Table 10.1 Basic `ArrayList` Methods

Method	Description	`ArrayList<String>` example
`add(value)`	adds the given value at the end of the list	`list.add("end");`
`add(index, value)`	adds the given value at the given index, shifting subsequent values right	`list.add(1, "middle");`
`clear()`	removes all elements from the list	`list.clear();`
`get(index)`	gets the value at the given index	`list.get(1)`
`remove(index)`	removes the value at the given index, shifting subsequent values left	`list.remove(1);`
`set(index, value)`	replaces the value at the given index with the given value	`list.set(2, "hello");`
`size()`	returns the current number of elements in the list	`list.size()`

Table 11.4 Common Set Operations, Given Two Sets A and B

Set operation	Method	Description
union	`addAll`	Set of all elements that are in A, B, or both
intersection	`retainAll`	Set of all elements that are in both A and B
difference	`removeAll`	Set of all elements that are in A but not in B
superset, subset	`containsAll`	Returns `true` if A is a superset of (contains all elements of) B

Table 11.5 Useful Methods of Maps

Method	Description
`clear()`	Removes all keys and values from a map
`containsKey(key)`	Returns `true` if the given key maps to some value in this map
`containsValue(value)`	Returns `true` if some key maps to the given value in this map
`get(key)`	Returns the value associated with this key, or `null` if not found
`isEmpty()`	Returns `true` if this collection contains no keys or values
`keySet()`	Returns a `Set` of all keys in this map
`put(key, value)`	Associates the given key with the given value
`putAll(map)`	Adds all key/value mappings from the given map to this map
`remove(key)`	Removes the given key and its associated value from this map
`size()`	Returns the number of key/value mappings in this map
`values()`	Returns a `Collection` of all values in this map