

Java Programming II

Lab8

514770-1

Fall 2020

11/10/2020

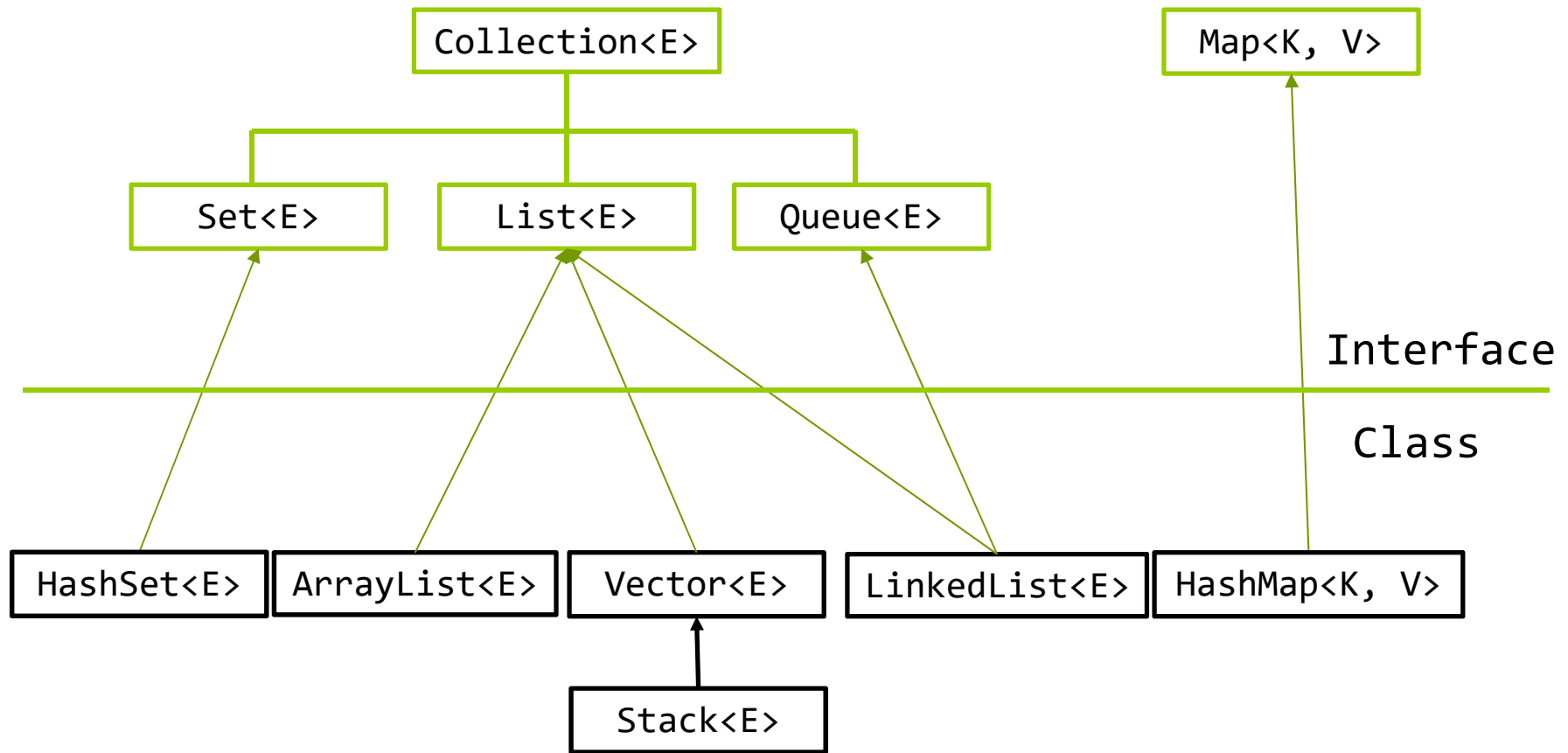
Kyoung Shin Park
Computer Engineering
Dankook University

