

Day 8

📅 Created	@Mar 17, 2021
👤 Created by	ⓧ VIGNESH J
🏷️ Tags	Java

Complexity

Transparency

Binding

Static Binding

Dynamic Binding

- Create object at runtime
- Based on need

Composition

- Static
 - Compile time
 - Abstract class
 - Achieved through generalization
 - Happens during construction time
 - Object has another object
 - Should be done in constructor
 -

```
package ga.veee.day8;

public class DecoratorDemo {
    public static void main(String[] args) {
        Food thali = new Rice(new Daal(new Chicken(new Rice())));
        System.out.println("Cost of Rice..+Daal..+chicken+rice.." + thali.cost());
    }
}

//static composition - static composition is achieved through generalization - part whole hierarchy
abstract class Food {
```

```

    public abstract int cost();
}

abstract class VegFood extends Food {
}

abstract class NonVegFood extends Food {
}

//To do static composition we need to use the constructor
class Rice extends VegFood {
    public Rice() {
        // TODO Auto-generated constructor stub
    }

    private Food food;

    public Rice(Food food) {
        this.food = food;
    }

    @Override
    public int cost() {
        // TODO Auto-generated method stub
        if (food != null) {
            return 10 + food.cost();
        } else {
            return 10;
        }
    }
}

class Daal extends VegFood {
    public Daal() {
        // TODO Auto-generated constructor stub
    }

    private Food food;

    public Daal(Food food) {
        this.food = food;
    }

    @Override
    public int cost() {
        // TODO Auto-generated method stub
        if (food != null) {
            return 5 + food.cost();
        } else {
            return 5;
        }
    }
}

class Chicken extends NonVegFood {
    public Chicken() {
        // TODO Auto-generated constructor stub
    }
}

```

```

    private Food food;

    public Chicken(Food food) {
        this.food = food;
    }

    @Override
    public int cost() {
        // TODO Auto-generated method stub
        if (food != null) {
            return 20 + food.cost();
        } else {
            return 20;
        }
    }
}

```

- Dynamic
 - Runtime
 - Through Interface
 - Quickly Create Object
- Reflection
 - Introspection
 - Field
 - Method
 -

```

package ga.veee.day8;

import java.lang.reflect.Field;
import java.lang.reflect.Method;

public class ReflrctionDemo {
    public static void main(String[] args) throws Exception {
        PoliceStation p1 = new PoliceStation();
        Politician ramu = new Politician();
        Doctor prakash = new Doctor();
        p1.arrest(ramu);
    }
}

class PoliceStation {
    public void arrest(Object p) throws Exception {
        //Politician pol=(Politician)p;
        //System.out.println(pol.name);
        //pol.doWork();
        //interrogation
    }
}

```

```

        Class c = p.getClass();
        Field f = c.getField("name");
        System.out.println(f.get(p));

        Method m = c.getMethod("doWork");
        m.invoke(p);

        //torture cell
        f = c.getDeclaredField("secretName");
        f.setAccessible(true);
        System.out.println(f.get(p));

        m = c.getDeclaredMethod("secretWork");
        m.setAccessible(true);
        m.invoke(p);
    }
}

class Politician {
    public String name = "good man";
    private String secretName = "thief";

    public void doWork() {
        System.out.println("I do social work.....");
    }


    private void secretWork() {
        System.out.println("I trouble people and steal there money...");
    }
}

class Doctor {
}

```

Best Free Online Video Recording & Screen Capture Software | Fluide

Fluide is the best all-in-one online screen capture & video recording software that is available for free. Fluide helps you record, edit, communicate & share your video messages in the most simplest way.

 <https://fluide.com/videos/detail/G6x-YcXGqkU5dkV1r#.YFGnMjvVdJk.link>

```

package ga.veee.day8;

import java.lang.reflect.Field;
import java.lang.reflect.Method;

public class ReflectionDemo2 {
    public static void main(String[] args) throws Exception {
        //Dynamic object creation... this can be used as a replacement for NEW keyword
        //new keyword is static and Class.forName(...) is dynamic
        Demo d = (Demo) Class.forName("day8.Demo").getConstructor().newInstance();
        String result = d.test("ramu", 30);
        System.out.println("The result is.." + result);
    }
}

```

```

d = (Demo) Class.forName("day8.Demo").getConstructor(String.class).newInstance("hello");

Class.forName("day8.Demo").getConstructor(int.class).newInstance(100);

Field f[] = d.getClass().getFields();
for (Field field : f) {
    System.out.println(field.getName());
}
Method m[] = d.getClass().getMethods();
for (Method method : m) {
    System.out.println(method.getName());
}

//Method testMethod=d.getClass().getMethod("test",new Class[] {String.class,Integer.class});
//result=(String) testMethod.invoke(d, new Object[] {"ramu",Integer.valueOf(20)});
Method testMethod = d.getClass().getMethod("test", String.class, int.class);
result = (String) testMethod.invoke(d, "ramu", 20);
System.out.println("The result is.." + result);

Method parentMethod = d.getClass().getSuperclass().getDeclaredMethod("parent");
parentMethod.setAccessible(true);
parentMethod.invoke(d);
}
}

class DemoParent {
    private void parent() {
        System.out.println("parent method....");
    }
}

class Demo extends DemoParent {
    public String f1, f2, f3;

    public Demo() {
        System.out.println("Constructor without variable called...");
    }

    public Demo(String s) {
        System.out.println("Constructor with variable called..." + s);
    }


    public Demo(int i) {
        System.out.println("Constructor with int variable called...." + i);
    }

    public String test(String name, int age) {
        return "The name is..." + name + " and age is.." + age;
    }
}

```

Best Free Online Video Recording & Screen Capture Software | Fluvuid

Fluvuid is the best all-in-one online screen capture & video recording software that is available for free. Fluvuid helps you record, edit, communicate & share your video messages in the most simplest way.

