Iterator Pattern

1. [Iterator Design Pattern](https://www.javatpoint.com/iterator-pattern)
2. [Advantage of Iterator DP](https://www.javatpoint.com/iterator-pattern#adv)
3. [Usage of Iterator DP](https://www.javatpoint.com/iterator-pattern#usage)
4. [UML of Iterator DP](https://www.javatpoint.com/iterator-pattern#uml)
5. [Example of Iterator DP](https://www.javatpoint.com/iterator-pattern#ex)

According to GoF, Iterator Pattern is used **"to access the elements of an aggregate object sequentially without exposing its underlying implementation".**

The Iterator pattern is also known as **Cursor.**

In collection framework, we are now using Iterator that is preferred over Enumeration.

java.util.Iterator interface uses Iterator Design Pattern.

Advantage of Iterator Pattern

* It supports variations in the traversal of a collection.
* It simplifies the interface to the collection.

Usage of Iterator Pattern:

It is used:

* When you want to access a collection of objects without exposing its internal representation.
* When there are multiple traversals of objects need to be supported in the collection.

Example of Iterator Pattern

Let's understand the example of iterator pattern pattern by the above UML diagram.

UML for Iterator Pattern:

