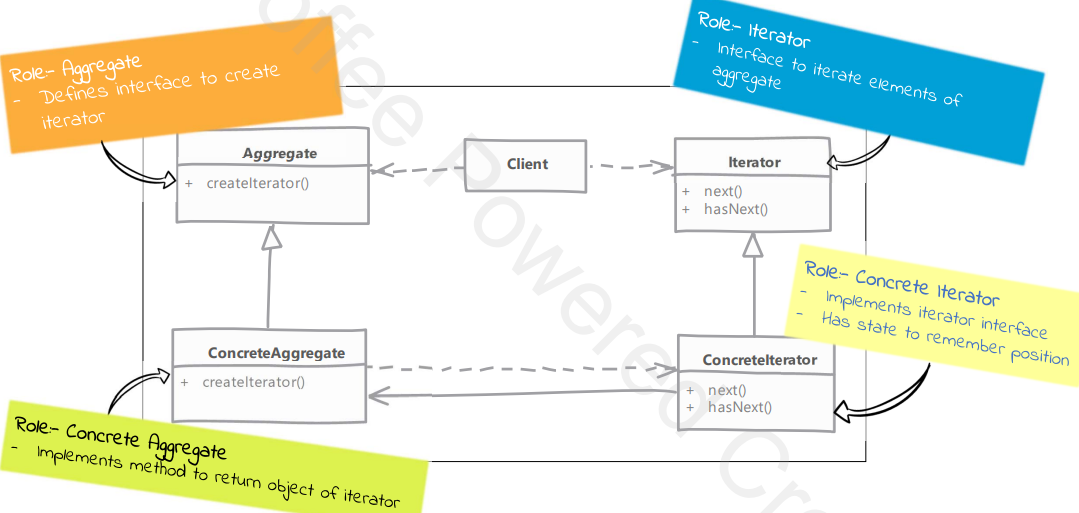
Iterator

Behavioral Design Pattern

**What is Iterator:**

1. Iterator allows a way to access elements/children of an aggregate object in sequence while hiding the actual internal data structure used.
2. In Java language Iterators are integral part of collection frameworks and they are implementations of this design pattern.
3. Iterators are stateful, meaning an iterator object remembers its position while iterating.

**UML:**



**Implementation steps:**

1. Defining Iterator interface.
   1. Iterator has methods to check whether there is an element available in sequence & to get that element.
2. Implement the iterator in a class. This is typically an inner class in our concrete aggregate. Making it an inner class makes it easy to access internal data structures.
3. Concrete iterator needs to maintain state to tell its position in collection of aggregate. IF the inner collection changes it can throw an exception to indicate invalid state.