JAVA LAB Assignment 6

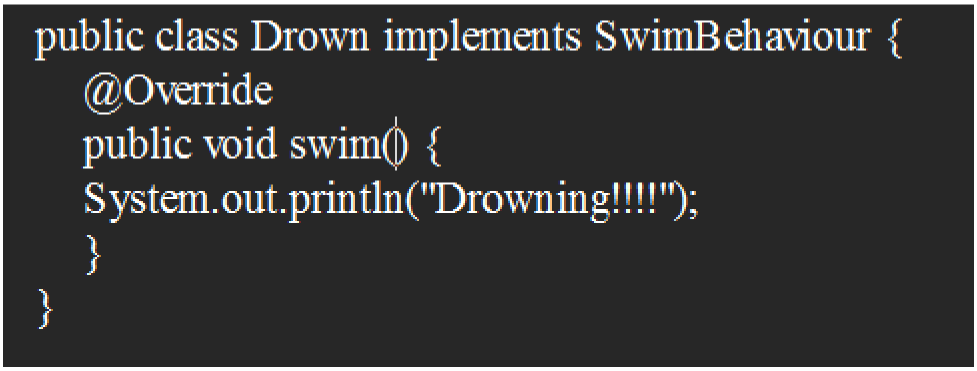
DUCK

Rishi Selam

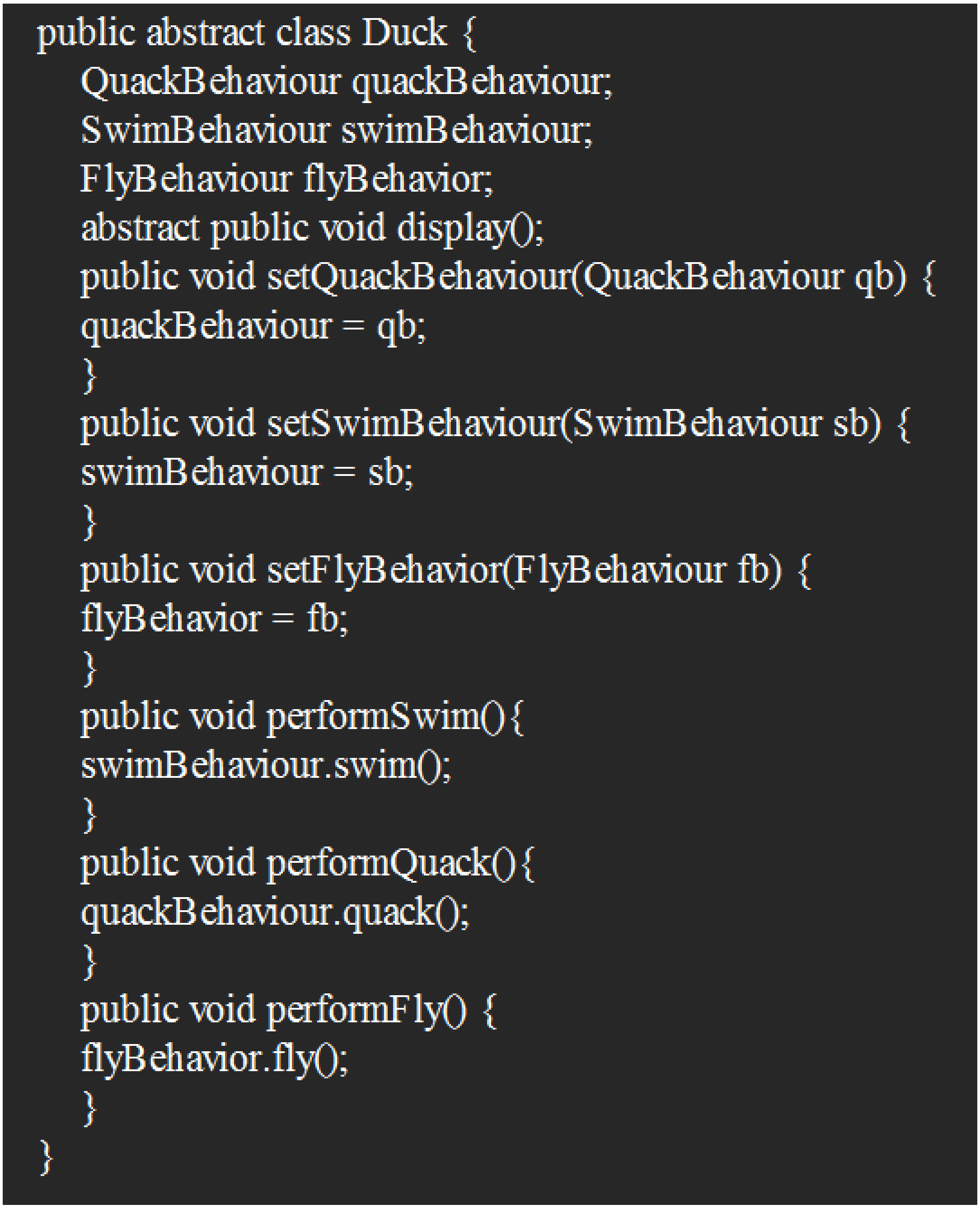
AIML B2

23070126107

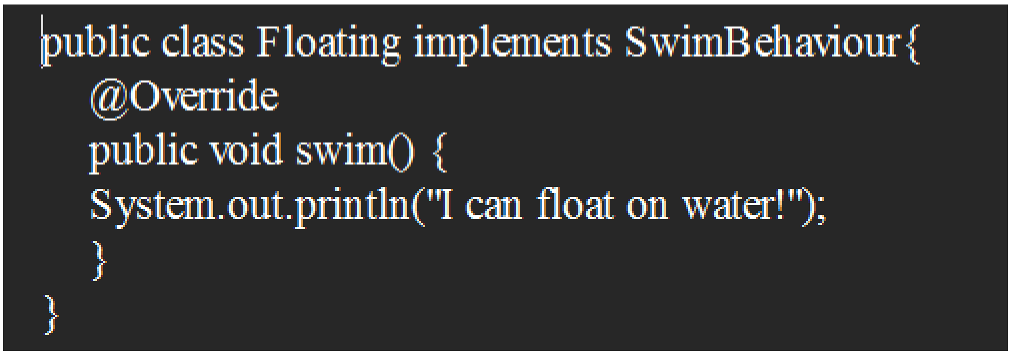
Code:



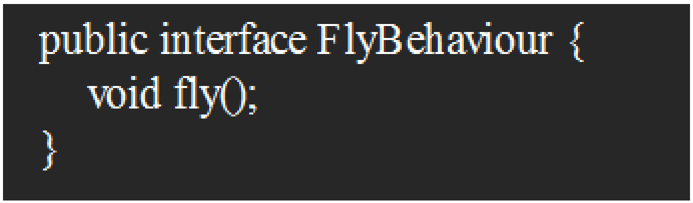
```
public class Drown implements SwimBehaviour {
    @Override
    public void swim() {
    System.out.println("Drowning!!!!");
    }
}
```



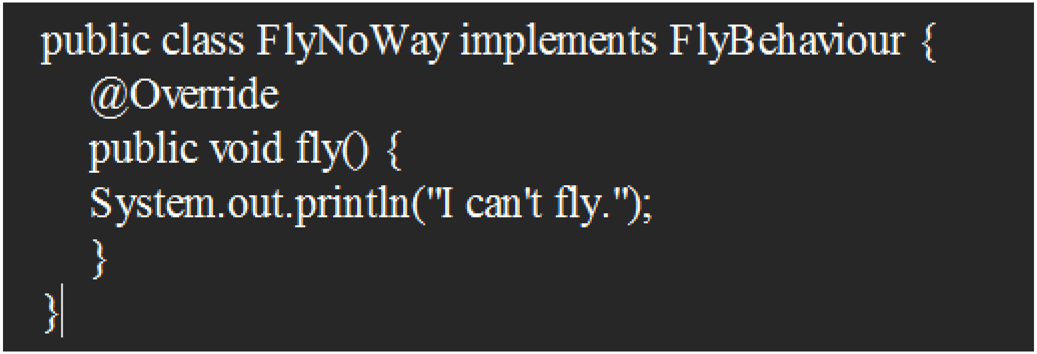
```
public abstract class Duck {
    QuackBehaviour quackBehaviour;
    SwimBehaviour swimBehaviour;
    FlyBehaviour flyBehavior;
    abstract public void display();
    public void setQuackBehaviour(QuackBehaviour qb) {
    quackBehaviour = qb;
    }
    public void setSwimBehaviour(SwimBehaviour sb) {
    swimBehaviour = sb;
    }
    public void setFlyBehavior(FlyBehaviour fb) {
    flyBehavior = fb;
    }
    public void performSwim(){
    swimBehaviour.swim();
    }
    public void performQuack(){
    quackBehaviour.quack();
    }
    public void performFly() {
    flyBehavior.fly();
    }
}
```



