

Singleton Design Pattern

Context

- Classes for which only one instance should exist (singleton).
- Provide a global point of access.

Problem

- How do you ensure that it is never possible to create more than one instance of a singleton class?

Forces

- The use of a public constructor cannot guarantee that no more than one instance will be created.
- The singleton instance must be accessible to all classes that require it.

Singleton: Solution



Clients access a singleton instance solely through Singleton's `getInstance()` operation.

Iterator Design Pattern

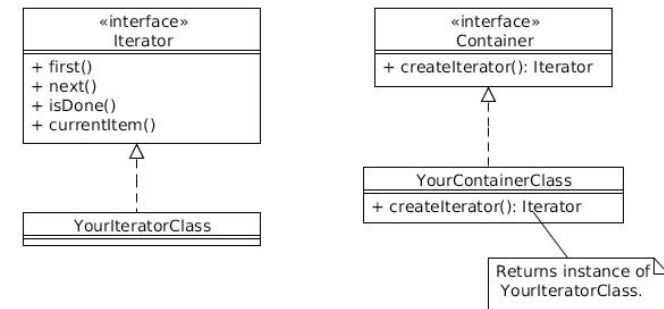
Context

- A container/collection object.

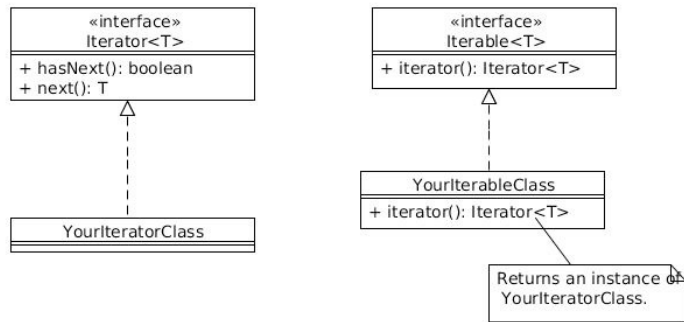
Problem

- Want a way to iterate over the elements of the container.
- Want to have multiple, independent iterators over the elements of the container.
- Do not want to expose the underlying representation (i.e., should not reveal how the elements are stored).

Iterator Design Pattern: Solution



Iterator Design Pattern: Java



Iterator: Example in Java

