

Selected files

2 printable files

src\arraysList\Clase_Tienda.java
src\arraysList\Articulo.java

src\arraysList\Clase_Tienda.java

```
1  package arraysList;
2
3  import java.util.ArrayList;
4  import java.util.Scanner;
5
6
7
8  class Clase_Tienda {
9
10
11     //PROGRAMA EJECUTABLE
12
13     public static void main(String[] args) {
14
15         Clase_Tienda tienda= new Clase_Tienda();
16         tienda.mostrarMenu();
17
18
19
20     }
21
22
23
24
25
26     // ArrayList para almacenar todos los articulos en el inventario
27     private ArrayList<Articulo> inventario;
28
29     // constructor de la clase Tienda
30     public Clase_Tienda() {
31         this.inventario = new ArrayList<>();
32     }
33
34     // Método principal para mostrar el menú y gestionar la interaccion con el
35     // usuario
36     public void mostrarMenu() {
37         Scanner entrada = new Scanner(System.in);
38
39         int opcion;
40
41         do {
42             System.out.println("\nMenú Principal: ");
43             System.out.println("1. Mostrar artículos.");
44             System.out.println("2. Venta al cliente.");
45             System.out.println("3. Comprar a proveedor.");
46             System.out.println("4. Gestionar artículos.");
47             System.out.println("5. SALIR.");
48             System.out.println("Seleccione una opción: ");
49
```

```
50         opcion = entrada.nextInt();
51
52         switch (opcion) {
53             case 1:
54                 mostrarArticulos();
55                 break;
56             case 2:
57                 realizarVenta();
58                 break;
59             case 3:
60                 realizarCompra();
61                 break;
62             case 4:
63                 gestionarArticulos();
64                 break;
65             case 5:
66                 System.out.println("Saliendo del programa...");
67                 break;
68             default:
69                 System.err.println("ERROR: opción no válida. Intente de nuevo.");
70                 break;
71         }
72     } while (opcion != 5);
73
74 }
75
76 // Método para mostrar toda la informacion en el inventario
77 private void mostrarArticulos() {
78     System.out.println("\nInventario: ");
79     for (Articulo articulo : inventario) {
80         System.out.println(articulo.obtenerInformacion());
81     }
82 }
83
84 // Método para realizar la venta de un artículo
85 private void realizarVenta() {
86     Scanner entrada = new Scanner(System.in);
87     mostrarArticulos();
88
89     System.out.print("Ingrese ID del artículo que desea vender: ");
90     int idArticulo = entrada.nextInt();
91
92     Articulo articulo = buscarArticulo(idArticulo);
93
94     if (articulo != null) {
95         System.out.println("Ingrese la cantidad a vender: ");
96
97         int cantidad = entrada.nextInt();
98
99         if (articulo.vender(cantidad)) {
100             System.out.println("Venta realizada con Éxito.");
101         }
102     }
103 }
104
105 }
106
107 // Método para realizar la compra de un artículo
108 private void realizarCompra() {
109     Scanner entrada = new Scanner(System.in);
```

```
110
111     mostrarArticulos();
112
113     System.out.println("Ingrese el ID del artículo que desea comprar: ");
114
115     int idArticulo = entrada.nextInt();
116
117     Articulo articulo = buscarArticulo(idArticulo);
118
119     if (articulo != null) {
120         System.out.println("Ingrese la cantidad a comprar: ");
121
122         int cantidad = entrada.nextInt();
123         if (articulo.comprar(cantidad)) {
124             System.out.println("Compra realizada con éxito.");
125         }
126     }
127 }
128
129 }
130
131 // Método para gestionar artículos (añadir, editar, eliminar)
132 private void gestionarArticulos() {
133     Scanner entrada = new Scanner(System.in);
134     int opcion;
135
136     do {
137         System.out.println("\nGestión de Artículos: ");
138         System.out.println("1. Añadir artículo.");
139         System.out.println("2. Editar artículo.");
140         System.out.println("3. Eliminar artículo.");
141         System.out.println("4. Volver.");
142         System.out.println("Seleccione una opción: ");
143
144         opcion = entrada.nextInt();
145
146         switch (opcion) {
147             case 1:
148                 añadirArticulo();
149                 break;
150             case 2:
151                 editarArticulo();
152                 break;
153             case 3:
154                 eliminarArticulo();
155                 break;
156             case 4:
157                 System.out.println("Volviendo al menú principal...");
158
159             default:
160                 System.err.println("ERROR: opcion no válida intente de nuevo.");
161                 break;
162         }
163     } while (opcion != 4);
164 }
165
166
167 }
168
169 // Método para añadir un nuevo artículo
```

```
170     private void añadirArticulo() {
171
172         Scanner entrada = new Scanner(System.in);
173
174         System.out.println("Ingrese el nombre del artículo: ");
175         String nombre = entrada.nextLine();