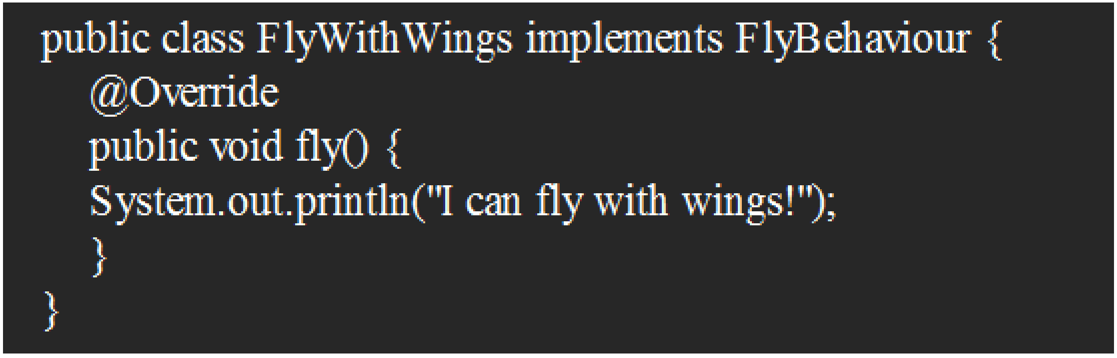
```
public class Floating implements SwimBehaviour{
    @Override
    public void swim() {
    System.out.println("I can float on water!");
    }
}
```



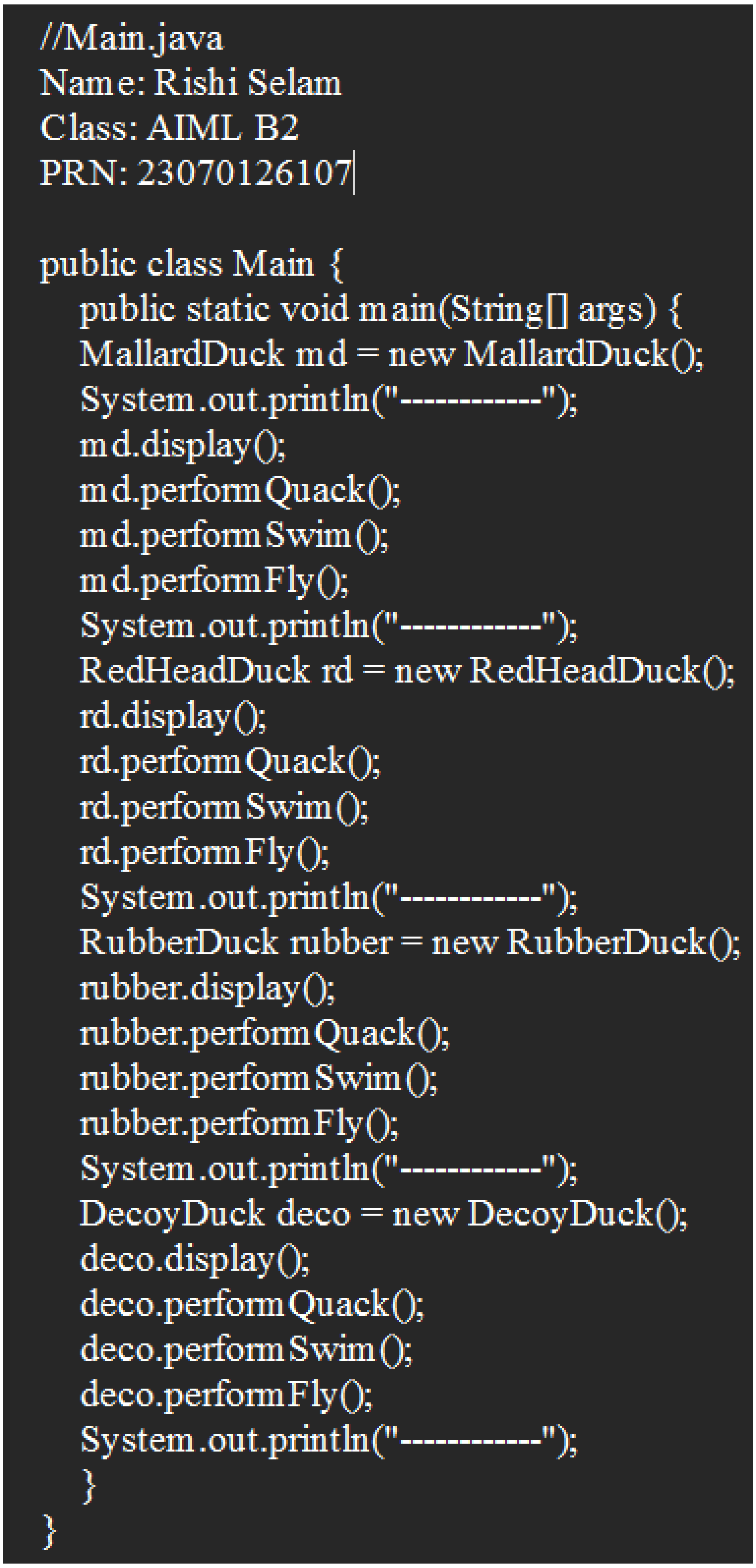
```
public interface FlyBehaviour {
    void fly();
}
```