Lab8

- Practice to write a program that adapts `List<E>` to `DataList<E>` using **Adapter pattern**.
 - `java.util.List<E>` interface
 - `ArrayList`, `LinkedList`, `Vector`, `Stack` class implements `List<E>`.
 - `DataList<E>` interface extends `java.util.Iterable<E>`.
 - `DynamicArray<E>` class implements *`DataList<E>`*.
 - `DataListAdapter<E>` adapter class takes the `List<E>` adaptee class to support the `DataList<E>` target interface.
 - `DataListAdapter` implements `DataList<E>`

Lab8

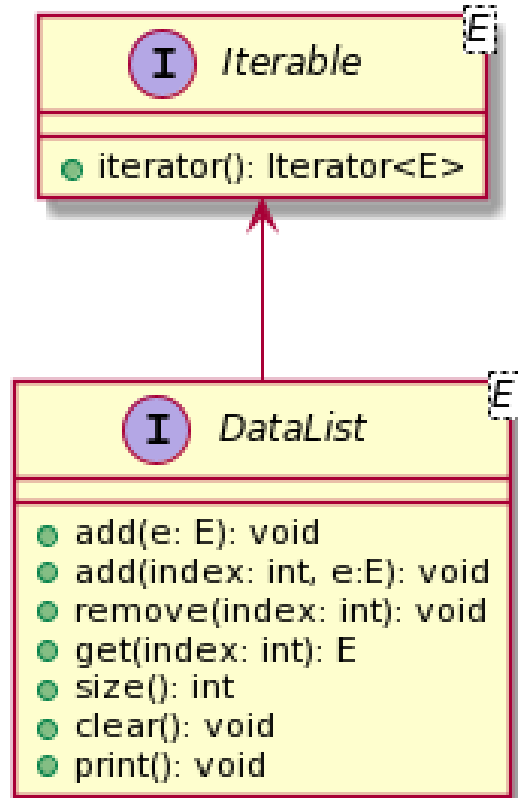
□ java.util.Collection



Lab8

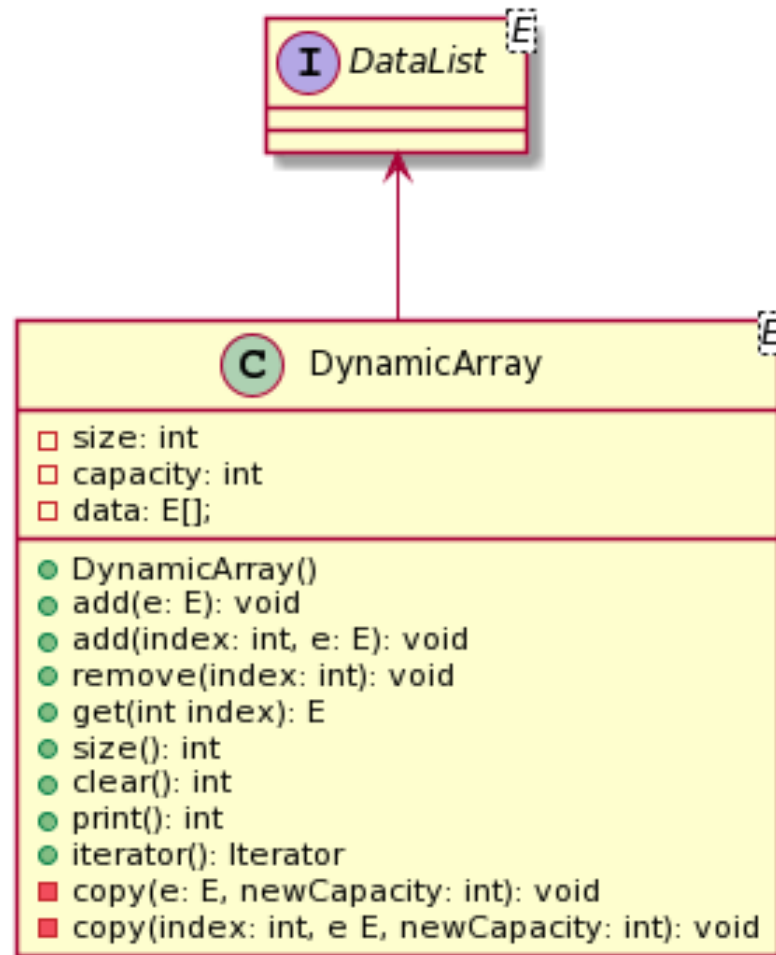
□ **DataList<E>** interface

```
import java.util.Iterator;  
public interface DataList<E> extends Iterable<E> {  
    void add(E e);  
    void add(int index, E e);  
    void remove(int index);  
    E get(int index);  
    int size();  
    void clear();  
    void print();  
}
```



