https://hc.labnet.sfbu.edu/~henry/sfbu/course/introjava/java\_class/slide/exercises3.html

Q18 ==> Hand

--------------------------

| Hand |

| |

| ------------- |

| | Palm | |

| | int size | |

| ------------- |

| ------------- |

| | Finger | ....... |

| | int len | ....... |

| ------------- |

--------------------------

Complete the following program, including compile, debug, and execute the program. This program is to implement the aggregation class "Hand". The execution result should be the same as the comment lines shown in main function.

import java.util.\*;

import java.lang.\*;

class Palm implements Cloneable {

////////////////////////

// Data memebers

////////////////////////

private int size;

//////////////////////////

// Member functions

/////////////////////////

// Manager function

public Palm(int size1) {

}

public Object clone() {

}

//////////////////////////

// Member functions

/////////////////////////

// Access functions

public boolean isLarge() { } // A palm is large if

// size is greater than 50

public int getSize() { return size; }

public void setSize(int size1) { size = size1; }

public boolean equals(Object obj) {

}

// Implementor function

public void enlarge(int sizeInc) { size += sizeInc; }

public void shrink(int sizeDec) { size -= sizeDec; }

public String toString() { // See the Demo class to figure out how to

// implement this function

}

}

class Finger implements Cloneable {

////////////////////////

// Data members

////////////////////////

private int len;

//////////////////////////

// Member functions

/////////////////////////

// Manager function

public Finger(int len1) {

}

public Object clone() {

}

// Access functions

public boolean isLong() { } // A finger is long if

// its length is longer than 30

public int getLen() { return len; }

public void setLen(int len1) { len = len1; }

public boolean equals(Object obj) {

}

// Implementor function

public void enlarge(int lenInc) { len += lenInc; }

public void shrink(int lenDec) { len -= lenDec; }

public String toString() { // See the example in the Demo class to figure

// out how to implement this function

}

}

class Hand implements Cloneable {

////////////////////////

// Data members

////////////////////////

private Palm palm;

private Vector fingers;

//////////////////////////

// Member functions

/////////////////////////

// Manager function

public Hand(Palm palm1, Vector fingers1) {

}

public Object clone()

{

}

// Predicate function

public boolean isNormal() { } // A hand is normal if it

// has 5 fingers

public boolean equals(Object obj) {

}

// Access functions

public Palm getPalm() {

}

public Finger getFinger(int i) {

}

public Vector getFingers() {

}

// Implementor function

public int numOfFingers() { }

public void lostOneFinger(int index) { }

public String toString() {

}

}

/\* Will display

Experiment clone ...

Size of palm is 15

Size of palm is 20

Size of palm is 20

Display information about Hand ...

Size of palm is 20

Finger 0: Length of finger is 3

Finger 1: Length of finger is 4

Finger 2: Length of finger is 5

\*/

public class Demo {

public static void main (String argv[])

{

System.out.println("\nExperiment clone ...");

Palm palm= new Palm(15);

System.out.println(palm); // print "Size of palm is 15"

Palm palm1= palm;

palm1.setSize(20);

System.out.println(palm); // print "Size of palm is 20"

Palm palm2= (Palm)palm.clone();

palm2.setSize(30);

System.out.println(palm); // print "Size of palm is 20"

Vector fingers = new Vector();

fingers.addElement(new Finger(3));

fingers.addElement(new Finger(4));

fingers.addElement(new Finger(5));

System.out.println("\nDisplay information about Hand ...");

Hand p = new Hand(palm, fingers);

System.out.println(p);

}

}

References

* [Class with dynamic object array attribute](https://hc.labnet.sfbu.edu/~henry/sfbu/course/introjava/java_class/slide/classStruct.htm#Class%20with%20dynamic%20object%20array%20attribute)
* [Class Structure : Simple Class vs. Aggregation](https://hc.labnet.sfbu.edu/~henry/sfbu/course/introjava/java_class/slide/classStruct.htm)
* [Constructor Types](https://hc.labnet.sfbu.edu/~henry/sfbu/course/introjava/java_class/slide/constructor_type.html)
* [Summary of toString(), clone(), equals()](https://hc.labnet.sfbu.edu/~henry/sfbu/course/introjava/java_class/slide/summary_table.html)
* [4 types of member functions](https://hc.labnet.sfbu.edu/~henry/sfbu/course/introjava/java_class/slide/javaMemFunc.html)
* [Vector](https://hc.labnet.sfbu.edu/~henry/sfbu/course/introjava/java_class/slide/vector.html)
* [Clone](https://hc.labnet.sfbu.edu/~henry/sfbu/course/introjava/java_class/slide/clone.html) (with and without object attributes)
* [toString()](https://hc.labnet.sfbu.edu/~henry/sfbu/course/introjava/java_class/slide/toString.html)
* [equals()](https://hc.labnet.sfbu.edu/~henry/sfbu/course/introjava/java_class/slide/equalMethod.html)