176
177         System.out.println("Ingrese precio de venta: ");
178         double precioVenta = entrada.nextDouble();
179
180         System.out.println("Ingrese precio compra: ");
181         double precioCompra = entrada.nextDouble();
182
183         System.out.println("Ingrece stock inicial: ");
184         int stock = entrada.nextInt();
185
186
187         Articulo nuevoArticulo = new Articulo(nombre, precioVenta, precioCompra, stock);
188
189         inventario.add(nuevoArticulo);
190
191         System.out.println("Artículo añadido con éxito.");
192
193     }
194
195     // Método para editar un artículo existente en el inventario
196     private void editarArticulo() {
197
198         mostrarArticulos();
199
200         System.out.println("Ingrese el ID del artículo que desea editar: ");
201
202         Scanner entrada = new Scanner(System.in);
203         int idArticulo = entrada.nextInt();
204
205         Articulo articulo = buscarArticulo(idArticulo);
206
207         if (articulo != null) {
208             System.out.println("Ingrese el nuevo precio de venta: ");
209
210             double nuevoPrecioVneta = entrada.nextDouble();
211             articulo.setPrecioVenta(nuevoPrecioVneta);
212
213             System.out.println("Ingrese el nuevo precio de compra: ");
214
215             double nuevoPrecioCompra = entrada.nextDouble();
216             articulo.setPrecioCompra(nuevoPrecioCompra);
217
218             System.out.println("Artículo editado con éxito.");
219         }
220
221     }
222
223     // Método para eliminar un artículo del inventario
224     private void eliminarArticulo() {
225         Scanner entrada = new Scanner(System.in);
226         mostrarArticulos();
227
228         System.out.println("Ingrese el ID del artículo que desea eliminar: ");
229         int idArticulo = entrada.nextInt();
```

```
230
231
232     Artículo articulo = buscarArticulo(idArticulo);
233
234     if (articulo != null) {
235         // elimina el artículo del inventario
236         inventario.remove(articulo);
237         System.out.println("Artículo eliminado con éxito.");
238     }
239
240 }
241
242 // Método para buscar un artículo en el inventario por su ID
243 private Artículo buscarArticulo(int id) {
244     for (Articulo articulo : inventario) {
245         if (articulo.getId() == id) {
246             return articulo;
247         }
248     }
249     // Muestra un mensaje de error si no se encuentra el artículo
250     System.err.println("ERROR: No se encontró el artículo con ID " + id);
251     return null;
252 }
253
254 }
255
```

src\arraysList\Articulo.java

```
1 package arraysList;
2
3 public class Articulo {
4
5     // contador estatico para asignar identificadores únicos a los artículos
6     private static int contadorArticulos = 0;
7
8     // Atributos de la clase Articulo
9     private int id;
10    private String nombre;
11    private double precioVenta;
12    private double precioCompra;
13    private final double iva;
14    private int stock;
15
16    // constructor de la clase articulo
17    public Articulo(String nombre, double precioVenta, double precioCompra, int stok) {
18        // asigno un identificador unico a cada articulo
19        this.id = ++contadorArticulos;
20        this.nombre = nombre;
21        this.precioVenta = precioVenta;
22        this.precioCompra = precioCompra;
23        this.iva = 0.21;
24        this.stock = stok;
25
26    }
27
28    // metodos getters y setters para acceder y modificar los atributos
29    public int getId() {
30        return id;
31    }
32
33    public String getNombre() {
34        return nombre;
35    }
36
37    public double getPrecioVenta() {
38        return precioVenta;
39    }
40
41    public double getPrecioCompra() {
42        return precioCompra;
43    }
44
45    public final double getIva() {
46        return iva;
47    }
48
49    public int getStock() {
50        return stock;
51    }
52
53    public void setId(int id) {
54        this.id = id;
55    }
56
57    public void setNombre(String nombre) {
58        this.nombre = nombre;
59    }
60
61    public void setPrecioVenta(double precioVenta) {
62        this.precioVenta = precioVenta;
63    }
64
65    public void setPrecioCompra(double precioCompra) {
66        this.precioCompra = precioCompra;
67    }
68
69    public void setIva(double iva) {
70        this.iva = iva;
71    }
72
73    public void setStock(int stock) {
74        this.stock = stock;
75    }
76
77    public void mostrarDatos() {
78        System.out.println("Artículo: " + nombre + ", ID: " + id + ", Precio Venta: " + precioVenta + ", Precio Compra: " + precioCompra + ", IVA: " + iva + ", Stock: " + stock);
79    }
80
81    public void actualizarPrecio(double porcentaje) {
82        precioVenta = precioVenta * (1 + porcentaje);
83        precioCompra = precioCompra * (1 + porcentaje);
84    }
85
86    public void eliminar() {
87        // Eliminar el artículo del inventario
88        // ...
89    }
90
91    public void agregar() {
92        // Agregar el artículo al inventario
93        // ...