Lab8

- **DynamicArray<E>** class implements **DataList<E>**
 - A generic dynamic array class using E[] data



Lab8

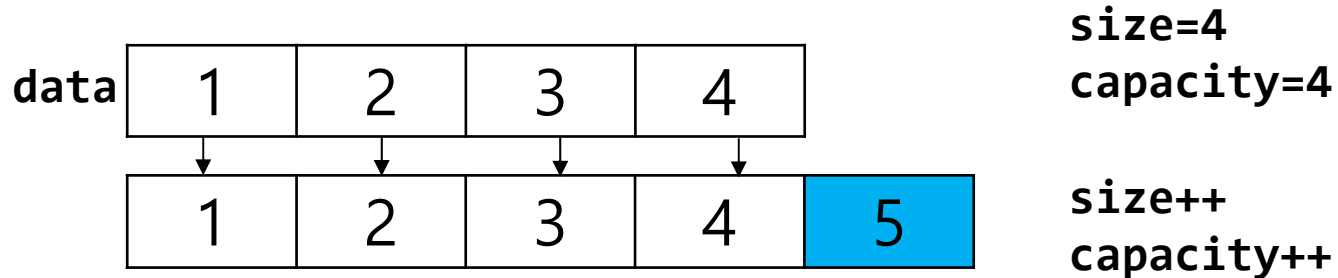
□ **DynamicArray<E>** class implements **DataList<E>**.

- private void copy(E data, int newCapacity); // increase array size & copy all data
- private void copy(int index, E data, int newCapacity); // increase array size & copy data before and after the index
- public void add(E e); // add element to the dynamic array
- public void add(int index, E e); // insert element at the index
- public void remove(int index); // remove element at the index
- public E get(int index); // get element at the index
- public int size(); // get the size of dynamic array (# of elements)
- public void clear(); // remove all elements and reset
- public void print(); // print all elements in the dynamic array (based on size)
- public Iterator<E> iterator(); // returns **DynamicArrayIterator()**

