1. What is an interface in Java?

*Interface is a keyword ,it is a blueprint of a class that have set of methods and other class must implement

*It does not have method implements ,the classes that implements it have method declarations

*Multiple inheritance is achieved through this interface

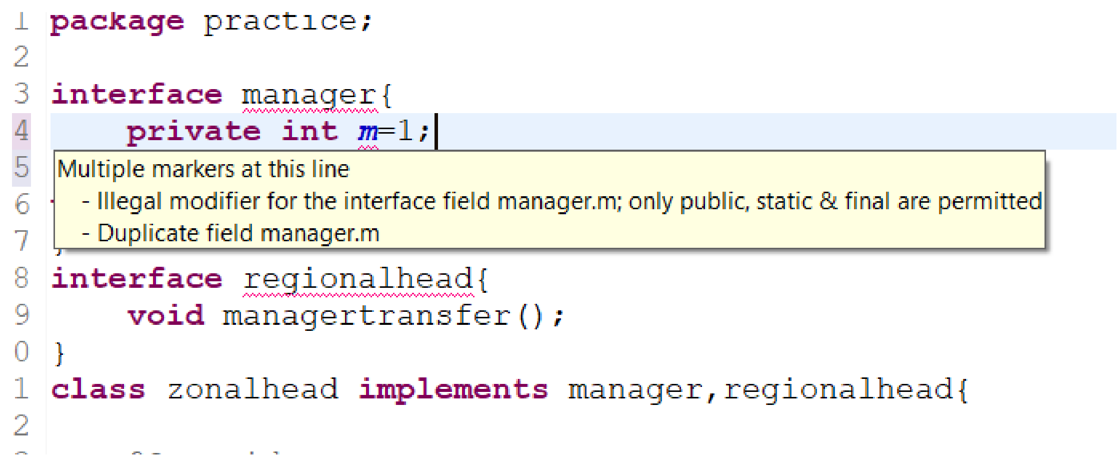
Program:

```
package practice;

interface manager{
void employeetransfer();
}
interface regionalhead{
	void managertransfer();
}
class zonalhead implements manager,regionalhead{

	@Override
	public void managertransfer() {
		System.out.println("poonamalee");

	}

	@Override
	public void employeetransfer() {
		System.out.println("porur");

	}

}
public class interfaceforexample {
public static void main(String[] args) {
	zonalhead z=new zonalhead();
	z.managertransfer();
	z.employeetransfer();
}
}
```

2. Can we define private and protected modifiers for data members (fields) in interfaces?

We cannot define the private and protected modifiers for data members (fields) in interfaces

program:



```
package practice;

interface manager{
    private int m=1;
```

Multiple markers at this line
- Illegal modifier for the interface field manager.m; only public, static & final are permitted
- Duplicate field manager.m

```
interface regionalhead{
    void managertransfer();
}
class zonalhead implements manager,regionalhead{
```

3. Which modifiers are allowed for methods in an Interface?

   Public,default,abstract,static these are the modifiers are allowed in interface

   Program:

```
package practice;

interface manager{

    abstract void employeetransfer();
     static void loanhead(){
          System.out.println("static method");
     }
}
interface regionalhead{
     void managertransfer();
     default void salesheadtrans() {
          System.out.println("we can define body by using defualt method");
     }

}
class zonalhead implements manager,regionalhead{

     @Override
     public void managertransfer() {
          System.out.println("poonamalee");

     }

     @Override
     public void employeetransfer() {
          System.out.println("porur");

     }

}
public class interfaceforexample {
public static void main(String[] args) {
     zonalhead z=new zonalhead();
     z.managertransfer();
     z.employeetransfer();
     z.salesheadtrans();

}
}
```

4. Suppose A is an interface. Can we create an object using new A()?

    No we cannot create an object for interface because it only accessed through the implemented classes

Program:

```
interface manager{
void employeetransfer();
}
interface regionalhead{
     void managertransfer();
}
class zonalhead implements manager,regionalhead{

     @Override
     public void managertransfer() {
          System.out.println("poonamalee");

     }

     @Override
     public void employeetransfer() {
          System.out.println("porur");

     }

}
public class interfaceforexample {
public static void main(String[] args) {
     zonalhead z=new zonalhead();
     z.managertransfer();
     z.employeetransfer();
}
}
```

5. Can we define an interface with a static modifier?

   *yes we can define it with a static modifier

   *it is the instance of a class that implements the interface

   Program:

```
interface manager{

 void employeetransfer();
     static void loandetails() {
          System.out.println("loan payments");
     }
```

6. Suppose A is an interface. Can we declare a reference variable a with type A like this: A a;

we can declare a reference variable for an interface .

program:

```
 package practice;
interface A {
    void someMethod();
}

class b implements A {
    @Override
    public void someMethod() {
        System.out.println("Implementation of someMethod");
    }
}
```

```
public class referanceinterface {
      public static void main(String[] args) {
            A a; // Declaring a reference variable
             a = new b(); // Assigning an object of the b class to the reference
variable
             a.someMethod(); // Calling the method using the object through the
interface reference
         }
}
```

7.Can an interface extends another interface in Java?

