

Java Collections

[Array](#)

[Sample main for People class](#)

[ArrayList](#)

[HashMap](#)

People

- Implement and test this class:
 - This class uses the Person class you implemented in WS1

```
public class People {

    private Person[] people;
    int size;

    // default constructor - allocate memory for n Person objects
    public People( int max_size ){ }

    /*****
    * CRUD methods and others
    *****/

    // get number of elements in list
    public int size(){ return 0; }

    // CREATE: add Person p at available position
    public void add( Person p ){ }

    // READ: get Person at index i
    public Person get( int i ){ return null; }

    // UPDATE: set the record at index i to Person p
    public void set( int i, Person p ){ }

    // DELETE: remove Person p from Array, shift elements to the left, and update size
    public void remove( int i ){ }

    // find the first index of Person p
    public int indexOf( Person p ){ return -1; }

    /*****
    * toString
    *****/

    // get a String representation of the array
    public String toString(){ return ""; }

    /*****
    * Main
    *****/
    public static void main(String[] args){}

}
```

Sample main for People class

```
public static void main(String[] args){

    // create a new People collection that can hold 10 people
    People ppl = new People(10);
    Person p;

    // CREATE - this loop tries to add 11 people but fails on the last
    for(int i=0; i<11; i++){
        // create a new Person and add it to the collection
        p = new Person("Johnny", "Number"+i);
        ppl.add(p);
    }
    // READ - tries to read 11 but fails (gracefully) on the last
    for(int i=0; i<11; i++){
        // get and print the Person at index i
        p = ppl.get(i);
        System.out.println(p);
    }
    // print the size
    System.out.println(ppl.size());

    // remove the Person at index 5
    ppl.remove(5);

    // print the collection (note that elements have shifted)
    System.out.println(ppl);
}
```

```
ADD ERROR: collection is full
Number0, Johnny 01-01-1970
Number1, Johnny 01-01-1970
Number2, Johnny 01-01-1970
Number3, Johnny 01-01-1970
Number4, Johnny 01-01-1970
Number5, Johnny 01-01-1970
Number6, Johnny 01-01-1970
Number7, Johnny 01-01-1970
Number8, Johnny 01-01-1970
Number9, Johnny 01-01-1970
GET ERROR: invalid index
null
10
[
    0      Number0, Johnny 01-01-1970
    1      Number1, Johnny 01-01-1970
    2      Number2, Johnny 01-01-1970
    3      Number3, Johnny 01-01-1970
    4      Number4, Johnny 01-01-1970
    5      Number6, Johnny 01-01-1970
    6      Number7, Johnny 01-01-1970
    7      Number8, Johnny 01-01-1970
    8      Number9, Johnny 01-01-1970
]
```

ArrayList

1. Each of the above methods are implemented in the Java ArrayList class. Demonstrate them in a demo class with only a main. (Use your ArrayList to hold People objects)
2. Compare and contrast Arrays and ArrayLists.

HashMap

1. Demonstrate the following methods of a HashMap: (Use your HashMap to hold < id, People> pairs)
 - a. put
 - b. get
 - c. remove
2. Compare and contrast ArrayLists and HashMaps