Lab8

□ Dynamic Array

- Add an element at the end of an array



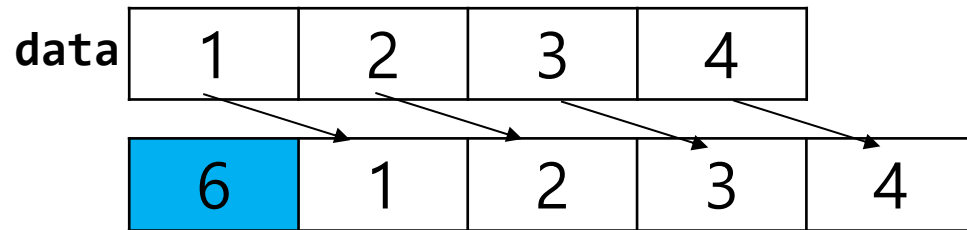
- Add an element at the end of an array



Lab8

□ Dynamic Array

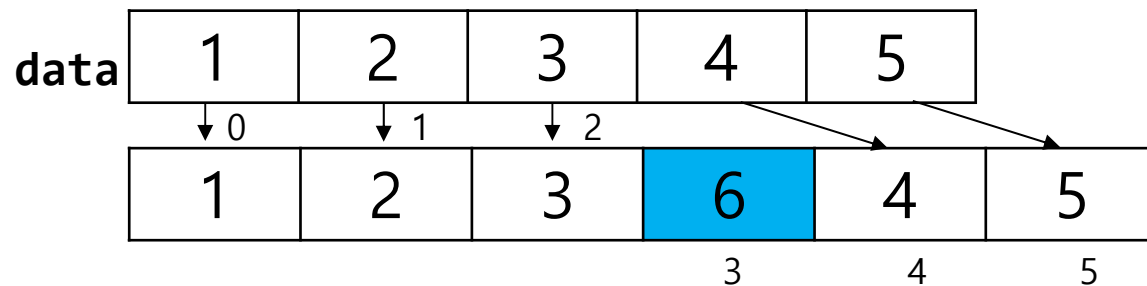
- Insert an element at the index 0



size=4
capacity=4

size++
capacity++

- Insert an element at the index 3



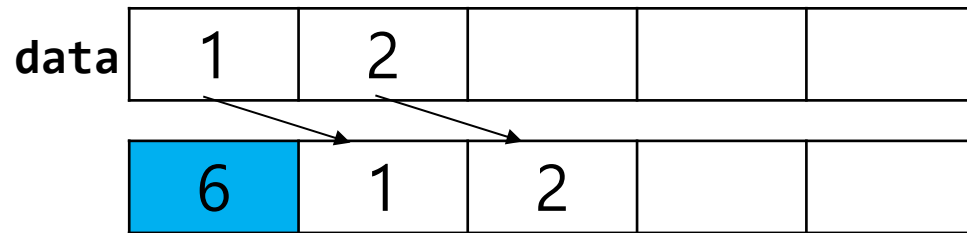
size=5
capacity=5

size++
capacity++

Lab8

□ Dynamic Array

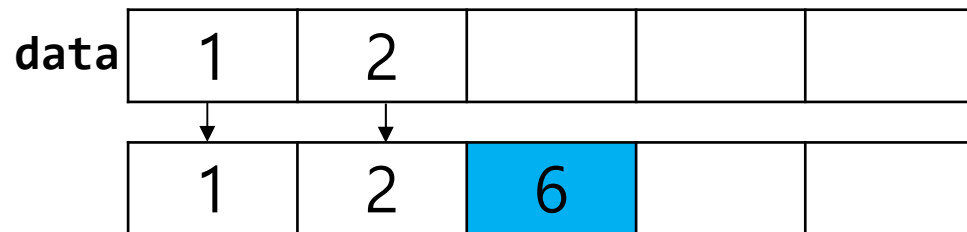
- Insert an element at the index 0



size=2
capacity=5

size++
capacity=5

- Insert an element at the index 2 – If (index > size) `ArrayIndexOutOfBoundsException`