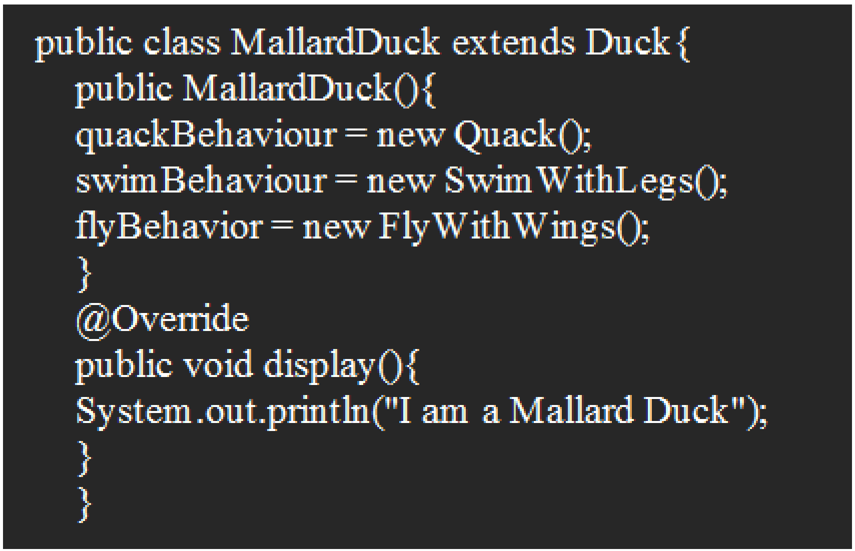
```
public class FlyNoWay implements FlyBehaviour {
    @Override
    public void fly() {
    System.out.println("I can't fly.");
    }
}
```



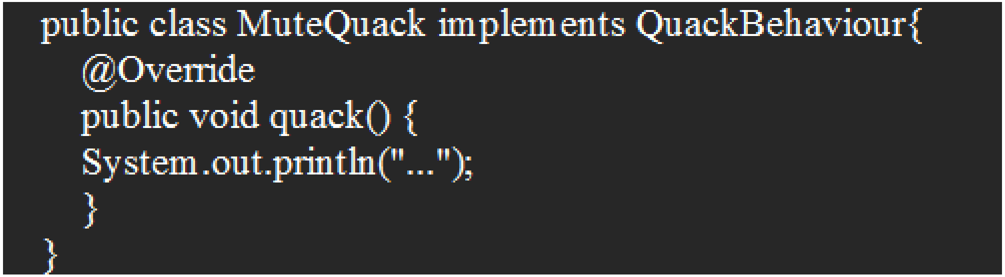
```
public class FlyWithWings implements FlyBehaviour {
    @Override
    public void fly() {
    System.out.println("I can fly with wings!");
    }
}
```



