CSE 212

SOFTWARE DEVELOPMENT METHODOLOGIES

Yeditepe University

Spring 2017

Due: End of Lab Session

GENERICS ASSIGNMENT

1. Make a class called **PairClass<T>** that stores two entries of a given type and has getter methods to retrieve them. Provide a **toString** method as well. Here are two examples of its use:

```
PairClass<String> twoNames = new PairClass<>("Ali", "Jale");
System.out.printf("twoNames=%s.%n", twoNames);
String name1 = twoNames.getItem1(); // Value is "Ali"
String name2 = twoNames.getItem2(); // Value is "Jale"
System.out.println(" First item: " + name1);
System.out.println(" Second item: " + name2);

PairClass<Integer> twoNums = new PairClass<>(4, 7);
System.out.println("twoNums=" + twoNums);//calling toString implicitly
Integer num1 = twoNums.getItem1(); // Value is 4
Integer num2 = twoNums.getItem2(); // Value is 7
System.out.println(" First item: " + num1);
System.out.println(" Second item: " + num2);
```

2. Write another class **PairClass2<F,S>** where the first pair has a type of **F** and second pair has a type of **S**. Provide get and set methods. Also implement **Comparable** interface so that when two objects of **PairClass2** is compared, the first pair (**F**) is going to be checked firstly. If two objects first pair is equal then, two second pairs (**S**) will be compared.

For example, if one object holds (Integer and String) 5 and “hello” and the other holds 5 and “world”, you should first check if the integers are greater or smaller than one another. If they match, then the strings will be compared. In this example the second object is greater because even if they have identical integers the second object’s string starts with ‘w’ so it is greater than ‘h’.