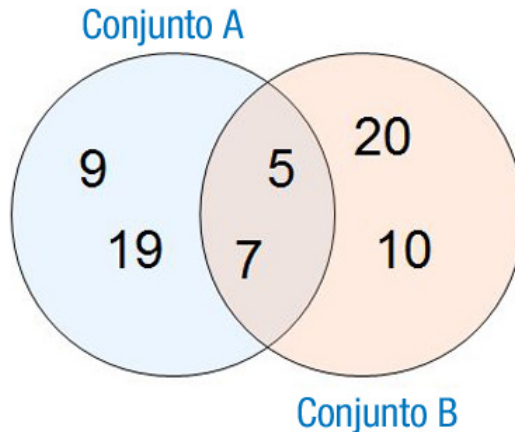


9.C. Conjuntos.

4. Conjuntos (IV).

¿Cómo podría copiar los elementos de un conjunto de uno a otro? ¿Hay que usar un bucle `for` y recorrer toda la lista para ello? ¡Qué va! Para facilitar esta tarea, los conjuntos, y las colecciones en general, facilitan un montón de operaciones para poder combinar los datos de varias colecciones. Ya se vieron en un apartado anterior, [aquí simplemente vamos poner un ejemplo de su uso.](#)

Partimos del siguiente ejemplo, en el que hay dos colecciones de diferente tipo, cada una con 4 números enteros:



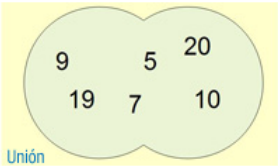
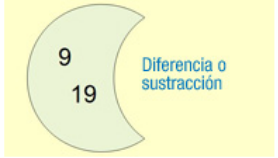
```
TreeSet<Integer> A= new TreeSet<Integer>();
```


```
A.add(9); A.add(19); A.add(5); A.add(7); // Elementos del conjunto A: 9, 19, 5 y 7
```

```
LinkedHashSet<Integer> B= new LinkedHashSet<Integer>();
```

```
B.add(10); B.add(20); B.add(5); B.add(7); // Elementos del conjunto B: 10, 20, 5 y 7
```

En el ejemplo anterior, el literal de número se convierte automáticamente a la clase envoltorio `Integer` sin tener que hacer nada, lo cual es una ventaja. Veamos las formas de combinar ambas colecciones:

Tipos de combinaciones.		
Combinación.	Código.	Elementos finales del conjunto A.
Unión. Añadir todos los elementos del conjunto B en el conjunto A.	<code>A.addAll(B)</code>	Todos los del conjunto A, añadiendo los del B, pero sin repetir los que ya están: 5, 7, 9, 10, 19 y 20. 
Diferencia. Eliminar los elementos del conjunto B que puedan estar en el conjunto A.	<code>A.removeAll(B)</code>	Todos los elementos del conjunto A, que no estén en el conjunto B: 9, 19. 

Tipos de combinaciones.		
Combinación.	Código.	Elementos finales del conjunto A.
Intersección. Retiene los elementos comunes a ambos conjuntos.	<code>A.retainAll(B)</code>	<p>Todos los elementos del conjunto A, que también están en el conjunto B: 5 y 7.</p> 

Recuerda, estas operaciones son comunes a todas las colecciones.

Para saber más

Puede que no recuerdes cómo era eso de los conjuntos, y dada la íntima relación de las colecciones con el álgebra de conjuntos, es recomendable que repases cómo era aquello, con el siguiente artículo de la Wikipedia.

[Álgebra de conjuntos.](#)

Autoevaluación

Tienes un `HashSet` llamado `vocales` que contiene los elementos "a", "e", "i", "o", "u", y otro, llamado `vocales_fuertes` con los elementos "a", "e" y "o". ¿De qué forma podríamos sacar una lista con las denominadas vocales débiles (que son aquellas que no son fuertes)?

- ☐ `vocales.retainAll(vocales_fuertes);`
- ☐ `vocales.removeAll(vocales_fuertes);`
- ☐ No es posible hacer esto con `HashSet`, solo se puede hacer con `TreeSet` o `LinkedHashSet`.