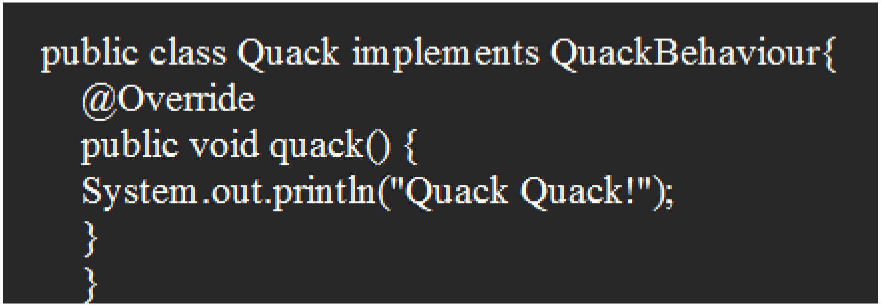
```
//Main.java
Name: Rishi Selam
Class: AIML B2
PRN: 23070126107

public class Main {
    public static void main(String[] args) {
    MallardDuck md = new MallardDuck();
    System.out.println("------------");
    md.display();
    md.performQuack();
    md.performSwim();
    md.performFly();
    System.out.println("------------");
    RedHeadDuck rd = new RedHeadDuck();
    rd.display();
    rd.performQuack();
    rd.performSwim();
    rd.performFly();
    System.out.println("------------");
    RubberDuck rubber = new RubberDuck();
    rubber.display();
    rubber.performQuack();
    rubber.performSwim();
    rubber.performFly();
    System.out.println("------------");
    DecoyDuck deco = new DecoyDuck();
    deco.display();
    deco.performQuack();
    deco.performSwim();
    deco.performFly();
    System.out.println("------------");
    }
}
```



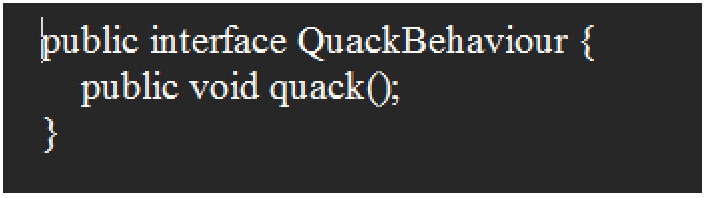
```
public class MallardDuck extends Duck{
    public MallardDuck(){
    quackBehaviour = new Quack();
    swimBehaviour = new SwimWithLegs();
    flyBehavior = new FlyWithWings();
    }
    @Override
    public void display(){
    System.out.println("I am a Mallard Duck");
    }
    }
```



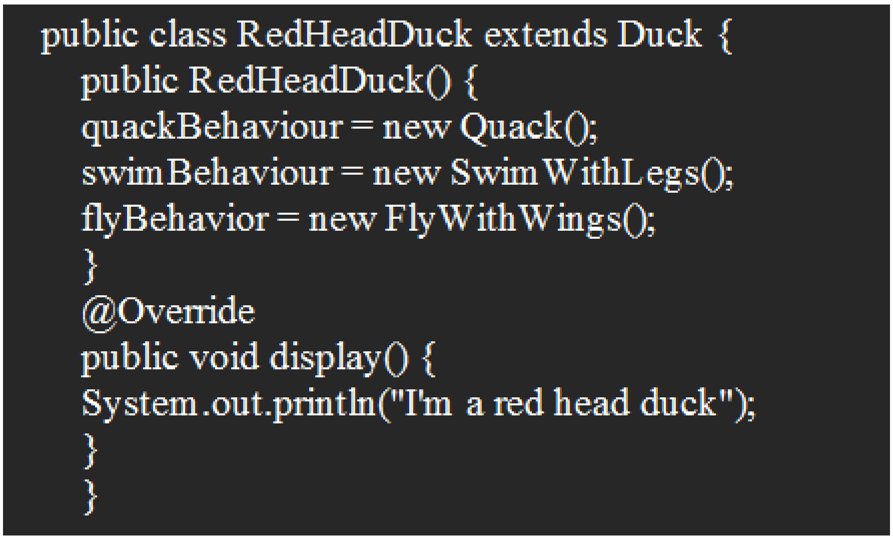
```
public class MuteQuack implements QuackBehaviour{
    @Override
    public void quack() {
    System.out.println("...");
    }
}
```



