taula
2. Canviar el nom d'una taula (nou client)
3. Alliberar una taula
4. Mostrar les taules
5. Sortir
Selecciona una opció: 1
Introdueix el número de taula (1-5): 3
Introdueix el nom del client: Canavaca
Client assignat correctament a la taula 3.

---- Restaurant ----
1. Assignar client a una taula
2. Canviar el nom d'una taula (nou client)
3. Alliberar una taula
4. Mostrar les taules
5. Sortir
Selecciona una opció: 4
Estat de les taules:
Taula 1: Edim
Taula 2: Nidaa
Taula 3: Canavaca
Taula 4: BUIT
Taula 5: BUIT
```

```
49
50 // Mètode per mostrar el menú d'opcions.
51 public static void mostrarMenu() {
52     System.out.println("---- Restaurant ----");
53     System.out.println("1. Assignar client a una taula");
54     System.out.println("2. Canviar el nom d'una taula (nou client)");
55     System.out.println("3. Alliberar una taula");
56     System.out.println("4. Mostrar les taules");
57     System.out.println("5. Sortir");
58     System.out.print("Selecciona una opció: ");
59 }
60
61 // Mètode per assignar un client a una taula si està buida.
62 public static void assignarClient(Scanner scanner) {
63     System.out.print("Introdueix el número de taula (1-5): ");
64     int taula = scanner.nextInt();
65     scanner.nextLine(); // Consumeix el salt de línia
66
67     if (taula < 1 || taula > 5) {
68         System.out.println("Número de taula invàlid.");
69         return;
70     }
71
72     if (taules.get(taula - 1) != null) {
73         System.out.println("La taula ja està ocupada.");
74     } else {
75         System.out.print("Introdueix el nom del client: ");
76         String client = scanner.nextLine();
77         taules.set(taula - 1, client);
78         System.out.println("Client assignat correctament a la taula " + taula + ".");
79     }
80 }
81
82 // Mètode per canviar el client d'una taula.
83 public static void canviarClient(Scanner scanner) {
84     System.out.print("Introdueix el número de taula (1-5) per canviar el client: ");
85     int taula = scanner.nextInt();
86     scanner.nextLine(); // Consumeix el salt de línia
87
88     if (taula < 1 || taula > 5) {
89         System.out.println("Número de taula invàlid.");
90         return;
91     }
92
93     System.out.print("Introdueix el nou nom del client: ");
94     String client = scanner.nextLine();
95     taules.set(taula - 1, client);
96 }
```

Console output:

```
---- Restaurant ----
1. Assignar client a una taula
2. Canviar el nom d'una taula (nou client)
3. Alliberar una taula
4. Mostrar les taules
5. Sortir
Selecciona una opció: 4
Estat de les taules:
Taula 1: Edim
Taula 2: Nídea
Taula 3: Caravaca
Taula 4: BUIT
Taula 5: BUIT

---- Restaurant ----
1. Assignar client a una taula
2. Canviar el nom d'una taula (nou client)
3. Alliberar una taula
4. Mostrar les taules
5. Sortir
Selecciona una opció: 3
Introdueix el número de taula (1-5) per alliberar: 1
La taula 1 ha estat alliberada.

---- Restaurant ----
1. Assignar client a una taula
2. Canviar el nom d'una taula (nou client)
3. Alliberar una taula
4. Mostrar les taules
5. Sortir
Selecciona una opció: 4
Estat de les taules:
Taula 1: BUIT
Taula 2: Nídea
Taula 3: Caravaca
Taula 4: BUIT
Taula 5: BUIT

---- Restaurant ----
1. Assignar client a una taula
2. Canviar el nom d'una taula (nou client)
3. Alliberar una taula
4. Mostrar les taules
5. Sortir
Selecciona una opció: 3
Introdueix el número de taula (1-5) per alliberar: 1
La taula 1 ha estat alliberada.

---- Restaurant ----
1. Assignar client a una taula
2. Canviar el nom d'una taula (nou client)
3. Alliberar una taula
4. Mostrar les taules
5. Sortir
Selecciona una opció: 4
Estat de les taules:
Taula 1: BUIT
Taula 2: Nídea
Taula 3: Caravaca
Taula 4: BUIT
Taula 5: BUIT
```

```
84 System.out.print("Introdueix el número de taula (1-5) per canviar el client: ");
85 int taula = scanner.nextInt();
86 scanner.nextLine(); // Consumeix el salt de línia
87
88 if (taula < 1 || taula > 5) {
89     System.out.println("Número de taula invàlid.");
90     return;
91 }
92
93 System.out.print("Introdueix el nou nom del client: ");
94 String client = scanner.nextLine();
95 taules.set(taula - 1, client);
96 System.out.println("5a ha canviat el client de la taula " + taula + ".");
97 }
98
99 // Mètode per alliberar una taula.
100 public static void alliberarTaula(Scanner scanner) {
101     System.out.print("Introdueix el número de taula (1-5) per alliberar: ");
102     int taula = scanner.nextInt();
103     scanner.nextLine();
104
105     if (taula < 1 || taula > 5) {
106         System.out.println("Número de taula invàlid.");
107         return;
108     }
109
110     if (taules.get(taula - 1) == null) {
111         System.out.println("La taula ja està buida.");
112     } else {
113         taules.set(taula - 1, null);
114         System.out.println("La taula " + taula + " ha estat alliberada.");
115     }
116 }
117
118 // Mètode per mostrar l'estat de totes les taules.
119 public static void mostrarTaules() {
120     System.out.println("Estat de les taules:");
121     for (int i = 0; i < 5; i++) {
122         if (taules.get(i) == null) {
123             System.out.println("Taula " + (i + 1) + ": BUIT");
124         } else {
125             System.out.println("Taula " + (i + 1) + ": " + taules.get(i));
126         }
127     }
128 }
129 }
130 }
```

Console output:

```
---- Restaurant ----
1. Assignar client a una taula
2. Canviar el nom d'una taula (nou client)
3. Alliberar una taula
4. Mostrar les taules
5. Sortir
Selecciona una opció: 4
Estat de les taules:
Taula 1: BUIT
Taula 2: Nídea
Taula 3: Caravaca
Taula 4: BUIT
Taula 5: BUIT

---- Restaurant ----
1. Assignar client a una taula
2. Canviar el nom d'una taula (nou client)
3. Alliberar una taula
4. Mostrar les taules
5. Sortir
Selecciona una opció: 3
Introdueix el número de taula (1-5) per alliberar: 1
La taula 1 ha estat alliberada.