Yes ,interface extends another interface

Program:

```
package practice;
interface A {
    void details();
    default void loan() {
     System.out.println("loan");
    }
}

interface B extends A {
    void creditcard();
}
class c implements A,B
```

```
{

      @Override
      public void creditcard() {
            // TODO Auto-generated method stub

      }

      @Override
      public void details() {
            // TODO Auto-generated method stub

      }

}
public class interextendsinter {
public static void main(String[] args) {
      A a;
      a=new c();
      a.details();
      a.loan();


}
}
```

8. Can an interface implement another interface?

The interface cannot implement another interface because the interface need classes to implement the methods of them

Program:

```
interface manager{
      void emtranfer();
}
interface rh{
      void bmtransfer();
      default void accountdetails(){
            System.out.println("account details of the customers ");
      }
}
class sthead implements manager,rh{
      @Override
      public void emtranfer() {
            // TODO Auto-generated method stub
            System.out.println("* Bm has rights to transfer employee *");
      }
      @Override
      public void bmtransfer() {
            // TODO Auto-generated method stub
            System.out.println("* Regoinal head has rights to tranfer bm *");
      }
```

9. Is it possible to define a class inside an interface?

No it is not possible to define a class inside an interface ,it has only methods ,modifiers inside

Program:

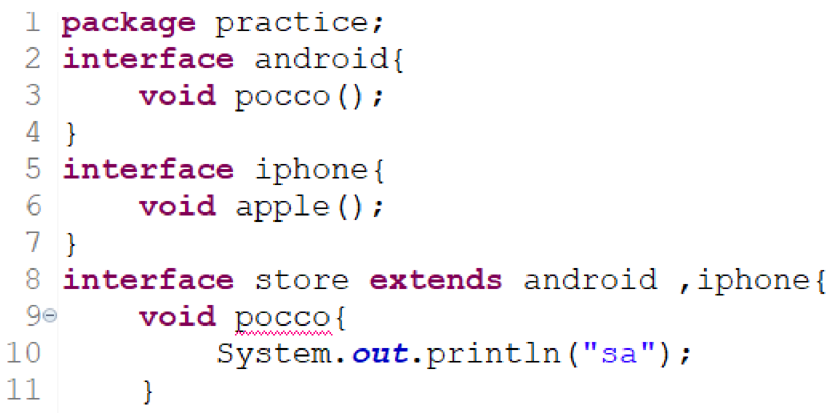
```
 interface manager{
     void emtranfer();
}
interface rh{
     void bmtransfer();
     default void accountdetails(){
          System.out.println("account details of the customers ");
     }
}
class sthead implements manager,rh{
     @Override
     public void emtranfer() {
          // TODO Auto-generated method stub
          System.out.println("* Bm has rights to transfer employee *");
     }
     @Override
     public void bmtransfer() {
          // TODO Auto-generated method stub
          System.out.println("* Regoinal head has rights to tranfer bm *");
```

13. Can an interface extend multiple interfaces?

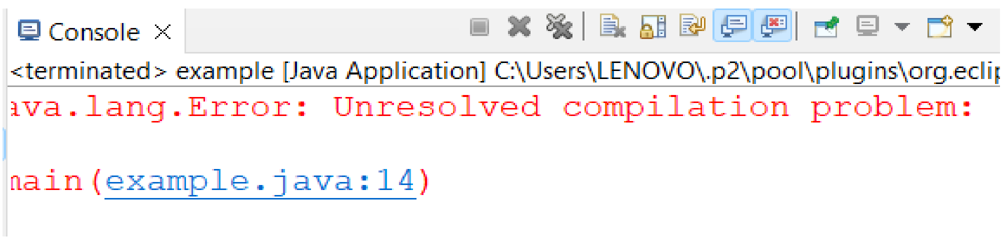
No it cannot extend multiple interfaces

But a class implements multiple interfaces as methods definitions

Program:



```
package practice;
interface android{
    void pocco();
}
interface iphone{
    void apple();
}
interface store extends android ,iphone{
    void pocco{
        System.out.println("sa");
    }
```



```
<terminated> example [Java Application] C:\Users\LENOVO\.p2\pool\plugins\org.eclip
ava.lang.Error: Unresolved compilation problem:

nain(example.java:14)
```