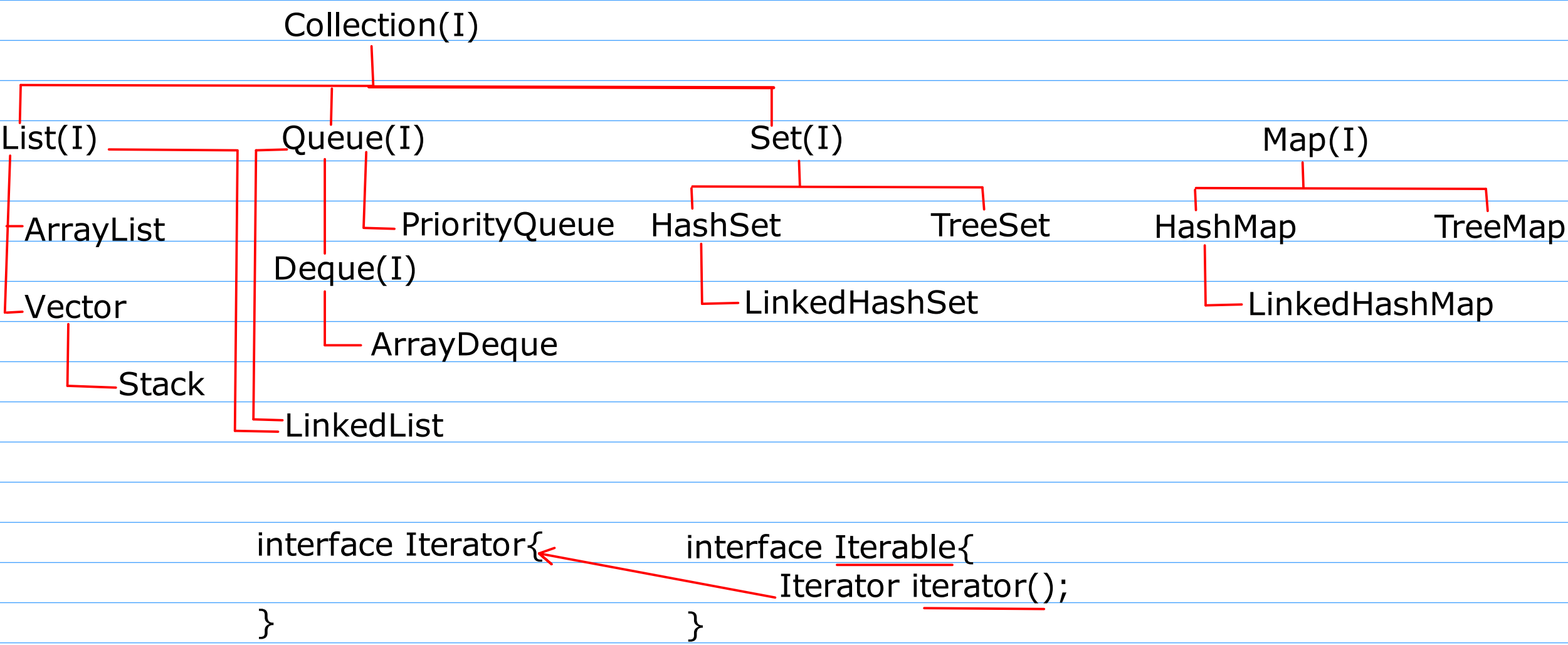


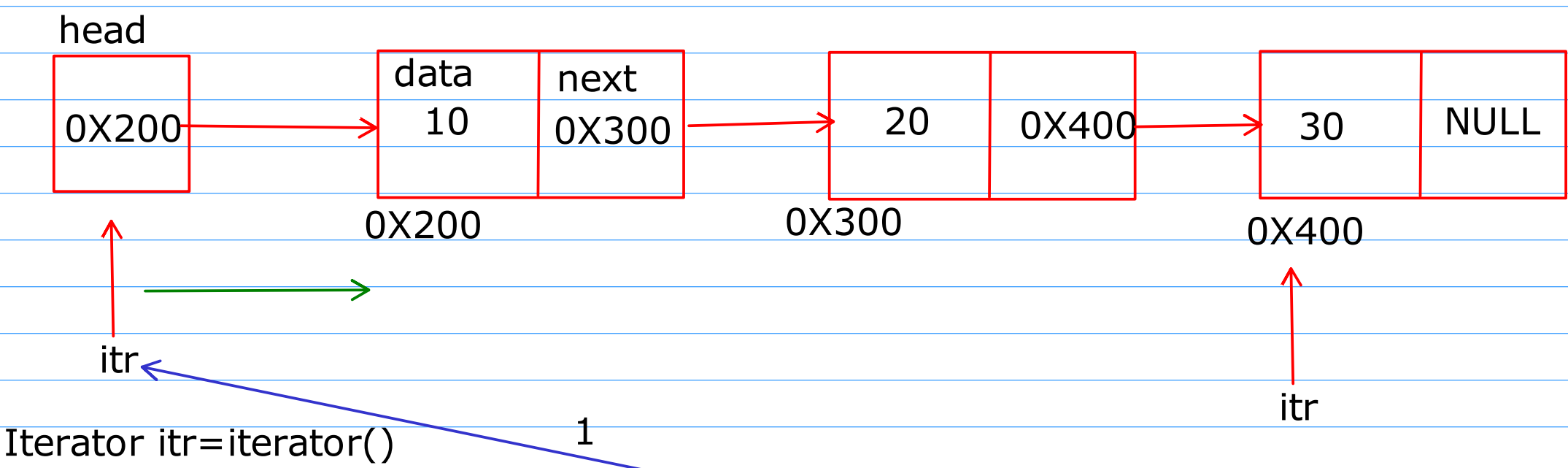
Collection Framework

- It is an implementation of Datastructure classes
- All these implemetation of the datastructures are given through generic interfaces and genric classes



For Traversal on your collection

1. using iterator
2. for-each
3. for-loop (List type of collection)
4. Enumaration (Vector)



```
Iterator itr=iterator()

Iterator<Integer> itr = c1.iterator();
while (itr.hasNext()) {
    Integer element = itr.next();
    System.out.println("element - " + element);
}
```

```

// Internal implementation of for-each
for (Iterator<Integer> itr = c1.iterator(); itr.hasNext();) {
    Integer element = itr.next();
    System.out.println("element - " + element);
}

// for-each
for (Integer element : c1)
    System.out.println("element - " + element);

```

```

class LinkedList implements Collection<Integer>{