94    }
95
96    public void mostrarInventario() {
97        // Mostrar el inventario
98        // ...
99    }
100
101    public void calcularGanancia() {
102        // Calcular la ganancia
103        // ...
104    }
105
106    public void calcularPerdida() {
107        // Calcular la pérdida
108        // ...
109    }
110
111    public void calcularPromedio() {
112        // Calcular el promedio
113        // ...
114    }
115
116    public void calcularDesviacion() {
117        // Calcular la desviación
118        // ...
119    }
120
121    public void calcularCorrelacion() {
122        // Calcular la correlación
123        // ...
124    }
125
126    public void calcularRegresion() {
127        // Calcular la regresión
128        // ...
129    }
130
131    public void calcularMatriz() {
132        // Calcular la matriz
133        // ...
134    }
135
136    public void calcularDeterminante() {
137        // Calcular el determinante
138        // ...
139    }
140
141    public void calcularInversa() {
142        // Calcular la inversa
143        // ...
144    }
145
146    public void calcularRango() {
147        // Calcular el rango
148        // ...
149    }
150
151    public void calcularNucleo() {
152        // Calcular el núcleo
153        // ...
154    }
155
156    public void calcularImagen() {
157        // Calcular la imagen
158        // ...
159    }
160
161    public void calcularKer() {
162        // Calcular el kernel
163        // ...
164    }
165
166    public void calcularFiltro() {
167        // Calcular el filtro
168        // ...
169    }
170
171    public void calcularTransformada() {
172        // Calcular la transformada
173        // ...
174    }
175
176    public void calcularDerivada() {
177        // Calcular la derivada
178        // ...
179    }
180
181    public void calcularIntegral() {
182        // Calcular la integral
183        // ...
184    }
185
186    public void calcularArea() {
187        // Calcular el área
188        // ...
189    }
190
191    public void calcularPerimetro() {
192        // Calcular el perímetro
193        // ...
194    }
195
196    public void calcularVolumen() {
197        // Calcular el volumen
198        // ...
199    }
200
201    public void calcularSuperficie() {
202        // Calcular la superficie
203        // ...
204    }
205
206    public void calcularLongitud() {
207        // Calcular la longitud
208        // ...
209    }
210
211    public void calcularAnchura() {
212        // Calcular la anchura
213        // ...
214    }
215
216    public void calcularAlto() {
217        // Calcular el alto
218        // ...
219    }
220
221    public void calcularPeso() {
222        // Calcular el peso
223        // ...
224    }
225
226    public void calcularDensidad() {
227        // Calcular la densidad
228        // ...
229    }
230
231    public void calcularVelocidad() {
232        // Calcular la velocidad
233        // ...
234    }
235
236    public void calcularAceleracion() {
237        // Calcular la aceleración
238        // ...
239    }
240
241    public void calcularFuerza() {
242        // Calcular la fuerza
243        // ...
244    }
245
246    public void calcularEnergia() {
247        // Calcular la energía
248        // ...
249    }
250
251    public void calcularPotencia() {
252        // Calcular la potencia
253        // ...
254    }
255
256    public void calcularTrabajo() {
257        // Calcular el trabajo
258        // ...
259    }
260
261    public void calcularMomento() {
262        // Calcular el momento
263        // ...
264    }
265
266    public void calcularImpulso() {
267        // Calcular el impulso
268        // ...
269    }
270
271    public void calcularPresion() {
272        // Calcular la presión
273        // ...
274    }
275
276    public void calcularTemperatura() {
277        // Calcular la temperatura
278        // ...
279    }
280
281    public void calcularPresion() {
282        // Calcular la presión
283        // ...
284    }
285
286    public void calcularTemperatura() {
287        // Calcular la temperatura
288        // ...
289    }
290
291    public void calcularPresion() {
292        // Calcular la presión
293        // ...
294    }
295
296    public void calcularTemperatura() {
297        // Calcular la temperatura
298        // ...
299    }
300
301    public void calcularPresion() {
302        // Calcular la presión
303        // ...
304    }
305
306    public void calcularTemperatura() {
307        // Calcular la temperatura
308        // ...
309    }
310
311    public void calcularPresion() {
312        // Calcular la presión
313        // ...
314    }
315
316    public void calcularTemperatura() {
317        // Calcular la temperatura
318        // ...
319    }
320
321    public void calcularPresion() {
322        // Calcular la presión
323        // ...
324    }
325
326    public void calcularTemperatura() {
327        // Calcular la temperatura
328        // ...
329    }
330
331    public void calcularPresion() {
332        // Calcular la presión
333        // ...
334    }
335
336    public void calcularTemperatura() {
337        // Calcular la temperatura
338        // ...
339    }
340
341    public void calcularPresion() {
342        // Calcular la presión
343        // ...
344    }
345
346    public void calcularTemperatura() {
347        // Calcular la temperatura
348        // ...
