Iterator pattern is very commonly used design pattern in Java and .Net programming environment. This pattern is used to **get a way to access the elements of a collection object in sequential manner without any need to know its underlying representation.**

Iterator pattern falls under **behavioral pattern category.**

## Implementation

We're going to create a *Iterator* interface which narrates navigation method(navigation in a collection like next and previous as in a linked list) and a *Container* interface which retruns the iterator(more like a layer between main class and Iterator) . Concrete classes implementing the *Container* interface will be responsible to implement *Iterator* interface and use it

*IteratorPatternDemo*, our demo class will use *NamesRepository*, a concrete class implementation to print a *Names* stored as a collection in *NamesRepository*.

Use of container as a layer or like acontainer which will give a concrete iterator quite lkike factory

Now by defintion how a iterator is working is hidden from client which concides with the defintion of factory where client do not know where object getrs instantiated

Spo to confuse the end user or hiding the implemation detail from end usder is a important characterstic of encapsulation and desgin pattern’