```

```

class MyIterator implements Iterator<Integer>
{

```

```

    boolean hasNext(){
        if(trav.next != null)
            return true;
        return false;
    }

```

```

    E next()
    {
        Integer element = trav.next.data;
        trav = trav.next;
    }
}

```

```

Iterator<Integer> iterator(){
    return new MyIterator();
}
}

```

```

interface Showable {
    void print();
}

```

```

interface Printable extends Showable {
    void display();
}

```

```

class MyObject {
    public void print() {
        System.out.println("MyObject Print");
    }
}

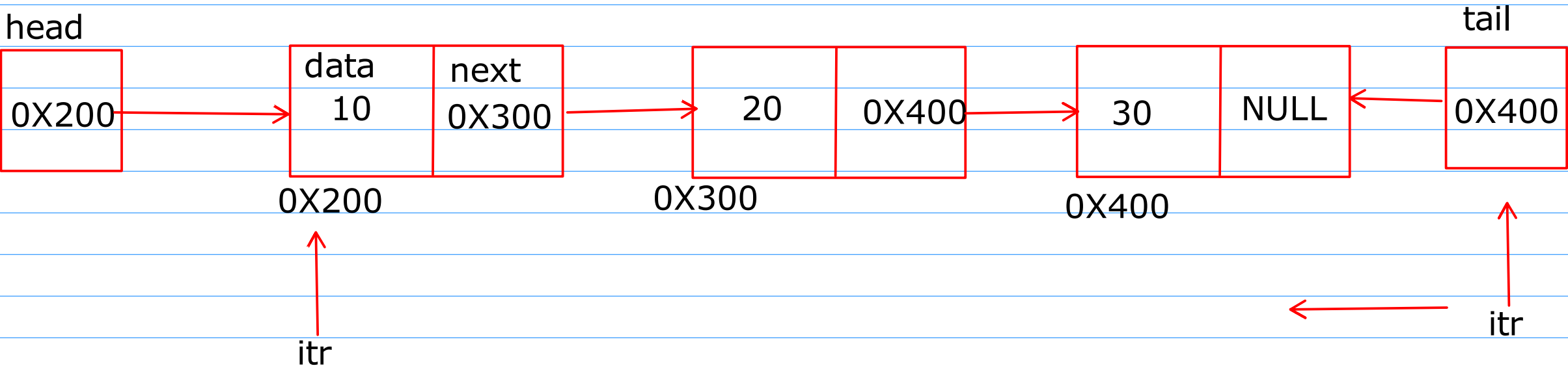
```

```

class MyArrayList extends MyObject implements Printable {

    @Override
    public void display() {
        // TODO Auto-generated method stub
    }
}

```



```
ListIterator itr=listIterator(l1.size())  
itr.hasprevious()  
itr.previous()
```