349    }
350
351    public void calcularPresion() {
352        // Calcular la presión
353        // ...
354    }
355
356    public void calcularTemperatura() {
357        // Calcular la temperatura
358        // ...
359    }
360
361    public void calcularPresion() {
362        // Calcular la presión
363        // ...
364    }
365
366    public void calcularTemperatura() {
367        // Calcular la temperatura
368        // ...
369    }
370
371    public void calcularPresion() {
372        // Calcular la presión
373        // ...
374    }
375
376    public void calcularTemperatura() {
377        // Calcular la temperatura
378        // ...
379    }
380
381    public void calcularPresion() {
382        // Calcular la presión
383        // ...
384    }
385
386    public void calcularTemperatura() {
387        // Calcular la temperatura
388        // ...
389    }
390
391    public void calcularPresion() {
392        // Calcular la presión
393        // ...
394    }
395
396    public void calcularTemperatura() {
397        // Calcular la temperatura
398        // ...
399    }
400
401    public void calcularPresion() {
402        // Calcular la presión
403        // ...
404    }
405
406    public void calcularTemperatura() {
407        // Calcular la temperatura
408        // ...
409    }
410
411    public void calcularPresion() {
412        // Calcular la presión
413        // ...
414    }
415
416    public void calcularTemperatura() {
417        // Calcular la temperatura
418        // ...
419    }
420
421    public void calcularPresion() {
422        // Calcular la presión
423        // ...
424    }
425
426    public void calcularTemperatura() {
427        // Calcular la temperatura
428        // ...
429    }
430
431    public void calcularPresion() {
432        // Calcular la presión
433        // ...
434    }
435
436    public void calcularTemperatura() {
437        // Calcular la temperatura
438        // ...
439    }
440
441    public void calcularPresion() {
442        // Calcular la presión
443        // ...
444    }
445
446    public void calcularTemperatura() {
447        // Calcular la temperatura
448        // ...
449    }
450
451    public void calcularPresion() {
452        // Calcular la presión
453        // ...
454    }
455
456    public void calcularTemperatura() {
457        // Calcular la temperatura
458        // ...
459    }
460
461    public void calcularPresion() {
462        // Calcular la presión
463        // ...
464    }
465
466    public void calcularTemperatura() {
467        // Calcular la temperatura
468        // ...
469    }
470
471    public void calcularPresion() {
472        // Calcular la presión
473        // ...
474    }
475
476    public void calcularTemperatura() {
477        // Calcular la temperatura
478        // ...
479    }
480
481    public void calcularPresion() {
482        // Calcular la presión
483        // ...
484    }
485
486    public void calcularTemperatura() {
487        // Calcular la temperatura
488        // ...
489    }
490
491    public void calcularPresion() {
492        // Calcular la presión
493        // ...
494    }
495
496    public void calcularTemperatura() {
497        // Calcular la temperatura
498        // ...
499    }
500
501    public void calcularPresion() {
502        // Calcular la presión
503        // ...
504    }
505
506    public void calcularTemperatura() {
507        // Calcular la temperatura
508        // ...
509    }
510
511    public void calcularPresion() {
512        // Calcular la presión
513        // ...
514    }
515
516    public void calcularTemperatura() {
517        // Calcular la temperatura
518        // ...
519    }
520
521    public void calcularPresion() {
522        // Calcular la presión
523        // ...
524    }
525
526    public void calcularTemperatura() {
527        // Calcular la temperatura
528        // ...
529    }
530
531    public void calcularPresion() {
532        // Calcular la presión
533        // ...
534    }
535
536    public void calcularTemperatura() {
537        // Calcular la temperatura
538        // ...
539    }
540
541    public void calcularPresion() {
542        // Calcular la presión
543        // ...
544    }
545
546    public void calcularTemperatura() {
547        // Calcular la temperatura
548        // ...
549    }
550
551    public void calcularPresion() {
552        // Calcular la presión
553        // ...
554    }
555
556    public void calcularTemperatura() {
557        // Calcular la temperatura
558        // ...
559    }
560
561    public void calcularPresion() {
562        // Calcular la presión
563        // ...
564    }
565
566    public void calcularTemperatura() {
567        // Calcular la temperatura
568        // ...
569    }
570
571    public void calcularPresion() {
572        // Calcular la presión
573        // ...
574    }
575
576    public void calcularTemperatura() {
577        // Calcular la temperatura
578        // ...
579    }
580
581    public void calcularPresion() {
582        // Calcular la presión
583        // ...
584    }
585
586    public void calcularTemperatura() {
587        // Calcular la temperatura
588        // ...
589    }
590
591    public void calcularPresion() {
592        // Calcular la presión
593        // ...
594    }
595
596    public void calcularTemperatura() {
597        // Calcular la temperatura
598        // ...
599    }
600
601    public void calcularPresion() {
602        // Calcular la presión
603        // ...
