

## Tutorial 3 - Behavioural Design Patterns

### 1) Iterator implementation example

In the code below complete the implementation of the private class **Itr**, i.e. fill in the ...

```
import java.util.*;

public class MyList extends AbstractCollection {

    private class Element {
        Object data;
        Element next;
        Element(Object d, Element n) {data=d; next=n;}
    }

    protected Element head = null;
    protected Element tail = null;
    protected int count = 0;

    public boolean add(Object o) {
        if (tail==null) {
            head = tail = new Element(o, null);
        } else {
            tail = tail.next = new Element(o, null);
        }
        ++count;
        return true;
    }

    public Iterator iterator() {
        return new Itr();
    }

    public int size() {return count;}

    private class Itr implements Iterator {
        ...
        Itr() {
            ...
        }
        public boolean hasNext() {
            ...
        }
        public Object next() {
            ...
        }
        public void remove() {
            throw new UnsupportedOperationException();
        }
    }
}
```