Strategy

Sim VDuck

Duck → Swim()
→ display() → abstract
quack()
fly()

Mallard Duck   RedHead Duck   Rubber Duck   Wooden Duck

∝
Quack
Squeak

Quack
/ ↓ ⟶
quack Squah Mute

quack()
{
Squeah
}

quack(){
Mute
}

IflyBehavior

fly()

Duck

IflyBehavior objf;
IQuakBehavior objq;

- swimm()
- display() → abs
- performfly()
- perform Quak

IQuakBehavior

quak()

FlyWithWings

fly() { }

flyNoWay

fly() { }

Quack

quak() { }

Squeek

quak() { }

Mute

quak() { }

Mallard Duck

display() { }

RedHead Duck

display() { }

WoodenDuck

display() { }

Rubber Duck

display() { }

```java
package StrategyPattern;

public interface FlyBehavior {
    public void fly();
}
```

```java
package StrategyPattern;

public class FlyNoWay implements FlyBehavior{

    @Override
    public void fly() {
        // TODO Auto-generated method stub
        System.out.println("Can't fly");
    }

}
```

```java
public class FlyWithWings implements FlyBehavior{

    @Override
    public void fly() {
        // TODO Auto-generated method stub
        System.out.println("Fly With Wings");
    }

}
```

```java
public abstract class Duck {
    private FlyBehavior flyBehavior;
    private QuackBehavior quackBehavior;

    Duck() {
        this.flyBehavior = new FlyWithWings();
        this.quackBehavior = new Quack();
    }

    Duck(FlyBehavior fb, QuackBehavior qb) {
        this.flyBehavior = fb;
        this.quackBehavior = qb;
    }

    abstract public void display();

    public void swim() {
        System.out.println("I am a duck and I swim");
    }

    public void performFly() {
        this.flyBehavior.fly();
    }

    public void performQuack() {
        this.quackBehavior.quack();
    }

    public void setFlyBehavior(FlyBehavior fb) {
        this.flyBehavior = fb;
    }

    public void setQuackBehavior(QuackBehavior qb) {
        this.quackBehavior = qb;
    }
}
```

```java
public interface QuackBehavior {
    public void quack();
}
```

```java
public class StayMute implements QuackBehavior {

    @Override
    public void quack() {
        // TODO Auto-generated method stub
        System.out.println("I stay Mute");
    }
}
```

```java
public class Quack implements QuackBehavior {

    @Override
    public void quack() {
        // TODO Auto-generated method stub
        System.out.println("Quack Quack");
    }
}
```

```java
public class Squeak implements QuackBehavior {

    @Override
    public void quack() {
        // TODO Auto-generated method stub
        System.out.println("Squeak Squeak");
    }
}
```

```java
package StrategyPattern;

public class MallardDuck extends Duck {

    MallardDuck() {
        super();
    }

    MallardDuck(FlyBehavior fb, QuackBehavior qb) {
        super(fb,qb);
    }

    @Override
    public void display() {
        // TODO Auto-generated method stub
        System.out.println("I am a Mallard Duck");
    }

}
```

I for Testing

```java
package StrategyPattern;

public class Test {

    public static void main(String[] args) {
        // TODO Auto-generated method stub

        MallardDuck mallardDuck = new MallardDuck();
        mallardDuck.performFly();
        mallardDuck.performQuack();

        MallardDuck md2 = new MallardDuck(new FlyNoWay(),new Squeak());
        md2.performFly();
        md2.performQuack();

        MallardDuck md3 = new MallardDuck();
        md3.setFlyBehavior(new FlyWithWings());
        md3.setQuackBehavior(new StayMute());
        md3.performFly();
        md3.performQuack();
    }

}
```