604    }
605
606    public void calcularTemperatura() {
607        // Calcular la temperatura
608        // ...
609    }
610
611    public void calcularPresion() {
612        // Calcular la presión
613        // ...
614    }
615
616    public void calcularTemperatura() {
617        // Calcular la temperatura
618        // ...
619    }
620
621    public void calcularPresion() {
622        // Calcular la presión
623        // ...
624    }
625
626    public void calcularTemperatura() {
627        // Calcular la temperatura
628        // ...
629    }
630
631    public void calcularPresion() {
632        // Calcular la presión
633        // ...
634    }
635
636    public void calcularTemperatura() {
637        // Calcular la temperatura
638        // ...
639    }
640
641    public void calcularPresion() {
642        // Calcular la presión
643        // ...
644    }
645
646    public void calcularTemperatura() {
647        // Calcular la temperatura
648        // ...
649    }
650
651    public void calcularPresion() {
652        // Calcular la presión
653        // ...
654    }
655
656    public void calcularTemperatura() {
657        // Calcular la temperatura
658        // ...
659    }
660
661    public void calcularPresion() {
662        // Calcular la presión
663        // ...
664    }
665
666    public void calcularTemperatura() {
667        // Calcular la temperatura
668        // ...
669    }
670
671    public void calcularPresion() {
672        // Calcular la presión
673        // ...
674    }
675
676    public void calcularTemperatura() {
677        // Calcular la temperatura
678        // ...
679    }
680
681    public void calcularPresion() {
682        // Calcular la presión
683        // ...
684    }
685
686    public void calcularTemperatura() {
687        // Calcular la temperatura
688        // ...
689    }
690
691    public void calcularPresion() {
692        // Calcular la presión
693        // ...
694    }
695
696    public void calcularTemperatura() {
697        // Calcular la temperatura
698        // ...
699    }
700
701    public void calcularPresion() {
702        // Calcular la presión
703        // ...
704    }
705
706    public void calcularTemperatura() {
707        // Calcular la temperatura
708        // ...
709    }
710
711    public void calcularPresion() {
712        // Calcular la presión
713        // ...
714    }
715
716    public void calcularTemperatura() {
717        // Calcular la temperatura
718        // ...
719    }
720
721    public void calcularPresion() {
722        // Calcular la presión
723        // ...
724    }
725
726    public void calcularTemperatura() {
727        // Calcular la temperatura
728        // ...
729    }
730
731    public void calcularPresion() {
732        // Calcular la presión
733        // ...
734    }
735
736    public void calcularTemperatura() {
737        // Calcular la temperatura
738        // ...
739    }
740
741    public void calcularPresion() {
742        // Calcular la presión
743        // ...
744    }
745
746    public void calcularTemperatura() {
747        // Calcular la temperatura
748        // ...
749    }
750
751    public void calcularPresion() {
752        // Calcular la presión
753        // ...
754    }
755
756    public void calcularTemperatura() {
757        // Calcular la temperatura
758        // ...
759    }
760
761    public void calcularPresion() {
762        // Calcular la presión
763        // ...
764    }
765
766    public void calcularTemperatura() {
767        // Calcular la temperatura
768        // ...
769    }
770
771    public void calcularPresion() {
772        // Calcular la presión
773        // ...
774    }
775
776    public void calcularTemperatura() {
777        // Calcular la temperatura
778        // ...
779    }
780
781    public void calcularPresion() {
782        // Calcular la presión
783        // ...
784    }
785
786    public void calcularTemperatura() {
787        // Calcular la temperatura
788        // ...
789    }
790
791    public void calcularPresion() {
792        // Calcular la presión
793        // ...
794    }
795
796    public void calcularTemperatura() {
797        // Calcular la temperatura
798        // ...
799    }
800
801    public void calcularPresion() {
802        // Calcular la presión
803        // ...
804    }
805
806    public void calcularTemperatura() {
807        // Calcular la temperatura
808        // ...
809    }
810
811    public void calcularPresion() {
812        // Calcular la presión
813        // ...
814    }
815
816    public void calcularTemperatura() {
817        // Calcular la temperatura
818        // ...
819    }
820
821    public void calcularPresion() {
822        // Calcular la presión
823        // ...
824    }
825
826    public void calcularTemperatura() {
827        // Calcular la temperatura
828        // ...
829    }
830
831    public void calcularPresion() {
832        // Calcular la presión
833        // ...
834    }
835
836    public void calcularTemperatura() {
837        // Calcular la temperatura
838        // ...
839    }
840
841    public void calcularPresion() {
842        // Calcular la presión
843        // ...
844    }
845
846    public void calcularTemperatura() {
847        // Calcular la temperatura
848        // ...
849    }
850
851    public void calcularPresion() {
852        // Calcular la presión
853        // ...
854    }
855
856    public void calcularTemperatura() {
857        // Calcular la temperatura
858        // ...
