

```
1 public class MyArrayList<AnyType> implements Iterable<AnyType>
2 {
3     private static final int DEFAULT_CAPACITY = 10;
4
5     private int theSize;
6     private AnyType [ ] theItems;
7
8     public MyArrayList( )
9     { clear( ); }
10
11    public void clear( )
12    {
13        theSize = 0;
14        ensureCapacity( DEFAULT_CAPACITY );
15    }
16
17    public int size( )
18    { return theSize; }
19    public boolean isEmpty( )
20    { return size( ) == 0; }
21    public void trimToSize( )
22    { ensureCapacity( size( ) ); }
23
24    public AnyType get( int idx )
25    {
26        if( idx < 0 || idx >= size( ) )
27            throw new ArrayIndexOutOfBoundsException( );
28        return theItems[ idx ];
29    }
30
31    public AnyType set( int idx, AnyType newVal )
32    {
33        if( idx < 0 || idx >= size( ) )
34            throw new ArrayIndexOutOfBoundsException( );
35        AnyType old = theItems[ idx ];
36        theItems[ idx ] = newVal;
37        return old;
38    }
39
40    public void ensureCapacity( int newCapacity )
41    {
42        if( newCapacity < theSize )
43            return;
44
45        AnyType [ ] old = theItems;
46        theItems = (AnyType [ ]) new Object[ newCapacity ];
47        for( int i = 0; i < size( ); i++ )
48            theItems[ i ] = old[ i ];
49    }
```

图 3-15 MyArrayList 类(第一部分, 共两部分)