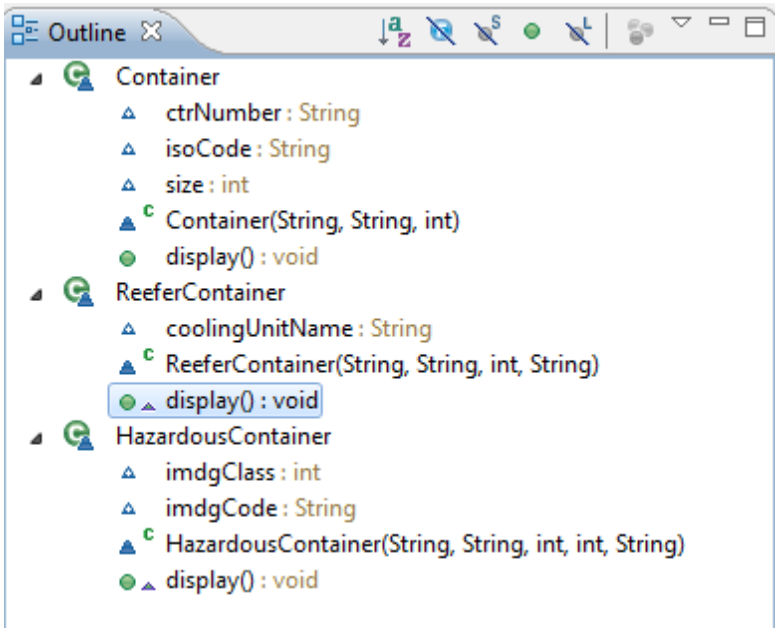
1) A container terminal is a facility where cargo containers are transshipped between different transport vehicles, for onward transportation. The transshipment may be between container ships and land vehicles, for example trains or trucks, in which case the terminal is described as a maritime container terminal. Alternatively the transshipment may be between land vehicles, typically between train and truck, in which case the terminal is described as an inland container terminal.

A terminal handles different types of containers like General purpose containers, Reefer Containers, Hazardous Container etc. Every container should mandatorily have Container number in the format XXXUNNNNNN9 where first three characters are owner code in alphabets, U - can be G - for general purpose, R for Reefer, H for Hazardous, 5th to 10th characters is serial number which is unique. Containers will have other attributes like ISO Code (String), Size (vaules can be 20, 40 or 45). Reefer Containers will have a cooling unit (String) where as Hazardous containers will have IMDG Class (Number), Code (String).

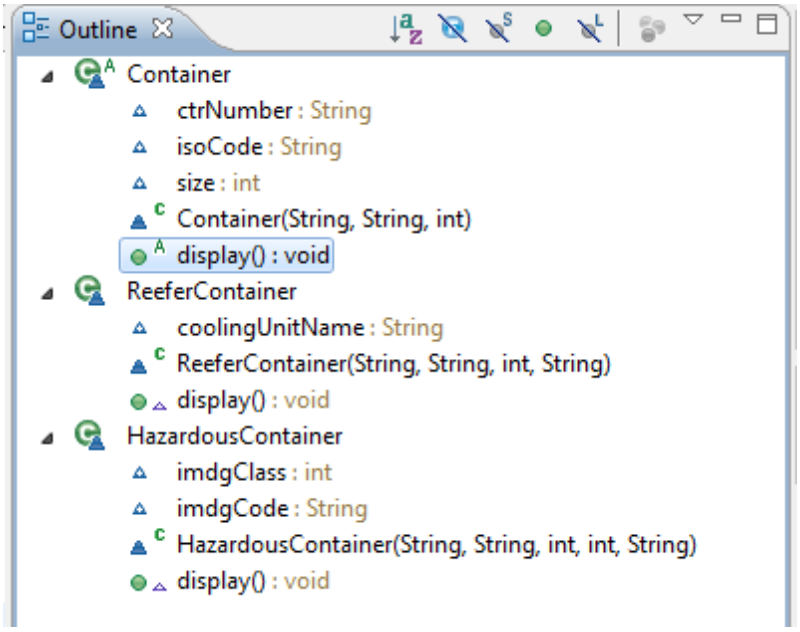
a) Identify the classes in the above scenario
b) Create classes to implement inheritance concept.
c) Add a method display to print the details of containers depending on the type.
d) Write a program to implement Polymorphism concept, Dynamic method dispatch concept.