859    }
860
861    public void calcularPresion() {
862        // Calcular la presión
863        // ...
864    }
865
866    public void calcularTemperatura() {
867        // Calcular la temperatura
868        // ...
869    }
870
871    public void calcularPresion() {
872        // Calcular la presión
873        // ...
874    }
875
876    public void calcularTemperatura() {
877        // Calcular la temperatura
878        // ...
879    }
880
881    public void calcularPresion() {
882        // Calcular la presión
883        // ...
884    }
885
886    public void calcularTemperatura() {
887        // Calcular la temperatura
888        // ...
889    }
890
891    public void calcularPresion() {
892        // Calcular la presión
893        // ...
894    }
895
896    public void calcularTemperatura() {
897        // Calcular la temperatura
898        // ...
899    }
900
901    public void calcularPresion() {
902        // Calcular la presión
903        // ...
904    }
905
906    public void calcularTemperatura() {
907        // Calcular la temperatura
908        // ...
909    }
910
911    public void calcularPresion() {
912        // Calcular la presión
913        // ...
914    }
915
916    public void calcularTemperatura() {
917        // Calcular la temperatura
918        // ...
919    }
920
921    public void calcularPresion() {
922        // Calcular la presión
923        // ...
924    }
925
926    public void calcularTemperatura() {
927        // Calcular la temperatura
928        // ...
929    }
930
931    public void calcularPresion() {
932        // Calcular la presión
933        // ...
934    }
935
936    public void calcularTemperatura() {
937        // Calcular la temperatura
938        // ...
939    }
940
941    public void calcularPresion() {
942        // Calcular la presión
943        // ...
944    }
945
946    public void calcularTemperatura() {
947        // Calcular la temperatura
948        // ...
949    }
950
951    public void calcularPresion() {
952        // Calcular la presión
953        // ...
954    }
955
956    public void calcularTemperatura() {
957        // Calcular la temperatura
958        // ...
959    }
960
961    public void calcularPresion() {
962        // Calcular la presión
963        // ...
964    }
965
966    public void calcularTemperatura() {
967        // Calcular la temperatura
968        // ...
969    }
970
971    public void calcularPresion() {
972        // Calcular la presión
973        // ...
974    }
975
976    public void calcularTemperatura() {
977        // Calcular la temperatura
978        // ...
979    }
980
981    public void calcularPresion() {
982        // Calcular la presión
983        // ...
984    }
985
986    public void calcularTemperatura() {
987        // Calcular la temperatura
988        // ...
989    }
990
991    public void calcularPresion() {
992        // Calcular la presión
993        // ...
994    }
995
996    public void calcularTemperatura() {
997        // Calcular la temperatura
998        // ...
999    }
1000
1001    public void calcularPresion() {
1002        // Calcular la presión
1003        // ...
1004    }
1005
1006    public void calcularTemperatura() {
1007        // Calcular la temperatura
1008        // ...
1009    }
1010
1011    public void calcularPresion() {
1012        // Calcular la presión
1013        // ...
1014    }
1015
1016    public void calcularTemperatura() {
1017        // Calcular la temperatura
1018        // ...
1019    }
1020
1021    public void calcularPresion() {
1022        // Calcular la presión
1023        // ...
1024    }
1025
1026    public void calcularTemperatura() {
1027        // Calcular la temperatura
1028        // ...
1029    }
1030
1031    public void calcularPresion() {
1032        // Calcular la presión
1033        // ...
1034    }
1035
1036    public void calcularTemperatura() {
1037        // Calcular la temperatura
1038        // ...
1039    }
1040
1041    public void calcularPresion() {
1042        // Calcular la presión
1043        // ...
1044    }
1045
1046    public void calcularTemperatura() {
1047        // Calcular la temperatura
1048        // ...
1049    }
1050
1051    public void calcularPresion() {
1052        // Calcular la presión
1053        // ...
1054    }
1055
1056    public void calcularTemperatura() {
1057        // Calcular la temperatura
1058        // ...
1059    }
1060
1061    public void calcularPresion() {
1062        // Calcular la presión
1063        // ...
1064    }
1065
1066    public void calcularTemperatura() {
1067        // Calcular la temperatura
1068        // ...
1069    }
1070
1071    public void calcularPresion() {
1072        // Calcular la presión
1073        // ...
1074    }
1075
1076    public void calcularTemperatura() {
1077        // Calcular la temperatura
1078        // ...
1079    }
1080
1081    public void calcularPresion() {
1082        // Calcular la presión
1083        // ...
1084    }
1085
1086    public void calcularTemperatura() {
1087        // Calcular la temperatura
1088        // ...
1089    }
1090
1091    public void calcularPresion() {
1092        // Calcular la presión
1093        // ...
1094    }
1095
1096    public void calcularTemperatura() {
1097        // Calcular la temperatura
1098        // ...
1099    }
1100
1101    public void calcularPresion() {
1102        // Calcular la presión
1103        // ...