---- Restaurant ----
1. Assignar client a una taula
2. Canviar el nom d'una taula (nou client)
3. Alliberar una taula
4. Mostrar les taules
5. Sortir
Selecciona una opció: 4
Estat de les taules:
Taula 1: BUIT
Taula 2: Nídea
Taula 3: Caravaca
Taula 4: BUIT
Taula 5: BUIT
```

Abans del mètode main declarem un arraylist static per que sigui accessible desde tots els mètodes del tipus string amb un tamany de 5.

```
public class Exercici_04_Arraylist {  
static ArrayList<String> taules = new ArrayList<>(5);
```

En el metode main utilitzem un bucle for per omplir l'arraylist de valors null, aixó ens permetrà modificar les diferents taules sense importar-ne l'ordre.

```
public static void main(String[] args) {  
Scanner scanner = new Scanner(System.in);  
int opcio;  
for (int i = 0; i < 5; i++) {  
taules.add(null);  
}
```

A continuació amb un bucle do-while i un switch case estructurarem les diferents opcions d'un menú principal i els mètodes que cridaran cada una de les seves opcions. Com sempre aquest bucle s'executarà repetidament fins que l'usuari no premi l'opció de sortir-ne.

```
do {  
mostrarMenu();  
opcio = scanner.nextInt();  
scanner.nextLine();  
switch (opcio) {  
case 1:  
assignarClient(scanner);  
break;  
case 2:  
canviarClient(scanner);  
break;  
case 3:  
alliberarTaula(scanner);  
break;  
case 4:  
mostrarTaules();  
break;  
case 5:  
System.out.println("Sortint...");  
break;  
default:  
System.out.println("Opció invàlida. Torna a intentar.");  
}  
System.out.println();  
} while (opcio != 5);  
scanner.close();  
}
```

En el primer mètode mostrem el menu al usuari imprimint les diferents opcions entre les que pot triar.

```
public static void mostrarMenu() {  
System.out.println("---- Restaurant ----");  
System.out.println("1. Assignar client a una taula");  
System.out.println("2. Canviar el nom d'una taula (nou client)");  
System.out.println("3. Alliberar una taula");  
System.out.println("4. Mostrar les taules");  
System.out.println("5. Sortir");  
System.out.print("Selecciona una opció: ");  
}
```

Amb l'opció 1 podem assignar un client, primer demanarem a quina taula el volem assignar de la 1 a la 5.

```
public static void assignarClient(Scanner scanner) {  
    System.out.print("Introdueix el número de taula (1-5): ");  
    int taula = scanner.nextInt();  
    scanner.nextLine();  
    Comprovem que s'ha introduït un numero de taula correcte.  
    if (taula < 1 || taula > 5) {  
        System.out.println("Número de taula invàlid.");  
        return;  
    }  
    Comprovem si la taula està ocupada i si no ho està demanem al usuari que introdueixi  
    el nom del client per assignar-l'ho.  
    if (taules.get(taula - 1) != null) {  
        System.out.println("La taula ja està ocupada.");  
    } else {  
        System.out.print("Introdueix el nom del client: ");  
        String client = scanner.nextLine();  
        taules.set(taula - 1, client);  
        System.out.println("Client assignat correctament a la taula " + taula + ".");  
    }  
}
```

Comprovem si la taula està ocupada i si no ho està demanem al usuari que introdueixi el nom del client per assignar-l'ho.

```
if (taules.get(taula - 1) != null) {  
    System.out.println("La taula ja està ocupada.");  
} else {  
    System.out.print("Introdueix el nom del client: ");  
    String client = scanner.nextLine();  
    taules.set(taula - 1, client);  
    System.out.println("Client assignat correctament a la taula " + taula + ".");  
}  
}
```

La segona opció del menú crida el mètode per canviar de client, com en el mètode anterior demanem al usuari que introdueix-hi el número de la taula a la que vol fer el canvi. Comprovem que sigui una taula correcte de la 1 al 5.

```
public static void canviarClient(Scanner scanner) {  
    System.out.print("Introdueix el número de taula (1-5) per canviar el client:  
    ");  
    int taula = scanner.nextInt();  
    scanner.nextLine(); // Consumeix el salt de línia  
    if (taula < 1 || taula > 5) {  
        System.out.println("Número de taula invàlid.");  
        return;  
    }  
    Un cop feta la comprovació es demana el nom del nou client i s'assigna.  
    System.out.print("Introdueix el nou nom del client: ");  
    String client = scanner.nextLine();  
    taules.set(taula - 1, client);  
    System.out.println("S'ha canviat el client de la taula " + taula + ".");  
}
```

Un cop feta la comprovació es demana el nom del nou client i s'assigna.

```
System.out.print("Introdueix el nou nom del client: ");  
String client = scanner.nextLine();  
taules.set(taula - 1, client);  
System.out.println("S'ha canviat el client de la taula " + taula + ".");  
}
```

La tercera opció del menu crida el mètode per deixar una taula lliure, com en els dos mètodes anteriors el primer que farem es demanar a l'usuari que introdueix-hi el número de la taula i comprovarem que el número introduït estigui entre el 1 i el 5.

```
public static void alliberarTaula(Scanner scanner) {  
    System.out.print("Introdueix el número de taula (1-5) per alliberar: ");  
    int taula = scanner.nextInt();  
    scanner.nextLine();  
    if (taula < 1 || taula > 5) {  
        System.out.println("Número de taula invàlid.");  
        return;  
    }  
}
```

Si és correcte també comprovem que la taula no estigui buida ja.

```
if (taules.get(taula - 1) == null) {  
    System.out.println("La taula ja està buida.");  
    Si està ocupada la buidarem.  
} else {  
    taules.set(taula - 1, null);  
    System.out.println("La taula " + taula + " ha estat alliberada.");  
}
```

L'últim mètode es crida amb l'opció quatre del menú i es per mostrar si les taules estan buides o plenes i si estan plenes que indiqui per quins clients. Recorrem un bucle for i si trobem un null indiquem que la taula amb aquest index està buida. I si no trobem null imprimim el nom del client en aquest index.

```
public static void mostrarTaules() {  
    System.out.println("Estat de les taules:");  
    for (int i = 0; i < 5; i++) {  
        if (taules.get(i) == null) {  
            System.out.println("Taula " + (i + 1) + ": BUIT");  
        } else {  
            System.out.println("Taula " + (i + 1) + ": " + taules.get(i));  
        }  
    }  
}
```

Exercici 5

Estàs fent la compra setmanal i vols fer una llista amb tot el que necessites comprar. Comences apuntant alguns productes, però t'adones que te n'has oblidat d'un i el vols afegir en una posició concreta. També t'adones que has apuntat dos cops un producte i l'has d'eliminar. El programa ha de permetre:

- Afegir productes.
- Afegir un producte en una posició concreta.
- Eliminar un producte.
- Mostrar la llista final de la compra.