 <https://fluvuid.com/videos/detail/ka7dEh85n5TOADQ3m#.YFGwSmoJjvk.link>

Interfaces

- Are primarily special classes
- Object is subject to the implementation
- Implementation
- Subjection
- Action Noun
- Must be implemented somewhere
-

```
package ga.veee.day8;

import java.lang.reflect.InvocationHandler;
import java.lang.reflect.Method;
import java.lang.reflect.Proxy;
import java.util.Scanner;

public class InterfaceDemo {
    public static void main(String[] args) {
        AlopathyMedicalCollege stanley = new AlopathyMedicalCollege();
        AyurvedMedicalCollege imcops = new AyurvedMedicalCollege();

        Human shoiab = new Human();//1974

        Doctor doctorShoiab = (Doctor) Proxy.newProxyInstance(shoiab.getClass().getClassLoader(),
            new Class[]{Doctor.class},
            new MyInvocationHandler(imcops));
        doctorShoiab.doCure();
    }
}

class MyInvocationHandler implements InvocationHandler {
    Doctor doctor;

    public MyInvocationHandler(Doctor doctor) {
        this.doctor = doctor;
    }

    @Override
    public Object invoke(Object proxy, Method method, Object[] args) throws Throwable {
        // TODO Auto-generated method stub
        System.out.println("method invoke invoked...");
        return method.invoke(doctor);
    }
}

interface Doctor {
    public void doCure();
}

class Human {
}
```

```


class AlopathyMedicalCollege implements Doctor {
    @Override
    public void doCure() {
        System.out.println("alopathy cure logic implemented...");
    }
}

class AyurvedMedicalCollege implements Doctor {
    @Override
    public void doCure() {
        System.out.println("ayurved cure logic implemented...");
    }
}

```

Best Free Online Video Recording & Screen Capture Software | Fluvid

Fluvid is the best all-in-one online screen capture & video recording software that is available for free. Fluvid helps you record, edit, communicate & share your video messages in the most simplest way.

 <https://fluvid.com/videos/detail/e-KdETk4M7U5qPZpv#.YFHxUL8cpM0.link>

```

package ga.veee.day8;

import java.lang.reflect.InvocationHandler;
import java.lang.reflect.Method;
import java.lang.reflect.Proxy;
import java.util.Scanner;

public class InterfaceDemo2 {
    public static void main(String[] args) {
        AlopathyMedicalCollege0 stanley = new AlopathyMedicalCollege0();
        AyurvedMedicalCollege0 imcops = new AyurvedMedicalCollege0();

        JetAcademy jet = new JetAcademy();

        Object shoiab = new Human0();//1974

        shoiab = Proxy.newProxyInstance(shoiab.getClass().getClassLoader(),
            new Class[]{Doctor0.class, Pilot.class},
            new MyInvocationHandler0(new Object[]{shoiab, jet, stanley}));

        Doctor0 doctorShoiab = (Doctor0) shoiab;
        doctorShoiab.doCure();

        Pilot pilot = (Pilot) shoiab;
        pilot.doFly();
        Scanner scan = new Scanner(System.in);
        System.out.println("waiting.....");
        String s = scan.next();

        StewardAcademy sa = new StewardAcademy();
        shoiab = Proxy.newProxyInstance(shoiab.getClass().getClassLoader(),
            new Class[]{Doctor0.class, Pilot.class, Steward.class},
            new MyInvocationHandler0(new Object[]{shoiab, sa}));
    }
}

```

```

        Steward stewardShoiab = (Steward) shoiab;
        stewardShoiab.serve();

        doctorShoiab = (Doctor0) shoiab;
        doctorShoiab.doCure();

        pilot = (Pilot) shoiab;
        pilot.doFly();
    }
}

class MyInvocationHandler0 implements InvocationHandler {
    Object obj[];

    public MyInvocationHandler0(Object obj[]) {
        this.obj = obj;
    }

    @Override
    public Object invoke(Object proxy, Method method, Object[] args) throws Throwable {
        Object returnObject = null;
        for (Object o : obj) {
            Method m[] = o.getClass().getDeclaredMethods();
            for (Method met : m) {
                if (met.getName().equals(method.getName())) {
                    met.setAccessible(true);
                    returnObject = method.invoke(o, args);
                }
            }
        }
        return returnObject;
    }
}

interface Steward {
    public void serve();
}

interface Pilot {
    public void doFly();
}

class StewardAcademy implements Steward {
    @Override
    public void serve() {
        System.out.println("serving.....");
    }
}

class JetAcademy implements Pilot {
    @Override
    public void doFly() {
        System.out.println("flying.....");
    }
}

interface Doctor0 {
    public void doCure();
}


```



```
class Human0 {  
}  
  
class AlopathyMedicalCollege0 implements Doctor0 {  
    @Override  
    public void doCure() {  
        System.out.println("alopathy cure logic implemented...");  
    }  
}  
  
class AyurvedMedicalCollege0 implements Doctor0 {  
    @Override  
    public void doCure() {  
        System.out.println("ayurved cure logic implemented...");  
    }  
}
```

Best Free Online Video Recording & Screen Capture Software | Fluvid

Fluvid is the best all-in-one online screen capture & video recording software that is available for free. Fluvid helps you record, edit, communicate & share your video messages in the most simplest way.

 <https://fluvid.com/videos/detail/e-KdETk4XES5qPQjd#.YFHw6wi31Cl.link>