1104    }
1105
1106    public void calcularTemperatura() {
1107        // Calcular la temperatura
1108        // ...
1109    }
1110
1111    public void calcularPresion() {
1112        // Calcular la presión
1113        // ...
1114    }
1115
1116    public void calcularTemperatura() {
1117        // Calcular la temperatura
1118        // ...
1119    }
1120
1121    public void calcularPresion() {
1122        // Calcular la presión
1123        // ...
1124    }
1125
1126    public void calcularTemperatura() {
1127        // Calcular la temperatura
1128        // ...
1129    }
1130
1131    public void calcularPresion() {
1132        // Calcular la presión
1133        // ...
1134    }
1135
1136    public void calcularTemperatura() {
1137        // Calcular la temperatura
1138        // ...
1139    }
1140
1141    public void calcularPresion() {
1142        // Calcular la presión
1143        // ...
1144    }
1145
1146    public void calcularTemperatura() {
1147        // Calcular la temperatura
1148        // ...
1149    }
1150
1151    public void calcularPresion() {
1152        // Calcular la presión
1153        // ...
1154    }
1155
1156    public void calcularTemperatura() {
1157        // Calcular la temperatura
1158        // ...
1159    }
1160
1161    public void calcularPresion() {
1162        // Calcular la presión
1163        // ...
1164    }
1165
1166    public void calcularTemperatura() {
1167        // Calcular la temperatura
1168        // ...
1169    }
1170
1171    public void calcularPresion() {
1172        // Calcular la presión
1173        // ...
1174    }
1175
1176    public void calcularTemperatura() {
1177        // Calcular la temperatura
1178        // ...
1179    }
1180
1181    public void calcularPresion() {
1182        // Calcular la presión
1183        // ...
1184    }
1185
1186    public void calcularTemperatura() {
1187        // Calcular la temperatura
1188        // ...
1189    }
1190
1191    public void calcularPresion() {
1192        // Calcular la presión
1193        // ...
1194    }
1195
1196    public void calcularTemperatura() {
1197        // Calcular la temperatura
1198        // ...
1199    }
1200
1201    public void calcularPresion() {
1202        // Calcular la presión
1203        // ...
1204    }
1205
1206    public void calcularTemperatura() {
1207        // Calcular la temperatura
1208        // ...
1209    }
1210
1211    public void calcularPresion() {
1212        // Calcular la presión
1213        // ...
1214    }
1215
1216    public void calcularTemperatura() {
1217        // Calcular la temperatura
1218        // ...
1219    }
1220
1221    public void calcularPresion() {
1222        // Calcular la presión
1223        // ...
1224    }
1225
1226    public void calcularTemperatura() {
1227        // Calcular la temperatura
1228        // ...
1229    }
1230
1231    public void calcularPresion() {
1232        // Calcular la presión
1233        // ...
1234    }
1235
1236    public void calcularTemperatura() {
1237        // Calcular la temperatura
1238        // ...
1239    }
1240
1241    public void calcularPresion() {
1242        // Calcular la presión
1243        // ...
1244    }
1245
1246    public void calcularTemperatura() {
1247        // Calcular la temperatura
1248        // ...
1249    }
1250
1251    public void calcularPresion() {
1252        // Calcular la presión
1253        // ...
1254    }
1255
1256    public void calcularTemperatura() {
1257        // Calcular la temperatura
1258        // ...
1259    }
1260
1261    public void calcularPresion() {
1262        // Calcular la presión
1263        // ...
1264    }
1265
1266    public void calcularTemperatura() {
1267        // Calcular la temperatura
1268        // ...
1269    }
1270
1271    public void calcularPresion() {
1272        // Calcular la presión
1273        // ...
1274    }
1275
1276    public void calcularTemperatura() {
1277        // Calcular la temperatura
1278        // ...
1279    }
1280
1281    public void calcularPresion() {
1282        // Calcular la presión
1283        // ...
1284    }
1285
1286    public void calcularTemperatura() {
1287        // Calcular la temperatura
1288        // ...
1289    }
1290
1291    public void calcularPresion() {
1292        // Calcular la presión
1293        // ...
1294    }
1295
1296    public void calcularTemperatura() {
1297        // Calcular la temperatura
1298        // ...
1299    }
1300
1301    public void calcularPresion() {
1302        // Calcular la presión
1303        // ...
1304    }
1305
1306    public void calcularTemperatura() {
1307        // Calcular la temperatura
1308        // ...
1309    }
1310
1311    public void calcularPresion() {
1312        // Calcular la presión
1313        // ...
1314    }
1315
1316    public void calcularTemperatura() {
1317        // Calcular la temperatura
1318        // ...
1319    }
1320
1321    public void calcularPresion() {
1322        // Calcular la presión
1323        // ...
1324    }
1325
1326    public void calcularTemperatura() {
1327        // Calcular la temperatura
1328        // ...
