```java
package com.torryharris.mainpack;

import com.torryharris.mydatepack.mydate;

public class Main {

    public static void main(String[] args) {
   // write your code here
        mydate d1 = new mydate(27,9,2021);
        mydate d2 = new mydate(27,9,2021);
        mydate d3 = d1;
        System.out.println("*****"+d1.equals(d2));
        System.out.println(d1.equals(d3));

        System.out.println(d1==d2);
        System.out.println(d1==d3);

        System.out.println("d1 hashcode= "+d1.hashCode());
        System.out.println("d2 hashcode= "+d2.hashCode());
        System.out.println("d3 hashcode= "+d3.hashCode());

    }
}
```

output:

```
*****true

true

false

true

d1 hashcode= 2039

d2 hashcode= 2039

d3 hashcode= 2039
```

==========

```java
package com.torryharris.mainpack;

import java.util.HashSet;
import java.util.LinkedHashSet;
import java.util.Set;
import java.util.TreeSet;

public class Main {

    public static void main(String[] args) {
   // write your code here
        Set s1 = new HashSet();
        s1.add(1);
        s1.add(12.554);
        s1.add("java");
        s1.add(null);
        s1.add(13.1f);
        System.out.println(s1);
        Set<String>  s2 = new HashSet<String>();
        s2.add("zzz");
        s2.add("aaa");
        s2.add("xyz");
        s2.add("aaa");
        s2.add("xyz");
        System.out.println(s2);
        Set<String> s3 = new LinkedHashSet<String>();
        s3.add("aaa");
        s3.add("xyz");
        s3.add("aaa");
        s3.add("xyz");
        s3.add("zzz");
        System.out.println(s3);
        Set<String> s4 = new TreeSet<String>();
        s4.add("varun");
        s4.add("suhas");
        s4.add("ajay");
        s4.add("abhilash");
        s4.add("varun");
        System.out.println(s4);


    }
}
```

output:

```
[12.554, null, 1, java, 13.1]

[aaa, xyz, zzz]

[aaa, xyz, zzz]

[abhilash, ajay, suhas, varun]
```

==========

```java
package com.torryharris.mainpack;

import java.util.Stack;
import java.util.Vector;

public class Main {

    public static void main(String[] args) {
   // write your code here
        //Vector<String> st = new Vector<<String>();
        Stack<String> st = new Stack<String>();
        st.push("aaa");
        st.push("bbb");
        st.push("ccc");
        System.out.println(st.pop());
        System.out.println(st.peek());
        System.out.println(st);
        st.add(1,"zzz");
        System.out.println(st);
        System.out.println(st.search("aaa"));
        System.out.println(st.search("bbb"));
    }
}
```

output:

```
ccc

bbb

[aaa, bbb]

[aaa, zzz, bbb]

3

1
```

==========

```java
package com.torryharris.mainpack;

import java.util.LinkedList;
import java.util.PriorityQueue;
import java.util.Queue;

public class Main {

    public static void main(String[] args) {
   // write your code here
        Queue<String> q = new LinkedList<String>();
        q.add("java");
        q.add("python");
        q.add("javascript");
```

```
        q.add("html");
        System.out.println(q.poll());
        System.out.println(q);
        System.out.println(q.peek());
        System.out.println(q);

    }
}
output:

java

[python, javascript, html]

python

[python, javascript, html]
```

==========

```
package com.torryharris.mainpack;

import java.util.*;

public class Main {

    public static void main(String[] args) {
   // write your code here
        /*List<String> l1 = new ArrayList<String>();
        l1.add("varun");
        l1.add("varun");
        l1.add("suhas");
        l1.add("rcb");
        l1.add("ajay");
        l1.add("abhi");
        l1.add(1,"noone");
        l1.remove(2);
        System.out.println(l1);
        List<String> sl2 = new ArrayList<String>();
        sl2.add("one");
        sl2.add("two");
        l1.addAll(1,sl2);
        System.out.println(l1);
        for(int i=0;i<l1.size();i++)
        {
            System.out.print(l1.get(i)+"  ");
        }
        System.out.println();
        for(String s:l1)
        {
            System.out.print(s+"    ");
        }
        System.out.println();
```

```
        Iterator<String> it = l1.iterator();
        while(it.hasNext())
        {
            System.out.print(it.next().toUpperCase()+"       ");
        }
        System.out.println();*/
        Scanner sc = new Scanner(System.in);
        ArrayList<Integer> l2 = new ArrayList<Integer>();
        System.out.println("enter size of array");
        int n = sc.nextInt();
        for(int i=0;i<n;i++)
        {
            System.out.println("enter the elements: ");
            int ele = sc.nextInt();
            l2.add(ele);
        }
        int sum=0;
        Iterator<Integer> it1 = l2.iterator();
        while (it1.hasNext())
        {
            int ele1= it1.next();
            System.out.print(ele1+"  ");
            sum+=ele1;
        }
        System.out.println();
        System.out.println("sum of the element: "+sum);


    }
}
```