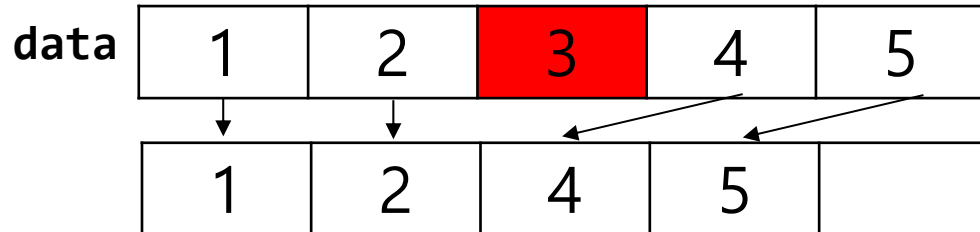
size=2
capacity=5

size++

Lab8

□ Dynamic Array

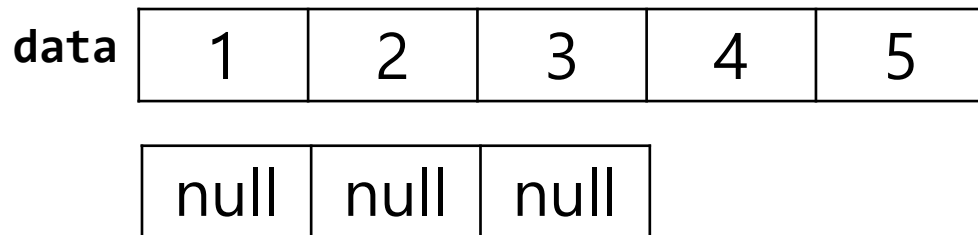
- Remove an element at the index 2



size=5
capacity=5

size--

- Clear

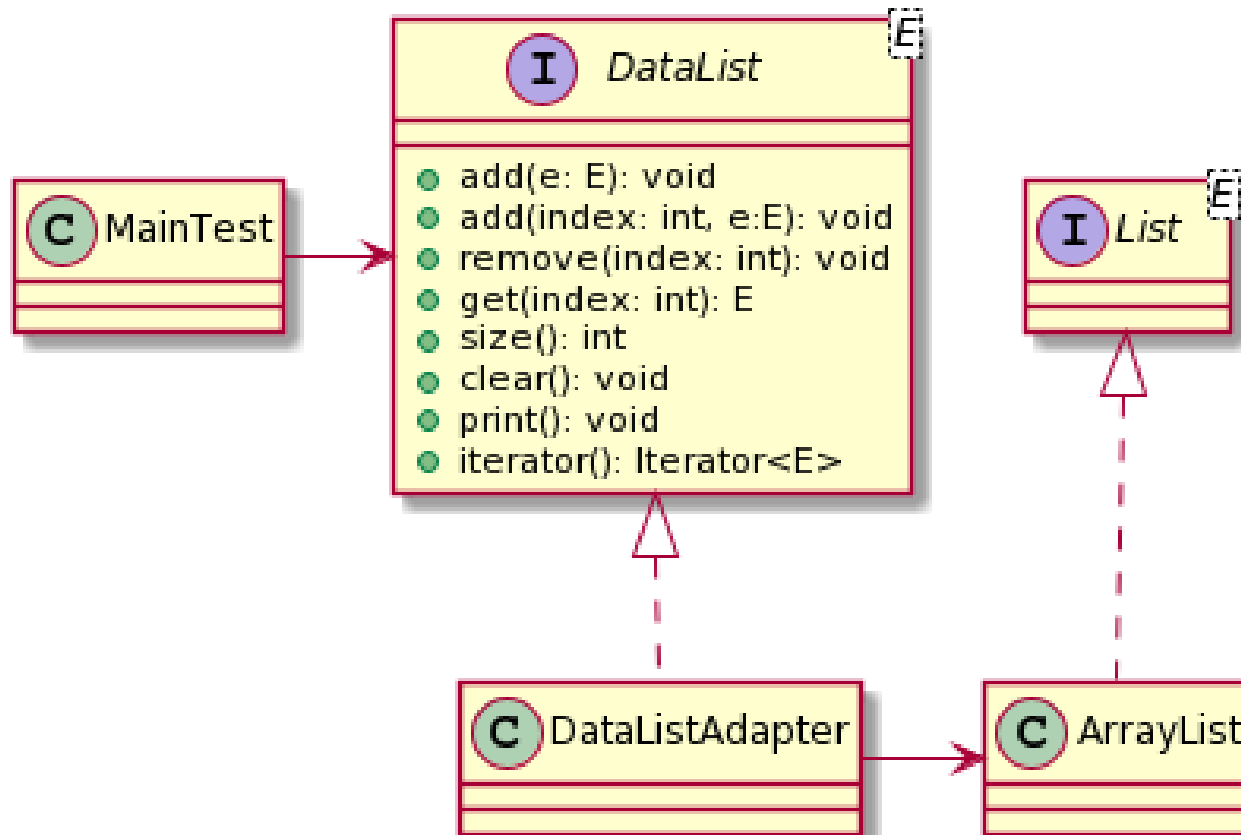


size=5
capacity=5

size=0 // reset
capacity=3 // reset

Lab8

- **DataListAdapter** adapter class takes the **List** adaptee class to support the **DataList** target interface.



Lab8

```
public class MainTest {
    public static void main(String[] args) {
        System.out.println("\n\nDynamicArray add & print");
        DataList<String> arr = new DynamicArray<>(5);
        arr.add("Seoul"); // add 12~16 elements
        arr.forEach(System.out::println); // test Iterable<E>
        // remove & remove all elements using arr.iterator()
        // insert & get & clear & print & for/while/foreach
        DataList<String> arr2 = new DataListAdapter<>(new ArrayList<>());
        arr2.add("Chicago"); // add elements
        // remove & remove all elements using arr2.iterator()
        // insert & get & clear & print & for/while/foreach
        DataList<String> arr3 = new DataListAdapter<>(new LinkedList<>());
        // ...
    }
}
```

Submit to e-learning

- ❑ Add your code (e.g., test all methods with DataList using ArrayList, LinkedList, Vector, Stack) in the Lab8 assignment.
- ❑ Submit the Lab8 assignment (JAVA20-2-Lab8-ID-name.zip including the report) to e-learning.