```
//Client

public class aTree {
int xCoord;           //extrinsic
int yCoord;           //extrinsic
TreeType theTreeType;  //intrinsic

public aTree(int x, int y, String aTreeType){
      xCoord = x;
      yCoord = y;
      theTreeType = TreeTypeFactory.getTreeType(aTreeType);
//reference to flyweight
}

public void displayTree(){
      System.out.println(theTreeType.displayTreeType());
      System.out.println("@ x coordinate = " + xCoord);
      System.out.println("@ y coordinate = " + yCoord);

}

}
```

```
import java.util.*;

public class ForestDriver {


	public static void main(String[] args) {

		TreeTypeFactory factory = new TreeTypeFactory();
		List<aTree> theForest = new ArrayList<aTree>();

		for(int x = 1; x < 10; ++x){
			for(int y = 1; y < 10; ++y){
				theForest.add(new aTree(x, y, "PineTree"));
				theForest.add(new aTree(x+10, y+10, "RedWood"));
			}
		}

		for(aTree a: theForest){
			a.displayTree();
			System.out.println("\n");
		}
	}

}
```

```
//Flyweight


public class PineTree implements TreeType {

	/*intrinsic states*/
	private String treeName;
	private String barkColor;
	private String leafType;
	private String leafColor;

	public PineTree(){
		treeName = "Pine Tree";
		barkColor = "Brown";
		leafType = "Ever Green";
		leafColor = "Green";
	}

	public int grow(int anAge, int growRate){
		return anAge*growRate;
	}

	public String displayTreeType(){

		return "Tree Name: " + treeName +
			"\nBark Color: " + barkColor +
			"\nLeaf Type: " + leafType +
			"\nLeaf Color: " + leafColor;
	}


}
```

```
//FlyWeight


public class RedWood implements TreeType {

	/*intrinsic states*/
	private String treeName;
	private String barkColor;
	private String leafType;
	private String leafColor;

	public RedWood(){
		treeName = "Red Wood";
		barkColor = "Redish Brown";
		leafType = "Compound";
		leafColor = "Green";
	}

	public int grow(int anAge, int growRate){
		return anAge*growRate;
	}

	public String displayTreeType(){

		return "Tree Name: " + treeName +
			"\nBark Color: " + barkColor +
			"\nLeaf Type: " + leafType +
			"\nLeaf Color: " + leafColor;
	}


}
```

```
//FlyWeight Interface

public interface TreeType {

      /*manage extrinsic states*/
      public int grow(int anAge, int aHeight);
      public String displayTreeType();

}
```

```
//FlyWight Factory

public class TreeTypeFactory {
      static TreeType aPineTree;
      static TreeType aRedWood;

      public TreeTypeFactory(){
            aPineTree = new PineTree();
            aRedWood = new RedWood();
      }

      /*Returns the same instance*/
      public static TreeType getTreeType(String aTreeType){
            if (aTreeType.equals("PineTree"))
                  return aPineTree;
            else
                  return aRedWood;
      }



}
```