```
1 package part_02;  
2  
3 import java.util.ArrayList;  
4  
5  
6 public class Exercici_05_Arraylist {  
7  
8     static ArrayList<String> compra = new ArrayList<>();  
9  
10    public static void main(String[] args) {  
11        Scanner scanner = new Scanner(System.in);  
12        int opcio;  
13  
14        do {  
15            mostrarMenu();  
16            opcio = scanner.nextInt();  
17            scanner.nextLine();  
18  
19            switch (opcio) {  
20                case 1:  
21                    afegirProducte(scanner);  
22                    break;  
23                case 2:  
24                    afegirProductePosicio(scanner);  
25                    break;  
26                case 3:  
27                    eliminarProducte(scanner);  
28                    break;  
29                case 4:  
30                    mostrarLlistaCompra();  
31                    break;  
32                case 5:  
33                    System.out.println("Sortint...");  
34                    break;  
35                default:  
36                    System.out.println("Opció invàlida. Torna a provar.");  
37            }  
38            System.out.println();  
39        } while (opcio != 5);  
40        scanner.close();  
41    }  
42  
43    // Mètode per mostrar el menú  
44    public static void mostrarMenu() {  
45        System.out.println("---- Llista de la compra ----");  
46        System.out.println("1. Afegir producte");  
47        System.out.println("2. Afegir producte en una posició concreta");  
48    }  
49  
50    // Mètode per afegir producte  
51    public static void afegirProducte(Scanner scanner) {  
52        System.out.println("Introdueix el nom del producte a afegir: sal");  
53        String producte = scanner.nextLine();  
54        compra.add(producte);  
55        System.out.println("Producte afegit correctament.");  
56    }  
57  
58    // Mètode per afegir producte en una posició concreta  
59    public static void afegirProductePosicio(Scanner scanner) {  
60        System.out.println("Introdueix el nom del producte a afegir: oli");  
61        String producte = scanner.nextLine();  
62        System.out.println("Introdueix la posició on vols inserir-lo (començant per 0): 2");  
63        int posicio = scanner.nextInt();  
64        compra.add(posicio, producte);  
65        System.out.println("Producte afegit a la posició " + posicio + " correctament.");  
66    }  
67  
68    // Mètode per eliminar producte  
69    public static void eliminarProducte(Scanner scanner) {  
70        System.out.println("Introdueix el nom del producte a eliminar: sucre");  
71        String producte = scanner.nextLine();  
72        if (compra.contains(producte)) {  
73            compra.remove(producte);  
74            System.out.println("Producte eliminat correctament.");  
75        } else {  
76            System.out.println("Producte no trobat.");  
77        }  
78    }  
79  
80    // Mètode per mostrar la llista final de la compra  
81    public static void mostrarLlistaCompra() {  
82        System.out.println("---- Llista de la compra ----");  
83        for (int i = 0; i < compra.size(); i++) {  
84            System.out.println(i + 1 + ". " + compra.get(i));  
85        }  
86        System.out.println("Llista final de la compra:");  
87        for (int i = 0; i < compra.size(); i++) {  
88            System.out.println(i + 1 + ". " + compra.get(i));  
89        }  
90    }  
91}
```

The screenshot shows the Eclipse IDE with the file `Exercici_05_Arraylist.java` open. The code implements the following methods:

- `mostrarMenu()`: Displays a menu with options: 1. Afegir producte, 2. Afegir producte en una posició concreta, 3. Eliminar producte, 4. Mostrar la llista final de la compra, 5. Sortir.
- `afegirProductePosicio(Scanner scanner)`: Prompts the user to enter a product name and a position (0 to size-1). It adds the product to the `compra` ArrayList at the specified position.
- `eliminarProducte(Scanner scanner)`: Prompts the user to enter a product name and removes it from the `compra` ArrayList if it exists.

The console output shows the menu being displayed and the user interacting with the program to add products at specific positions.

The screenshot shows the Eclipse IDE with the file `Exercici_05_Arraylist.java` open. The code implements the following methods:

- `eliminarProducte(Scanner scanner)`: Prompts the user to enter a product name and removes it from the `compra` ArrayList if it exists.
- `mostrarLlistaCompra()`: Displays the final list of products in the `compra` ArrayList.

The console output shows the user entering a product name to be removed and the final list of products being displayed.

Abans del mètode main declarem un arraylist de string estatica.

```
public class Exercici_05_Arraylist {  
static ArrayList<String> compra = new ArrayList<>();
```

En el metode main com en els dos exercics anteriors creem un menu am un bucle do-while i amb un switch case. Cada opció cridarà els diferents mètodes del codi.

```
public static void main(String[] args) {  
Scanner scanner = new Scanner(System.in);  
int opcio;  
do {  
mostrarMenu();  
opcio = scanner.nextInt();  
scanner.nextLine();  
switch (opcio) {  
case 1:  
afegirProducte(scanner);  
break;  
case 2:  
afegirProductePosicio(scanner);  
break;  
case 3:  
eliminarProducte(scanner);  
break;  
case 4:  
mostrarLlistaCompra();  
break;  
case 5:  
System.out.println("Sortint...");  
break;  
default:  
System.out.println("Opció invàlida. Torna a provar.");  
}  
System.out.println();  
} while (opcio != 5);  
scanner.close();  
}
```

El primer mètode torna a ser el menú amb les opcions impreses per que l'usuari esculli entre elles.

```
public static void mostrarMenu() {  
System.out.println("---- Llista de la compra ----");  
System.out.println("1. Afegir producte");  
System.out.println("2. Afegir producte en una posició concreta");  
System.out.println("3. Eliminar producte");  
System.out.println("4. Mostrar la llista final de la compra");  
System.out.println("5. Sortir");  
System.out.print("Selecciona una opció: ");  
}
```

Amb l'opció 1 cridem el mètode per afegir un producte a la llista, demanem al usuari que introdueix-hi el nom del producte i afegim al arraylist amb una confirmació.

```
public static void afegirProducte(Scanner scanner) {  
System.out.print("Introdueix el nom del producte a afegir: ");  
String producte = scanner.nextLine();  
compra.add(producte);  
System.out.println("Producte afegit correctament.");  
}
```


La segona opció crida el metode per afegir un producte pero determinant en quina opció el volem introduir. Demanem el nom del producte i la posició desitjada.

```
public static void afegirProductePosicio(Scanner scanner) {  
    System.out.print("Introdueix el nom del producte a afegir: ");  
    String producte = scanner.nextLine();  
    System.out.print("Introdueix la posició on vols inserir-lo (començant per 0):  
    ");  
    int posicio = scanner.nextInt();  
    scanner.nextLine();
```

Comprovem que la posició desitjada està creada previamente al arraylist, de no ser així s'afegirà a la ultima posició.