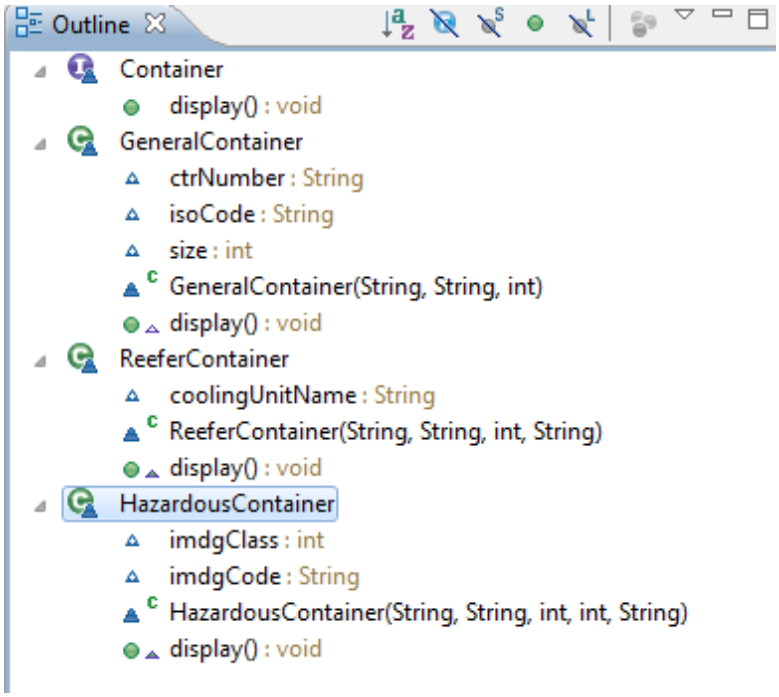
```
1
2  class Container {
3
4      /**
5       * @param args
6       */
7      String ctrNumber;
8      String isoCode;
9      int     size;
10
11     Container(String ctrNumber, String isoCode, int size){
12         this.ctrNumber = ctrNumber;
13         this.isoCode = isoCode;
14         this.size = size;
15     }
16
17     public void display(){
18         System.out.println("Container Number:"+ctrNumber+" ISO Code:"+isoCode+" Size:"+size);
19         }
20 }
21
22 class ReeferContainer extends Container {
23
24     String coolingUnitName;
25
26     ReeferContainer(String ctrNumber, String isoCode, int size, String coolingUnitName){
27         super(ctrNumber, isoCode, size);
28         this.coolingUnitName=coolingUnitName;
29     }
30
31     public void display(){
32         System.out.println("======================");
33         System.out.println("Reefer Container Details:");
34         super.display();
```

```
35          System.out.println(" Cooling Unit Name:"+coolingUnitName);
36      }
37  }
38
39  class HazardousContainer extends Container {
40      int imdgClass;
41      String imdgCode;
42
43      HazardousContainer(String ctrNumber, String isoCode, int size, int imdgClass, String imdgCode){
44          super(ctrNumber, isoCode, size);
45          this.imdgClass=imdgClass;
46          this.imdgCode=imdgCode;
47      }
48
49      public void display(){
50          System.out.println("=======================");
51          System.out.println("Hazardous Container Details:");
52          super.display();
53          System.out.println(" Imdg Class:"+imdgClass+" ImdgCode:"+imdgCode);
54      }
55  }
56
57  class ContainerDetails{
58      public static void main(String[] args){
59          Container c = new Container("OOCG2121219","20TO",20);
60          c.display();
61
62          ReeferContainer r = new ReeferContainer("ONLR4534349","40RH", 40,"FREEZER");
63          r.display();
64
65          HazardousContainer h = new HazardousContainer("PONH8743549","45EU", 45,123,"DAN");
66          h.display();
67      }
68  }
```

2) In the assignment 1 above, Make the Container class as abstract and implement Dynamic method dispatch



3) In the assignment 1 above, Make the container as an interface and implement Dynamic method dispatch



4) Write a program to use super, instance of keywords in an appropriate way.