AssertJ Iterable `contains…` assertions [Cheat Sheet]

Assertion	Actual must match Expected Order?	Actual can contain Duplicates?	Actual can contain elements other than in Expected?	Examples, in the form of:
				
				`assertThat(actual)`
				`    .contains(expected);`

`contains`	no ⛔	yes ✅	yes ✅	PASSES ✅:
				
				`assertThat(List.of("1", "2", "3"))`
				`    .contains("1");`
				
				`assertThat(List.of("1", "2", "3"))`
				`    .contains("1", "3");`
				
				FAILS ❌:
				
				`// could not find "4"`
				`assertThat(List.of("1", "2", "3"))`
				`    .contains("4");`

`containsOnly`	no ⛔	yes ✅	no ⛔	PASSES ✅:
				
				`assertThat(List.of("1", "2", "3"))`
				`    .containsOnly("3", "2", "1");`
				
				`assertThat(List.of("1", "1", "2", "2", "3", "3"))`
				`    .containsOnly("1", "2", "3");`
				
				FAILS ❌:
				
				`// did not expect "2"`
				`assertThat(List.of("1", "2", "3"))`
				`    .containsOnly("1", "3");`

Created by Ted M. Young (January 9, 2023) https://ted.dev

Assertion	Actual must match Expected Order?	Actual can contain Duplicates?	Actual can contain elements other than in Expected?	Examples, in the form of:
				
				`assertThat(actual)`
				`    .contains(expected);`

`containsOnlyOnce`	no ⛔	no ⛔ not for items being asserted	yes ✅	PASSES ✅:
				
				``java
				assertThat(List.of("1", "2", "3"))
				    .containsOnlyOnce("1", "2", "3");
				
				assertThat(List.of("1", "2", "2", "3"))
				    .containsOnlyOnce("1", "3");
				``
				
				FAILS ❌:
				
				``java
				// "2" found twice:
				assertThat(List.of("1", "2", "2", "3"))
				    .containsOnlyOnce("1", "2", "3");
				
				// dupes in expected doesn't matter:
				assertThat(List.of("1", "2", "2", "3"))
				    .containsOnlyOnce("1", "2", "2", "3");
				``

Created by Ted M. Young (January 9, 2023) https://ted.dev

Assertion	Actual must match Expected Order?	Actual can contain Duplicates?	Actual can contain elements other than in Expected?	Examples, in the form of:
				
				`assertThat(actual)`
				`    .contains(expected);`

`containsExactly`
 	
Note: use only when there is a predictable order of elements. Avoid use with Sets.
	yes ✅	yes ✅ if *Expected* also contains those duplicates	no ⛔	PASSES ✅:
				
				`assertThat(List.of("1", "2", "3"))`
				`    .containsExactly("1", "2", "3");`
				
				`assertThat(List.of("1", "1", "2", "2"))`
				`    .containsExactly("1", "1", "2", "2");`
				
				FAILS ❌:
				
				`// same elements, but different order`
				`assertThat(List.of("1", "2", "3"))`
				`    .containsExactly("3", "1", "2");`
				
				`// could not find "4"`
				`assertThat(List.of("1", "2", "3"))`
				`    .containsExactly("1", "2", "3", "4");`
				
				`// did not expect "2", "3"`
				`assertThat(List.of("1", "2", "3"))`
				`    .containsExactly("1");`
				
				`// did not expect the extra "1" and "2"`
				`assertThat(List.of("1", "1", "2", "2"))`
				`    .containsExactly("1", "2");`
				
				`// order not defined for sets`
				`// so this can fail`
				`assertThat(Set.of("3", "2", "1"))`
				`    .containsExactly("3", "2", "1");`

Assertion	Actual must match Expected Order?	Actual can contain Duplicates?	Actual can contain elements other than in Expected?	Examples, in the form of:
				
				``java
				assertThat(actual)
				    .contains(expected);
				``

`containsExactlyInAnyOrder`
 	
Note: use when order of elements is undefined, e.g., Set.
	no ⛔	yes ✅ if *Expected* also contains those duplicates	no ⛔	PASSES ✅:
				
				``java
				assertThat(List.of("1", "2", "3"))
				    .containsExactlyInAnyOrder("1", "2", "3");
				
				assertThat(List.of("1", "1", "2", "2"))
				    .containsExactlyInAnyOrder("1", "1", "2", "2");
				
				assertThat(Set.of("3", "2", "1"))
				    .containsExactlyInAnyOrder("1", "3", "2");
				``
				
				FAILS ❌:
				
				``java
				    // did not expect the extra "1" and "2"
				assertThat(List.of("1", "1", "2", "2"))
				    .containsExactlyInAnyOrder("1", "2");
				``