```
if (posicio < 0 || posicio > compra.size()) {  
    System.out.println("La posició és invàlida. S'afegeix al final de la llista.");  
    compra.add(producte);
```

Si existeix algun producte a la posició desitjada s'afegeix i la resta de productes es mouen ja que és una de les funcions de les arraylist.

```
    } else {  
        compra.add(posicio, producte);  
        System.out.println("Producte afegit a la posició " + posicio + "  
        correctament.");  
    }  
}
```

L'opció tres crida el metode per eliminar un producte. Demanem el nom del producte i declarem un booleà per comprovar si està eliminat o no el producte i retornar un true o un false segons els resultat.

```
public static void eliminarProducte(Scanner scanner) {  
    System.out.print("Introdueix el nom del producte a eliminar: ");  
    String producte = scanner.nextLine();  
    boolean eliminat = compra.remove(producte);  
    if (eliminat) {  
        System.out.println("Producte eliminat correctament.");  
    } else {  
        System.out.println("El producte no es troba a la llista.");  
    }  
}
```

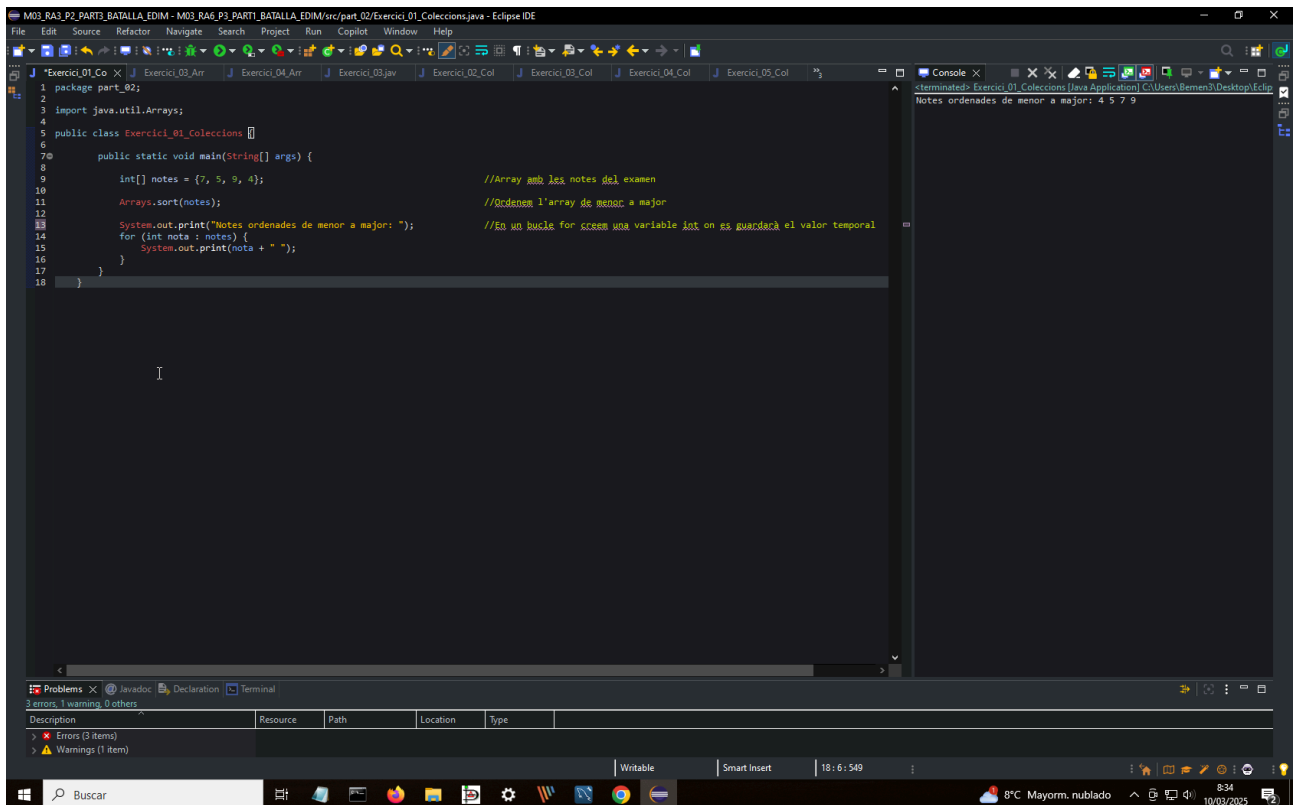
L'últim i la quarta opció crida el mètode per mostrar la llista de la compra. Comprovem que el arraylist no estigui buit i si no ho està recorrem un bucle for i anem imprimint els noms dels productes seguint l'index.

```
public static void mostrarLlistaCompra() {  
    System.out.println("Llista final de la compra:");  
    if (compra.isEmpty()) {  
        System.out.println("La llista està buida.");  
    } else {  
        for (int i = 0; i < compra.size(); i++) {  
            System.out.println(i + ". " + compra.get(i));  
        }  
    }  
}
```

COL·LECCIONS

Exercici 1

Tens una llista amb les notes d'un examen (per exemple, 7, 5, 9, 4). Fes un programa que ordeni les notes de menor a major i les mostri per pantalla.



The screenshot shows the Eclipse IDE with a Java project named 'M03_RA3_P3_PART3_BATALLA_EDIM'. The main editor displays the file 'Exercici_01_Coleccions.java'. The code defines a package 'part_02', imports 'java.util.Arrays', and contains a class 'Exercici_01_Coleccions' with a 'main' method. The 'main' method initializes an array 'notes' with values {7, 5, 9, 4}, sorts it using 'Arrays.sort', and prints the sorted array using a 'for' loop. The console on the right shows the output: 'Notes ordenades de menor a major: 4 5 7 9'. The bottom status bar indicates '18: 6: 549' and '8°C Mayorm. nublado'.

