```
// interface shape{
//     double pi=3.14; //public, static, final
//     double getSquare(double r);//public and abstract
// }
// class circle implements shape{
//     public double getSquare(double r){
//         return r*r;
//     }
//     void fun(){
//         System.out.println("hey, I implemented interface shape ");
//     }
// }

// public class lec_12 {
//     public static void main(String args[]){
//         circle c= new circle();
//         System.out.println(c.getSquare(4));
//         c.fun();
//     }
// }
// --------------------------------

// Interface allows multiple Inheritence
interface m{
    int x=20;
    void fun();
}
interface n{
    int x=40;
    void fun();
}
class c implements m,n{
    public void fun(){
        System.out.println("Hello, I'm function class c");
        System.out.println(m.x); // ambiguity resloved using fully qualified name
    }
}

interface O extends m,n{ //multiple inheritence through interfaces
    void fun();
}

class d implements O{
    public void fun(){
        System.out.println("Hello, I'm function class d");
    }
}
```

```
public class lec12 {
    public static void main(String args[]){
        c c1=new c();
        c1.fun();
        d d1=new d();
        d1.fun();
    }
}
```