output:

```
enter size of array

5

enter the elements:

1

enter the elements:

2

enter the elements:

3

enter the elements:

4

enter the elements:

5

1  2  3  4  5
```

```
sum of the element: 15

================================================================================
```

```
package com.torryharris.mainpack;

import java.util.Collection;
import java.util.Collections;
import java.util.LinkedList;
import java.util.ListIterator;

public class Main {

    public static void main(String[] args) {
   // write your code here
        LinkedList<String> ll1 = new LinkedList<String >();
        ll1.add("orrange");
        ll1.add("apple");
        ll1.add("banana");
        ll1.add("pineapple");
        System.out.println("is empty?: "+ll1.isEmpty());
        System.out.println("contains apple?: "+ll1.contains("apple"));
        ListIterator<String> it = ll1.listIterator();
        System.out.println("in forward direction");
        while(it.hasNext())
        {
            System.out.print(it.next()+"  ");
        }
        System.out.println();
        System.out.println("in backword direction");
        while(it.hasPrevious())
        {
            System.out.print(it.previous()+"  ");
        }
        System.out.println();
        Collections.reverse(ll1);
        System.out.println("reversed list: "+ll1);
        Collections.sort(ll1);
        System.out.println("sorted list: "+ll1);
        Collections.shuffle(ll1);
        System.out.println("shuffled list :"+ll1);
    }
}
```

```
output:

is empty?: false

contains apple?: true

in forward direction

orrange  apple  banana  pineapple

in backword direction

pineapple  banana  apple  orrange
```

```
reversed list: [pineapple, banana, apple, orrange]

sorted list: [apple, banana, orrange, pineapple]

shuffled list :[apple, banana, pineapple, orrange]

================================================================================
```

```
package com.torryharris.mainpack;

import java.util.Enumeration;
import java.util.Iterator;
import java.util.Vector;

public class Main {

    public static void main(String[] args) {
   // write your code here
        Vector<String> v = new Vector<String>();
        v.add("rcb");
        v.add("abd");
        v.add("vk");
        v.add("ddp");
        v.add("maxi");
        Enumeration<String> e = v.elements();
        while(e.hasMoreElements())
        {
            System.out.print(e.nextElement()+"  ");
        }
        System.out.println();
        Iterator<String> it = v.iterator();
        while(it.hasNext())
        {
            System.out.print(it.next()+"  ");
        }
    }
}
output:

rcb  abd  vk  ddp  maxi

rcb  abd  vk  ddp  maxi

================================================================================
```

```
package com.torryharris.mainpack;

import java.util.*;

public class Main {

    public static void main(String[] args) {
   // write your code here
```

```
        Map<Integer,String> hm = new HashMap<Integer,String>();
        hm.put(500,"books");
        hm.put(200,"papers");
        hm.put(100,"pens");
        hm.put(null,null);
        hm.put(300,null);
        hm.put(400,null);
        System.out.println(hm);

        Map<Integer,String> lm = new LinkedHashMap<Integer,String>();
        lm.put(500,"books");
        lm.put(200,"papers");
        lm.put(100,"pens");
        lm.put(null,null);
        lm.put(300,null);
        lm.put(400,null);
        System.out.println(lm);

        Map<Integer,String> tm = new TreeMap<Integer,String>();
        tm.put(500,"books");
        tm.put(200,"papers");
        tm.put(100,"pens");
        tm.put(300,null);
        tm.put(400,null);
        System.out.println(tm);

        Map<Integer,String> ht = new Hashtable<Integer,String>();
        ht.put(500,"books");
        ht.put(200,"papers");
        ht.put(100,"pens");
        ht.put(300,"cars");
        ht.put(400,"bikes");
        System.out.println(ht);



    }
}
output:

{null=null, 400=null, 500=books, 100=pens, 200=papers, 300=null}

{500=books, 200=papers, 100=pens, null=null, 300=null, 400=null}

{100=pens, 200=papers, 300=null, 400=null, 500=books}

{500=books, 400=bikes, 300=cars, 200=papers, 100=pens}

================================================================================

package com.torryharris.mainpack;

import com.torryharris.custpack.customer;
```

```java
import java.util.HashMap;
import java.util.Map;

public class Main {

    public static void main(String[] args) {
   // write your code here
        Map<String,Integer> hm = new HashMap<String,Integer>();
        hm.put("sofa",5000);
        hm.put("table",3000);
        hm.put("lamp",1000);
        System.out.println(hm.values());
        System.out.println(hm.keySet());
        System.out.println(hm.entrySet());
        System.out.println("printing the map elements using for loop");
        for(String key:hm.keySet())
        {
            System.out.println(key+" - "+hm.get(key));
        }
        int totamt=0;
        for(String key:hm.keySet())
        {
            totamt+=hm.get(key);
        }
        System.out.println("total investment RS "+totamt);
        HashMap<customer,Integer> c1 = new HashMap<customer,Integer>();
        c1.put(new customer(10,"varun"),5000);
        c1.put(new customer(20,"suhas"),15000);
        c1.put(new customer(30,"ajay"),10000);
        for(customer cust:c1.keySet())
        {
            System.out.println(cust+" "+c1.get(cust));
        }
        int totalsalesamt=0;
        for(customer cust:c1.keySet())
        {
            totalsalesamt+=c1.get(cust);
        }
        System.out.println("total sales amt RS "+totalsalesamt);
    }
}
```

output:

```
[5000, 1000, 3000]

[sofa, lamp, table]

[sofa=5000, lamp=1000, table=3000]

printing the map elements using for loop

sofa - 5000

lamp - 1000
```

```
table - 3000
total investment RS 9000
customer{id=20, name='suhas'} 15000
customer{id=10, name='varun'} 5000
customer{id=30, name='ajay'} 10000
total sales amt RS 30000
================================================================================
```