```
1 package part_02;
2
3 import java.util.Arrays;
4
5 public class Exercici_01_Coleccions {
6
7     public static void main(String[] args) {
8
9         int[] notes = {7, 5, 9, 4};           //Array amb les notes del examen
10
11         Arrays.sort(notes);                  //Ordenem l'array de menor a major
12
13         System.out.print("Notes ordenades de menor a major: ");
14         for (int nota : notes) {             //En un bucle for cream una variable int on es guarderà el valor temporal
15             System.out.print(nota + " ");
16         }
17     }
18 }
```

En aquest exercici declarem un array amb quatre notes i amb la funció `Arrays.sort` ordenem els valors de menor a major.

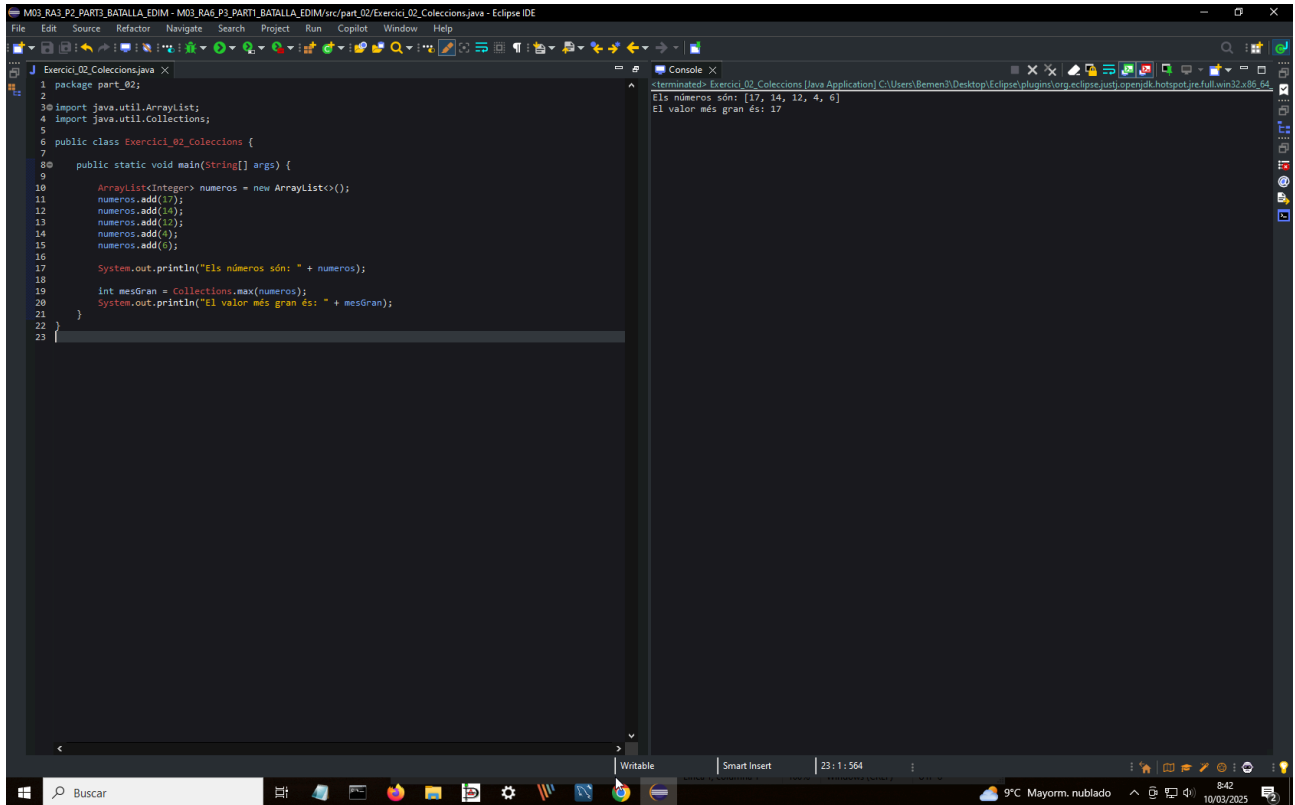
```
public static void main(String[] args) {  
int[] notes = {7, 5, 9, 4};  
Arrays.sort(notes);
```

A continuació els imprimim per pantalla amb un bucle `for` que itera sobre cada element de notes i l'assigna a `nota`.

```
System.out.print("Notes ordenades de menor a major: ");  
for (int nota : notes) {  
System.out.print(nota + " ");  
}  
}  
}
```

Exercici 2

En una llista de números enters, vols saber quin és el més gran. Fes un programa que creï la llista, la mostri i després digui el valor més gran.



The screenshot shows the Eclipse IDE with a Java file named `Exercici_02_Coleccions.java` open. The code defines a package `part_02`, imports `java.util.ArrayList` and `java.util.Collections`, and contains a `main` method. In the `main` method, an `ArrayList<Integer>` named `numeros` is created and populated with the values 17, 14, 12, 4, and 6. The list is printed to the console, and then the maximum value is found using `Collections.max(numeros)` and printed.

```
1 package part_02;
2
3 import java.util.ArrayList;
4 import java.util.Collections;
5
6 public class Exercici_02_Coleccions {
7
8     public static void main(String[] args) {
9
10         ArrayList<Integer> numeros = new ArrayList<>();
11         numeros.add(17);
12         numeros.add(14);
13         numeros.add(12);
14         numeros.add(4);
15         numeros.add(6);
16
17         System.out.println("Els números són: " + numeros);
18
19         int mesGran = Collections.max(numeros);
20         System.out.println("El valor més gran és: " + mesGran);
21     }
22 }
23
```

The console output shows the list of numbers and the maximum value:

```
<terminated> Exercici_02_Coleccions [Java Application] C:\Users\Bemen3\Desktop\Eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64
Els números són: [17, 14, 12, 4, 6]
El valor més gran és: 17
```

En el mètode `main` creem un `ArrayList` de integers anomenada `numeros` i l'omplim amb diferents valors amb la funció `.add`

