Lab 11: Generic Methods

Exercise 1 : Add elements of array to same kind of list

```
String[] strs = {"a", "b", "c","d"};
List<String> strList = new ArrayList<>();
addToCollection(strs,strList);

Integer[] ints = {1, 2, 3, 4};
List<Integer> intList = new ArrayList<>();
addToCollection(ints,intList);
```

1. Implement the method addToColection which copies the content of the given array to the given List of same time of objects..
2. Modify your implementation to support generics.

Exercise 2 : Copy elements of list to a list containing more general types

```
List<Number> numList = new ArrayList<>();

List<Integer> intList = Arrays.asList(1,2,3,4);
copyCollection(intList,numList);


List<Double> dList = Arrays.asList(1.3,2.0,3.5,4.0);
copyCollection(dList,numList);
```

1.Implement the method copyCollection which copies the content of the source list to the destination list containing more general types of elements.
2. Modify your implementation to support generics.

Exercise 3 : Implement a method which finds the maximum of given objects based on their natural orders

```
System.out.printf("Maximum of %d, %d and %d is %d\n\n",
    3, 4, 5, max( 3, 4, 5 ));

System.out.printf("Maximum of %.2f, %.2f and %.2f is %.2f\n\n",
    6.6, 8.8, 7.7, max( 6.67, 8.89, 7.75 ));

System.out.printf("Maximum of %s, %s and %s is %s\n","pear",
    "apple", "orange", max("pear", "apple", "orange"));
```

1. Implement the max method without generics.
2. Modify your implementation to support generics.

NOTE: Your lab will **not be graded** if

- Your account name does not have the format described in lab1.pdf
- Your repository name is not lab
- Your files have compilation errors
- You haven't complete the steps described in exercises
- Your added/modified files are not submitted to Bitbucket.
  - You have to add commit and push files as described in lab1.pdf