

- Experiment 8 -

Create a Java project named Experiment8 and create the following classes in the project.

1. Define a public class ArrayListTest, which has a no-argument constructor, and complete the following tasks in the constructor:

- Create an ArrayList object and call *add* method to add some names: "James", "Tom", "Steven", "Alice", "Bob", "Eve", "Mary", "Jerry", and use the codes below to output all the names in the ArrayList object:

```
for (int count=0; count<list.size(); count++)  
    System.out.print(list.get(count)+" ");  
System.out.println();
```
- Use an iterator to output all names in the ArrayList object, and delete Tom.
- Call the sort method of Collections to sort the names in ascending order.
- Use for-each loop to output all the names in the ArrayList object to see if Tom has been removed or not.

Next, create an ArrayListTest object in the main method of class ArrayListTest to call the no-argument constructor.

[\[Sample\]](#)

```
Console x
<terminated> ArrayListTest [Java Application]
James Tom Steven Alice Bob Eve Mary Jerry
James, Tom, Steven, Alice, Bob, Eve, Mary, Jerry,
Alice, Bob, Eve, James, Jerry, Mary, Steven,
```

2. Write a public class LinkedListTest, use LinkedList instead of ArrayList to redo the above experiment.

[\[Sample\]](#)

```
Console x
<terminated> LinkedListTest [Java Application]
James Tom Steven Alice Bob Eve Mary Jerry
James, Tom, Steven, Alice, Bob, Eve, Mary, Jerry,
Alice, Bob, Eve, James, Jerry, Mary, Steven,
```

3. Define a class named Student, whose private instance fields are: id, name, age, and its methods are: constructor with parameters, no-argument constructor, getName, and toString.

Define a public class named HashMapTest, write a no-argument constructor, complete the following tasks in this constructor:

- Add the following students into a HashMap object:
id: "17H002", name: "孙悟空", age: 2000

```
id: "17H001", name: "唐僧", age: 40
id: "17H003", name: "猪八戒", age: 1000
id: "17H004", name: "沙僧", age: 900
```

- Output information of the student whose id is 17H003
- Output information of all students
- remove 唐僧
- Turn HashMap into an array of type Student, then use a for-each loop to print out all the elements of the array.

And create a HashMapTest object in the main method of class HashMapTest to call the no-argument constructor.

[Sample]

```
Console
<terminated> HashMapTest [Java Application]
学号17H003的学生信息：
学号：17H003，姓名：猪八戒，年龄1000
所有学生信息：
学号：17H003，姓名：猪八戒，年龄1000
学号：17H002，姓名：孙悟空，年龄2000
学号：17H004，姓名：沙僧，年龄900
学号：17H001，姓名：唐僧，年龄40
删除唐僧后所有学生信息：
学号：17H003，姓名：猪八戒，年龄1000
学号：17H002，姓名：孙悟空，年龄2000
学号：17H004，姓名：沙僧，年龄900
把HashMap变成数组后，所有学生信息：
学号：17H003，姓名：猪八戒，年龄1000
学号：17H002，姓名：孙悟空，年龄2000
学号：17H004，姓名：沙僧，年龄900
```

4. Given a generic interface as follows:

```
interface Arithmetical <T> {
    T add(T obj);
    T subtract(T obj);
    T multiply(T obj);
    T divide(T obj);
}
```

Define class Complex or class Fraction to implement the generic interface Arithmetical <T>, where class Complex implements the addition, subtraction, multiplication and division of complex numbers, and class Fraction implements the addition, subtraction, multiplication and division of fractions.

Next, define a main class named MyGenericProgram to test the validity of the class you defined. Your program needn't take into account the case of division by 0.

[Sample]

```
Console
<terminated> MyGenericProgram [Java Application]

下面进行3次分数计算：
第1次分数计算：
分数1： a/b， 分别输入整数 a 和 b : 1 2
分数2： a/b， 分别输入整数 a 和 b : 3 4
分数1： 1/2 分数2： 3/4
分数1+分数2： 10/8
分数1-分数2： -2/8
分数1*分数2： 3/8
分数1/分数2： 4/6
第2次分数计算：
分数1： a/b， 分别输入整数 a 和 b : -1 2
分数2： a/b， 分别输入整数 a 和 b : 3 -4
分数1： -1/2 分数2： -3/4
分数1+分数2： -10/8
分数1-分数2： 2/8
分数1*分数2： 3/8
分数1/分数2： 4/6
第3次分数计算：
分数1： a/b， 分别输入整数 a 和 b : 0 2
分数2： a/b， 分别输入整数 a 和 b : 5 -7
分数1： 0/2 分数2： -5/7
分数1+分数2： -10/14
分数1-分数2： 10/14
分数1*分数2： 0/14
分数1/分数2： 0/10
--- 分数计算结束! ---
```

Or

Console

<terminated> MyGenericProgram [Java Application]

下面进行3次复数计算：

第1次复数计算：

复数1: $a+bi$, 分别输入实数 a 和 b : 1 2

复数2: $a+bi$, 分别输入实数 a 和 b : 3 4

复数1: $1.0+2.0i$ 复数2: $3.0+4.0i$

复数1+复数2: $4.0+6.0i$

复数1-复数2: $-2.0-2.0i$

复数1*复数2: $-5.0+10.0i$

复数1/复数2: $0.44+0.08i$

第2次复数计算：

复数1: $a+bi$, 分别输入实数 a 和 b : -1 2

复数2: $a+bi$, 分别输入实数 a 和 b : 3 -4

复数1: $-1.0+2.0i$ 复数2: $3.0-4.0i$

复数1+复数2: $2.0-2.0i$

复数1-复数2: $-4.0+6.0i$

复数1*复数2: $5.0+10.0i$

复数1/复数2: $-0.44+0.08i$

第3次复数计算：

复数1: $a+bi$, 分别输入实数 a 和 b : 0 5

复数2: $a+bi$, 分别输入实数 a 和 b : 6 0

复数1: $5.0i$ 复数2: 6.0

复数1+复数2: $6.0+5.0i$

复数1-复数2: $-6.0+5.0i$

复数1*复数2: $30.0i$

复数1/复数2: $0.8333333333333334i$

--- 复数计算结束! ---