```
public static void main(String[] args) {  
ArrayList<Integer> numeros = new ArrayList<>();  
numeros.add(17);  
numeros.add(14);  
numeros.add(12);  
numeros.add(4);  
numeros.add(6);
```

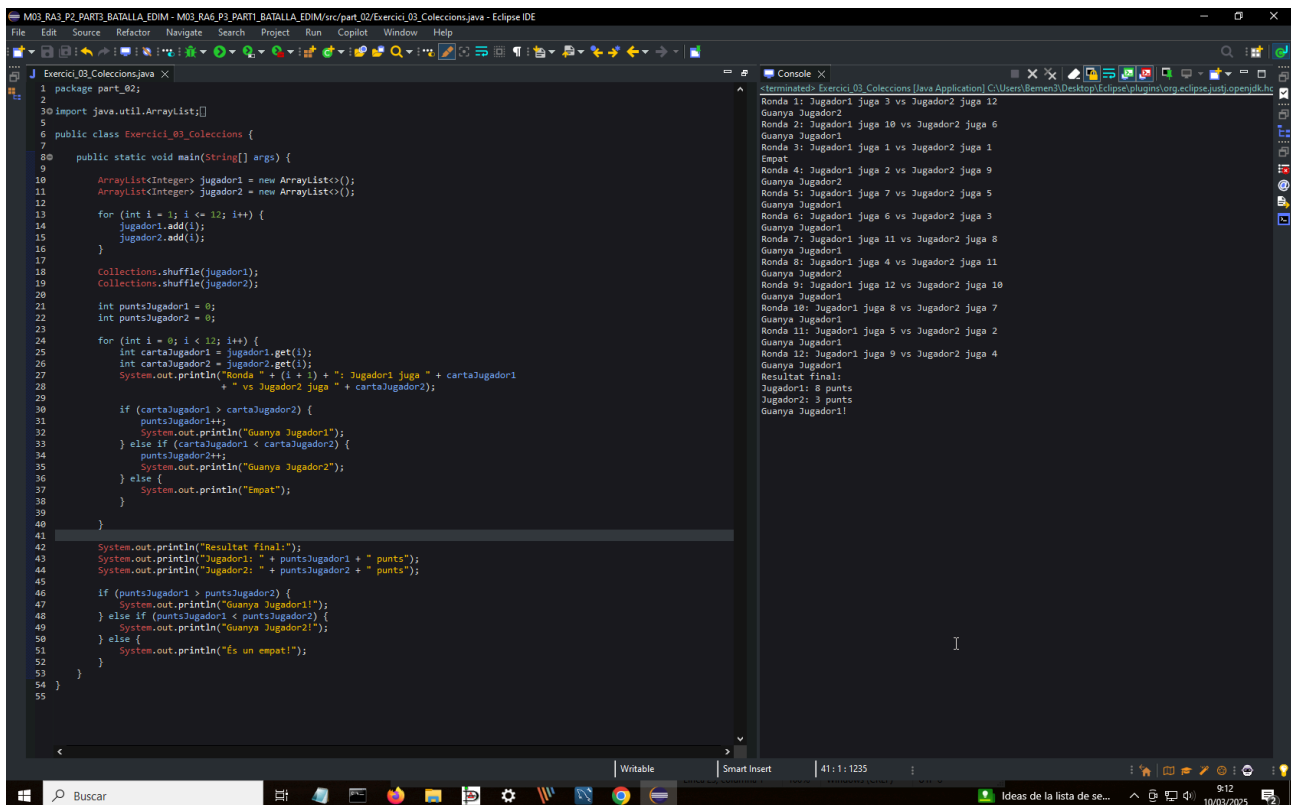
Amb la funció `Collections.max` recorrem el `ArrayList` i retornem el valor més gran, el guardem en una variable i l'imprimim per pantalla.

```
System.out.println("Els números són: " + numeros);  
int mesGran = Collections.max(numeros);  
System.out.println("El valor més gran és: " + mesGran);  
}  
}
```

Exercici 3 - Batalla de cartes

Tens 2 llistes amb les cartes d'un pal d'una baralla espanyola, del 1 al 12 per a cada jugador. Fes un programa que,

- barregi aleatòriament l'ordre de les cartes.
- Hi haurà 12 rondes i el sistema tirarà de forma ordenada les cartes de les posicions del array.
- Per a cada enfrontament la carte que tingui el número més alt, guanyarà 1 punt.
- Al final has de mostrar el resultat.



```
1 package part_02;
2
3 import java.util.ArrayList;
4
5 public class Exercici_03_Coleccions {
6
7     public static void main(String[] args) {
8
9         ArrayList<Integer> jugador1 = new ArrayList<>();
10        ArrayList<Integer> jugador2 = new ArrayList<>();
11
12        for (int i = 1; i <= 12; i++) {
13            jugador1.add(i);
14            jugador2.add(i);
15        }
16
17        Collections.shuffle(jugador1);
18        Collections.shuffle(jugador2);
19
20        int puntsJugador1 = 0;
21        int puntsJugador2 = 0;
22
23        for (int i = 0; i < 12; i++) {
24            int cartaJugador1 = jugador1.get(i);
25            int cartaJugador2 = jugador2.get(i);
26            System.out.println("Ronda " + (i + 1) + ": Jugador1 juga " + cartaJugador1
27                               + " vs Jugador2 juga " + cartaJugador2);
28
29            if (cartaJugador1 > cartaJugador2) {
30                puntsJugador1++;
31                System.out.println("Guanya Jugador1");
32            } else if (cartaJugador1 < cartaJugador2) {
33                puntsJugador2++;
34                System.out.println("Guanya Jugador2");
35            } else {
36                System.out.println("Empat");
37            }
38        }
39
40        System.out.println("Resultat final:");
41        System.out.println("Jugador1: " + puntsJugador1 + " punts");
42        System.out.println("Jugador2: " + puntsJugador2 + " punts");
43
44        if (puntsJugador1 > puntsJugador2) {
45            System.out.println("Guanya Jugador1");
46        } else if (puntsJugador1 < puntsJugador2) {
47            System.out.println("Guanya Jugador2");
48        } else {
49            System.out.println("Es un empat");
50        }
51    }
52 }
53
54 }
55 }
```

Console Output:

```
<terminated> Exercici_03_Coleccions [Java Application] C:\Users\Beren3\Desktop\Eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.jre\bin\java.exe
Ronda 1: Jugador1 juga 3 vs Jugador2 juga 12
Guanya Jugador2
Ronda 2: Jugador1 juga 10 vs Jugador2 juga 6
Guanya Jugador1
Ronda 3: Jugador1 juga 1 vs Jugador2 juga 1
Empat
Ronda 4: Jugador1 juga 2 vs Jugador2 juga 9
Guanya Jugador2
Ronda 5: Jugador1 juga 7 vs Jugador2 juga 5
Guanya Jugador1
Ronda 6: Jugador1 juga 6 vs Jugador2 juga 3
Guanya Jugador1
Ronda 7: Jugador1 juga 11 vs Jugador2 juga 8
Guanya Jugador1
Ronda 8: Jugador1 juga 4 vs Jugador2 juga 11
Guanya Jugador2
Ronda 9: Jugador1 juga 12 vs Jugador2 juga 10
Guanya Jugador1
Ronda 10: Jugador1 juga 8 vs Jugador2 juga 7
Guanya Jugador1
Ronda 11: Jugador1 juga 5 vs Jugador2 juga 2
Guanya Jugador1
Ronda 12: Jugador1 juga 9 vs Jugador2 juga 4
Guanya Jugador1
Resultat final:
Jugador1: 8 punts
Jugador2: 3 punts
Guanya Jugador1!
```

En el mètode main declarem dues arraylist de integers una per cada jugador que representa una baralla de dotze cartes.

```
public static void main(String[] args) {  
ArrayList<Integer> jugador1 = new ArrayList<>();  
ArrayList<Integer> jugador2 = new ArrayList<>();
```

Amb un bucle for afegim les cartes del 1 al 12 iguals per cada jugador.

