Excerpts from an Example of the Iterator Pattern

```java
// AN ARRAY OF STRINGS WITH SUPPORT FOR AN ITERATOR

class StringArray
{
    private String[] data = { "AARDVARK", "BUFFALO", "CAT", "DOG", "ELEPHANT" };

    public Iterator iterator()
    {
        return new Iterator() {

            private int position = 0;

            public boolean hasNext()
            {
                return position < data.length;
            }

            public Object next()
            {
                Object value = data[position];
                position ++;
                return value;
            }

            public void remove() throws UnsupportedOperationException
            {
                throw new UnsupportedOperationException("remove");
            }
        };
    }
}
```

```java
// MAIN PROGRAM CLASS FOR DEMONSTRATION
class Demo
{
    public static void main(String [] args)

    ...

    /** Write out the contents of any collection, given an iterator for it
     *
     *   @param iterator the iterator for the collection
     */
    public static void writeOut(Iterator iterator)
    {
        while (iterator.hasNext())
            System.out.println(iterator.next());
        System.out.println("*** Done ***");
        System.out.println();
    }
}
```