1329    }
1330
1331    public void calcularPresion() {
1332        // Calcular la presión
1333        // ...
1334    }
1335
1336    public void calcularTemperatura() {
1337        // Calcular la temperatura
1338        // ...
1339    }
1340
1341    public void calcularPresion() {
1342        // Calcular la presión
1343        // ...
1344    }
1345
1346    public void calcularTemperatura() {
1347        // Calcular la temperatura
1348        // ...
1349    }
1350
1351    public void calcularPresion() {
1352        // Calcular la presión
1353        // ...
1354    }
1355
1356    public void calcularTemperatura() {
1357        // Calcular la temperatura
1358        // ...
1359    }
1360
1361    public void calcularPresion() {
1362        // Calcular la presión
1363        // ...
1364    }
1365
1366    public void calcularTemperatura() {
1367        // Calcular la temperatura
1368        // ...
1369    }
1370
1371    public void calcularPresion() {
1372        // Calcular la presión
1373        // ...
1374    }
1375
1376    public void calcularTemperatura() {
1377        // Calcular la temperatura
1378        // ...
1379    }
1380
1381    public void calcularPresion() {
1382        // Calcular la presión
1383        // ...
1384    }
1385
1386    public void calcularTemperatura() {
1387        // Calcular la temperatura
1388        // ...
1389    }
1390
1391    public void calcularPresion() {
1392        // Calcular la presión
1393        // ...
1394    }
1395
1396    public void calcularTemperatura() {
1397        // Calcular la temperatura
1398        // ...
1399    }
1400
1401    public void calcularPresion() {
1402        // Calcular la presión
1403        // ...
1404    }
1405
1406    public void calcularTemperatura() {
1407        // Calcular la temperatura
1408        // ...
1409    }
1410
1411    public void calcularPresion() {
1412        // Calcular la presión
1413        // ...
1414    }
1415
1416    public void calcularTemperatura() {
1417        // Calcular la temperatura
1418        // ...
1419    }
1420
1421    public void calcularPresion() {
1422        // Calcular la presión
1423        // ...
1424    }
1425
1426    public void calcularTemperatura() {
1427        // Calcular la temperatura
1428        // ...
1429    }
1430
1431    public void calcularPresion() {
1432        // Calcular la presión
1433        // ...
1434    }
1435
1436    public void calcularTemperatura() {
1437        // Calcular la temperatura
1438        // ...
1439    }
1440
1441    public void calcularPresion() {
1442        // Calcular la presión
1443        // ...
1444    }
1445
1446    public void calcularTemperatura() {
1447        // Calcular la temperatura
1448        // ...
1449    }
1450
```

```
31     }
32
33     public String getNombre() {
34         return nombre;
35     }
36
37     public double getPrecioVenta() {
38         return precioVenta;
39     }
40
41     public void setPrecioVenta(double precioVenta) {
42         // validacion: precio de venta debe ser mayor que cero
43         if (precioVenta > 0) {
44             this.precioVenta = precioVenta;
45         } else {
46             System.err.println("ERROR: el precio de venta debe ser mayor que cero.");
47         }
48     }
49
50
51     public double getPrecioCompra() {
52         return precioCompra;
53     }
54
55     public void setPrecioCompra(double precioCompra) {
56         // validacion: precio compra debe ser mayor que cero
57         if (precioCompra > 0) {
58             this.precioCompra = precioCompra;
59         } else {
60             System.err.println("ERROR: el precio de compra debe ser mayor que cero. ");
61         }
62     }
63
64
65     public double getIva() {
66         return iva;
67     }
68
69     public int getStock() {
70         return stock;
71     }
72
73     // metodo para realizar una venta de unidades del articulo
74     public boolean vender(int cantidad) {
75         // validacion: verificar si hay suficiente stock para realizar la venta
76         if (cantidad > 0 && cantidad <= stock) {
77             stock -= cantidad; // actualiza el stock despues de la venta
78             return true;
79         } else {
80             System.err.println("ERROR: no hay suficiente stock para realizar la venta.");
81             return false;
82         }
83     }
84
85
86     // Método para realizar una compra de unidades del artículo
87     public boolean comprar(int cantidad) {
88         // validacion: la cantidad a comprar debe ser mayor que cero
89         if (cantidad > 0) {
90             stock += cantidad; // Actualiza el stock después de la compra
```

```
91         return true;
92     } else {
93         System.err.println("ERROR: la cantidad a comprar debe ser mayor que cero. ");
94         return false;
95     }
96 }
97
98 // Método para obtener una cadena con la informacion del artículo
99 public String obtenerInformacion() {
100     return String.format("ID: %d, NOMBRE: %s, PRECIO VENTA: %.2f, PRECIO COMPRA: %.2f,
IVA: %.2f, STOCK: %d",
101         id, nombre, precioVenta, precioCompra, iva, stock);
102 }
103
104 }
105
```