```
for (int i = 1; i <= 12; i++) {  
jugador1.add(i);  
jugador2.add(i);  
}
```

Amb el mètode Collections.shuffle barregem aleatòriament els elements de les dues arraylists.

```
Collections.shuffle(jugador1);  
Collections.shuffle(jugador2);
```

Iniciem dos variables per comptabilitzar els punts per cada un dels jugadors.

```
int puntsJugador1 = 0;  
int puntsJugador2 = 0;
```

Tornem a recorre un bucle for que itera 12 vegades i es reparteix seguint l'índex una carta per cada jugador.

```
for (int i = 0; i < 12; i++) {  
    int cartaJugador1 = jugador1.get(i);  
    int cartaJugador2 = jugador2.get(i);
```

Mostrem per la consola cada ronda i les cartes que juga cada jugador.

```
System.out.println("Ronda " + (i + 1) + ": Jugador1 juga " + cartaJugador1  
+ " vs Jugador2 juga " + cartaJugador2);
```

Si la carta del jugador 1 es major que la carta del jugador 2 es suma un punt al comptador del jugador 1 i informem que guanya aquesta ronda.

```
if (cartaJugador1 > cartaJugador2) {  
    puntsJugador1++;  
    System.out.println("Guanya Jugador1");
```

Si no es compleix la condició anterior voldrà dir que la carta del jugador és major que la carta del jugador 1 i es sumarà un punt al comptador del jugador 2 i informem que guanya la ronda.

```
} else if (cartaJugador1 < cartaJugador2) {  
    puntsJugador2++;
```

```
System.out.println("Guanya Jugador2");
```

```
} else {  
    System.out.println("Empat");  
}  
}
```

Mostrem la suma dels punts que han fet cada un dels jugadors, les rondes que han guanyat.

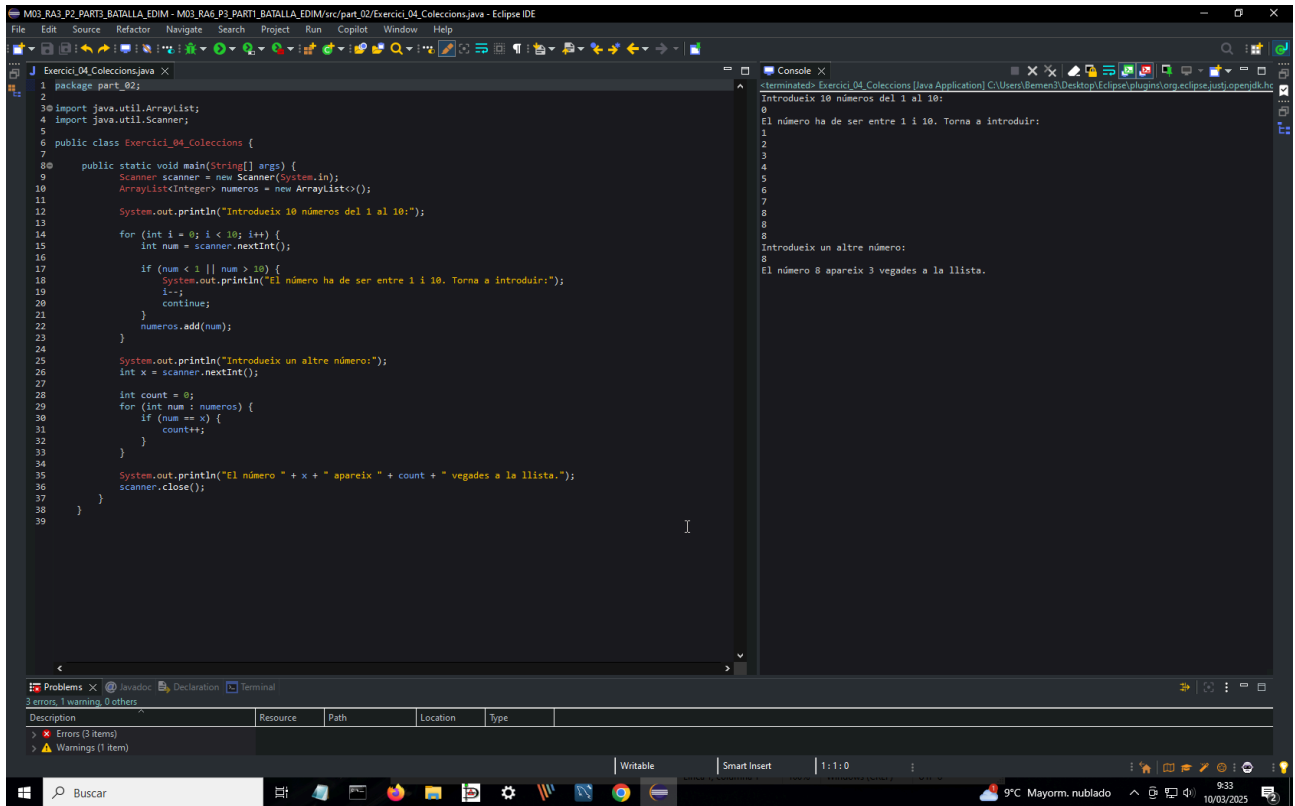
```
System.out.println("Resultat final:");  
System.out.println("Jugador1: " + puntsJugador1 + " punts");  
System.out.println("Jugador2: " + puntsJugador2 + " punts");
```

I declarem un vençador comparant-los.

```
if (puntsJugador1 > puntsJugador2) {  
    System.out.println("Guanya Jugador1!");  
} else if (puntsJugador1 < puntsJugador2) {  
    System.out.println("Guanya Jugador2!");  
} else {  
    System.out.println("És un empat!");  
}  
}  
}
```

Exercici 4

Demana a l'usuari que introdueixi 10 números del 1 al 10. Llavors demana a l'usuari que introdueixi un altre número. Fes un programa que digui quantes vegades apareix aquest número a l'arraylist de números.



The screenshot shows the Eclipse IDE with a Java file named `Exercici_04_Coleccions.java`. The code is as follows:

```
1 package part_02;
2
3 import java.util.ArrayList;
4 import java.util.Scanner;
5
6 public class Exercici_04_Coleccions {
7
8     public static void main(String[] args) {
9         Scanner scanner = new Scanner(System.in);
10        ArrayList<Integer> numeros = new ArrayList<>();
11
12        System.out.println("Introdueix 10 números del 1 al 10:");
13
14        for (int i = 0; i < 10; i++) {
15            int num = scanner.nextInt();
16
17            if (num < 1 || num > 10) {
18                System.out.println("El número ha de ser entre 1 i 10. Torna a introduir:");
19                i--;
20                continue;
21            }
22            numeros.add(num);
23        }
24
25        System.out.println("Introdueix un altre número:");
26        int x = scanner.nextInt();
27
28        int count = 0;
29        for (int num : numeros) {
30            if (num == x) {
31                count++;
32            }
33        }
34
35        System.out.println("El número " + x + " apareix " + count + " vegades a la llista.");
36        scanner.close();
37    }
38 }
39
```

The console output shows the program's execution:

