Assignments for Week-9
Generic & Collection

1. Write a program to find the maximum value.
   a. Use Scanner class to get positive integers until user input -1.
   b. Save all input numbers in **Vector**
   c. Search maximum number and print it

```
Enter integer(-1 to quit)>> 1 100 50 8 7 9 -1
Maximum number is 100
```

2. Write a program to find the average
   a. Use Scanner class to get the grade of 6 students(‘A’, ‘B’, ‘C’, ‘D’, ‘E’, ‘F’)
   b. Store input grades to ArrayList
   c. Convert grade A=4.0, B=3.0, C=2.0, D=1.0, F=0.0
   d. Calculate average of converted values & print it
   e. If any of input grades is not A, B, C, D or F, print “Invalid”

```
Enter grades of 6 students (A/B/C/D/F)>> A B C D F A
Average : 2.3333333333333335
```

```
Enter grades of 6 students (A/B/C/D/F)>> A B C D E F
Invalid
```

3. Write a program
   a. Use Scanner class to get “Nation” & “Population” and save it in HashMap.
   b. If user input “stop” for “Nation”, stop to get and display “Search population of “
   c. Use Scanner class to get “Nation”
   d. Search nation and print the result.
   e. If the input nation is not found, print error message
   f. If the user enters “exit”, quit.

```
Enter nation & population (Ex: Korea 5000)
Nation & population >> Korea 5000
Nation & population >> France 3000
Nation & population >> USA 100000
Nation & population >> India 1400000
Nation & population >> stop
Search population of Korea
Population of Korea is 5000
Search population of China
Cannot find China
Search population of India
```

```
Population of India is 1400000
Search population of exit
```

4. Write a program to maintain the amount of rainfall
    a. Get current amount of rainfall and save in Vector collection
    b. Print all input amounts of rainfall and average.
    c. if User input 0, quit.

```
Enter amount of rainfall (0 to quit)>> 5
5
Current Average : 5
Enter amount of rainfall (0 to quit)>> 80
5 80
Current Average : 42
Enter amount of rainfall (0 to quit)>> 53
5 80 53
Current Average : 46
Enter amount of rainfall (0 to quit)>> 22
5 80 53 22
Current Average : 40
Enter amount of rainfall (0 to quit)>> 0
```

5. Student class represents the information of a student. It has the following fields

    name
    major
    id
    average of grade

    a. Create a Student object for 4 students
    b. Save information to ArrayList<Student> (Get information by using Scanner class) The following is the method to get user input.

```
private void read() {
  System.out.println("Enter student name, major, id and average
of grade.");
  for (int i=0; i<4; i++) {
    System.out.print(">> ");
    String text = scanner.nextLine();
    StringTokenizer st = new StringTokenizer(text, ",");
    String name = st.nextToken().trim();
    String department = st.nextToken().trim();
    String id = st.nextToken().trim();
    double grade = Double.parseDouble(st.nextToken().trim());
    // TODO : Add your code at here to save information
  }
}
```

c. Print all 4 students' information. Make printAll() method.
d. Get name of student, search and print information. Make processQuery() method.
e. If user input “exit”, quit. In processQuery() method.

```
Enter student name, major, id and average of grade.
>> Jobs, Mobile, 1, 4.1
>> Cook, SW, 2, 3.9
>> Gates, Design, 3, 3.5
>> Zukerberg, Mobile, 4, 3.1
---------------------------
Name : Jobs
Major : Mobile
ID : 1
Avg. Grade : 4.1
---------------------------
---------------------------
Name : Cook
Major : SW
ID : 2
Avg. Grade : 3.9
---------------------------
---------------------------
Name : Gates
Major : Design
ID : 3
Avg. Grade : 3.5
---------------------------
---------------------------
Name : Zukerberg
Major : Mobile
ID : 4
Avg. Grade : 3.1
---------------------------
Student name >> Gates
Gates, Design, 3, 3.5
Student name >> Bill
Student name >> Jobs
Jobs, Mobile, 1, 4.1
Student name >> exit
```

6. Rewrite Q5 by using HashMap<String, Student>. The key is the name of the student.

7. Write a program to save and find the location of a City.
   a. Create the Location class which has City, Latitude and Longitude information.
   b. Create HashMap<String, Location> collection
   c. Get 4 City names, latitudes and longitudes. Modify read() method of Q5.
   d. Print all 4 city information. Make prinAll() method.
   e. Get the city name & print the latitude and longitude. Make processQuery() method.

```
Enter City name, latitude, longitude
```

```
>> Seoul, 37, 126
>> LA, 34, -118
>> Paris, 2, 48
>> Sydney, 151, -33
---------------------------
Seoul   37.0    126.0
LA      34.0    -118.0
Paris   2.0     48.0
Sydney  151.0   -33.0
---------------------------
City name >> Paris
Paris   2.0     48.0
City name >> Seoul
Seoul   37.0    126.0
City name >> Tokyo
Location of Tokyo is unknown.
City name >> exit
```

8. Write a program to find scholarship students.
    a. Get 5 students' names and grades(~4.5), save it in HashMap.
    b. Get the limit of grade for scholarship
    c. Find all students who can get the scholarship

```
DK Scholarship management system.
Name & grade >> Jobs 4.1
Name & grade >> Cook 3.5
Name & grade >> Musk 2.0
Name & grade >> Gates 3.7
Name & grade >> Bezos 3.1
Input grade limitation >> 3.5
Scholarship students : Jobs Cook Gates
```

9. Write a program to manage Reward points.
    a. Get a customer name & reward point.
    b. Save it in HashMap
    c. If the customer name exist, add reward point and print information
    d. If the customer name does not exist, add a new customer.
    e. If input "exit", quit.

```
** REWARD Management System **
Enter name & reward point >> Woojin 40
(Woojin,40)
Enter name & reward point >> Woojin 30
(Woojin,70)
Enter name & reward point >> Musk 100
(Musk,100)(Woojin,70)
Enter name & reward point >> Musk 10
(Musk,110)(Woojin,70)
```

```
Enter name & reward point >> Woojin 50
(Musk,110)(Woojin,120)
Enter name & reward point >> exit
```

10. The following is IStack<T> interface

```
public interface IStack<T> {
  public T pop();
  public boolean push(T ob);
}
```

   a. Write the MyStack<T> class which implements the IStack<T> interface
   b. Test the following StackManage class which specialize MyStack<T> to MyStack<Integer>

```
public class StackManager {
      public static void main (String[] args) {
            IStack<Integer> stack = new MyStack<Integer>();
            for (int i=0; i<10; i++) stack.push(i);
            while (true) {
                  Integer n = stack.pop();
                  if(n == null) break;
                  System.out.print(n + " ");
            }
      }
}
```

   c. The result should be :

```
9 8 7 6 5 4 3 2 1 0
```