```
<terminated> Exercici_04_Coleccions [Java Application] C:\Users\Beren\Desktop\Eclipse\plugins\org.eclipse.jdt.ui.openide.h...
Introdueix 10 números del 1 al 10:
0
El número ha de ser entre 1 i 10. Torna a introduir:
1
2
3
4
5
6
7
8
8
Introdueix un altre número:
8
El número 8 apareix 3 vegades a la llista.
```

The bottom of the screenshot shows the 'Problems' view with 2 errors and 1 warning, and the 'Terminal' view.

En el mètode main declarem un array de integers que l'usuari ha d'omplir amb 10 números del 1 al 10.

```
public static void main(String[] args) {  
Scanner scanner = new Scanner(System.in);  
ArrayList<Integer> numeros = new ArrayList<>();
```

Amb un bucle for anem demanant números al usuari fins que s'introdueixen 10 valors. Si algun dels números introduïts no està entre l'1 y el 10 informem que el número no es correcte.

```
System.out.println("Introdueix 10 números del 1 al 10:");  
for (int i = 0; i < 10; i++) {  
int num = scanner.nextInt();  
if (num < 1 || num > 10) {  
System.out.println("El número ha de ser entre 1 i 10. Torna a  
introduir:");  
i--;  
continue;  
}
```

Afegim els valors al arraylist números i quan hi han 10 valors en demanem un mes.

```
numeros.add(num);  
}  
System.out.println("Introdueix un altre número:");
```

Guardem aquest últim valor en una variable i recorrem el arraylist per buscar coincidències, iniciem un comptador i cada cop que hi ha coincidència augmentem amb un punt el comptador.

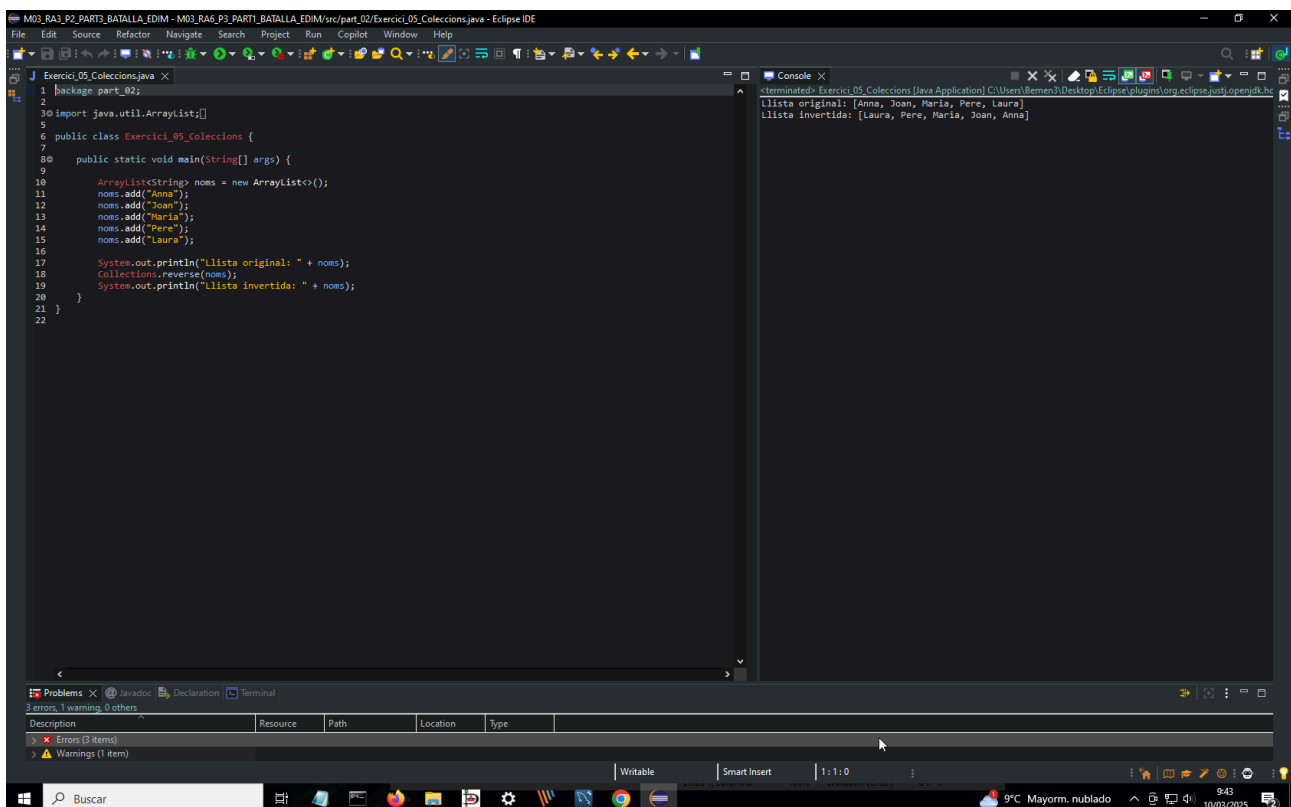
```
int x = scanner.nextInt();
int count = 0;
for (int num : numeros) {
    if (num == x) {
        count++;
    }
}
```

Per acabar imprimim el resultat per pantalla.

```
System.out.println("El número " + x + " apareix " + count + " vegades a la llista.");
scanner.close();
}
```

Exercici 5

Crea una llista amb diferents noms. Fes un programa que inverteixi l'ordre dels noms i els mostri per pantalla.



En aquest últim exercici de la pràctica declarem un arraylist de Strings anomenat noms al mètode main.

```
public static void main(String[] args) {  
ArrayList<String> noms = new ArrayList<>();
```

Afegim cinc noms al arraylist.

```
noms.add("Anna");  
noms.add("Joan");  
noms.add("Maria");  
noms.add("Pere");  
noms.add("Laura");
```

Imprimim la llista en l'ordre original, tal i com hem introduït els usuaris.

```
System.out.println("Llista original: " + noms);
```

Amb el mètode Collections.reverse invertim l'ordre dels valors del arraylist i després ho imprimim per pantalla.

```
Collections.reverse(noms);  
System.out.println("Llista invertida: " + noms);  
}  
}
```