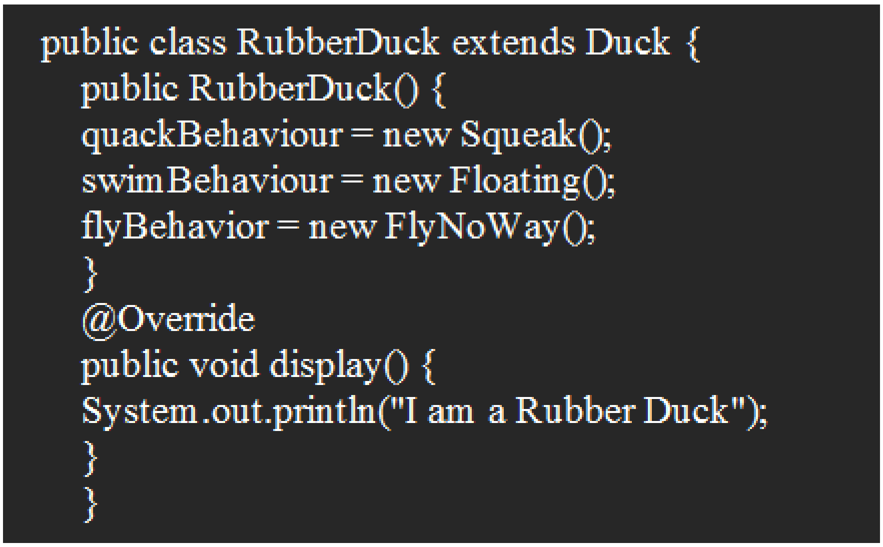
```
public class Quack implements QuackBehaviour{
    @Override
    public void quack() {
    System.out.println("Quack Quack!");
    }
    }
```



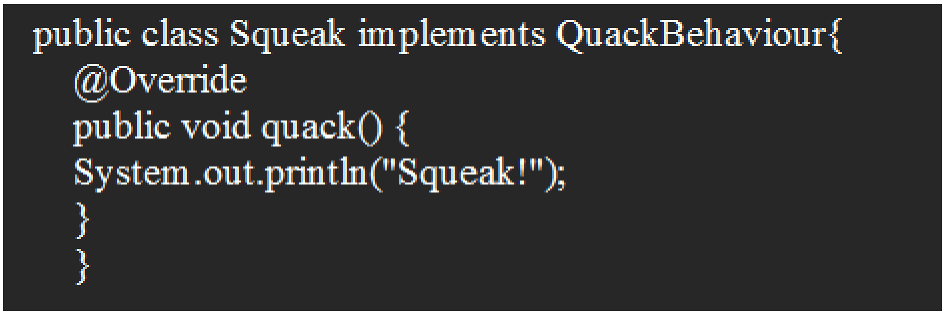
```
public interface QuackBehaviour {
    public void quack();
}
```



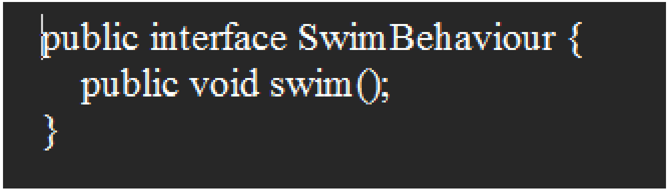
```
public class RedHeadDuck extends Duck {
    public RedHeadDuck() {
    quackBehaviour = new Quack();
    swimBehaviour = new SwimWithLegs();
    flyBehavior = new FlyWithWings();
    }
    @Override
    public void display() {
    System.out.println("I'm a red head duck");
    }
    }
```



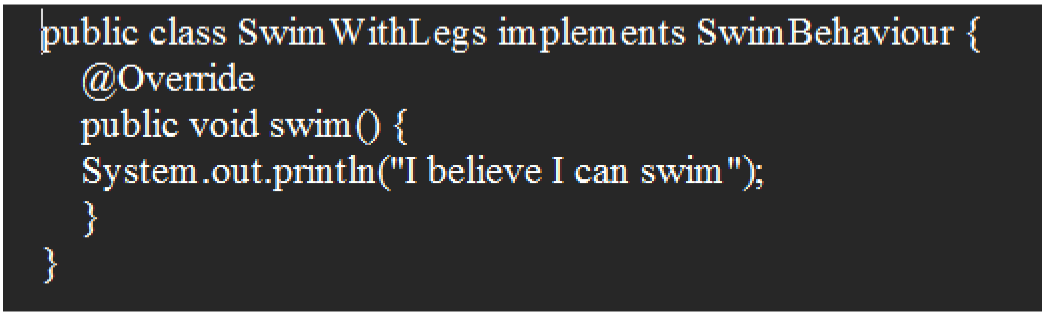
```
public class RubberDuck extends Duck {
    public RubberDuck() {
    quackBehaviour = new Squeak();
    swimBehaviour = new Floating();
    flyBehavior = new FlyNoWay();
    }
    @Override
    public void display() {
    System.out.println("I am a Rubber Duck");
    }
    }
```



```
public class Squeak implements QuackBehaviour{
    @Override
    public void quack() {
    System.out.println("Squeak!");
    }
    }
```

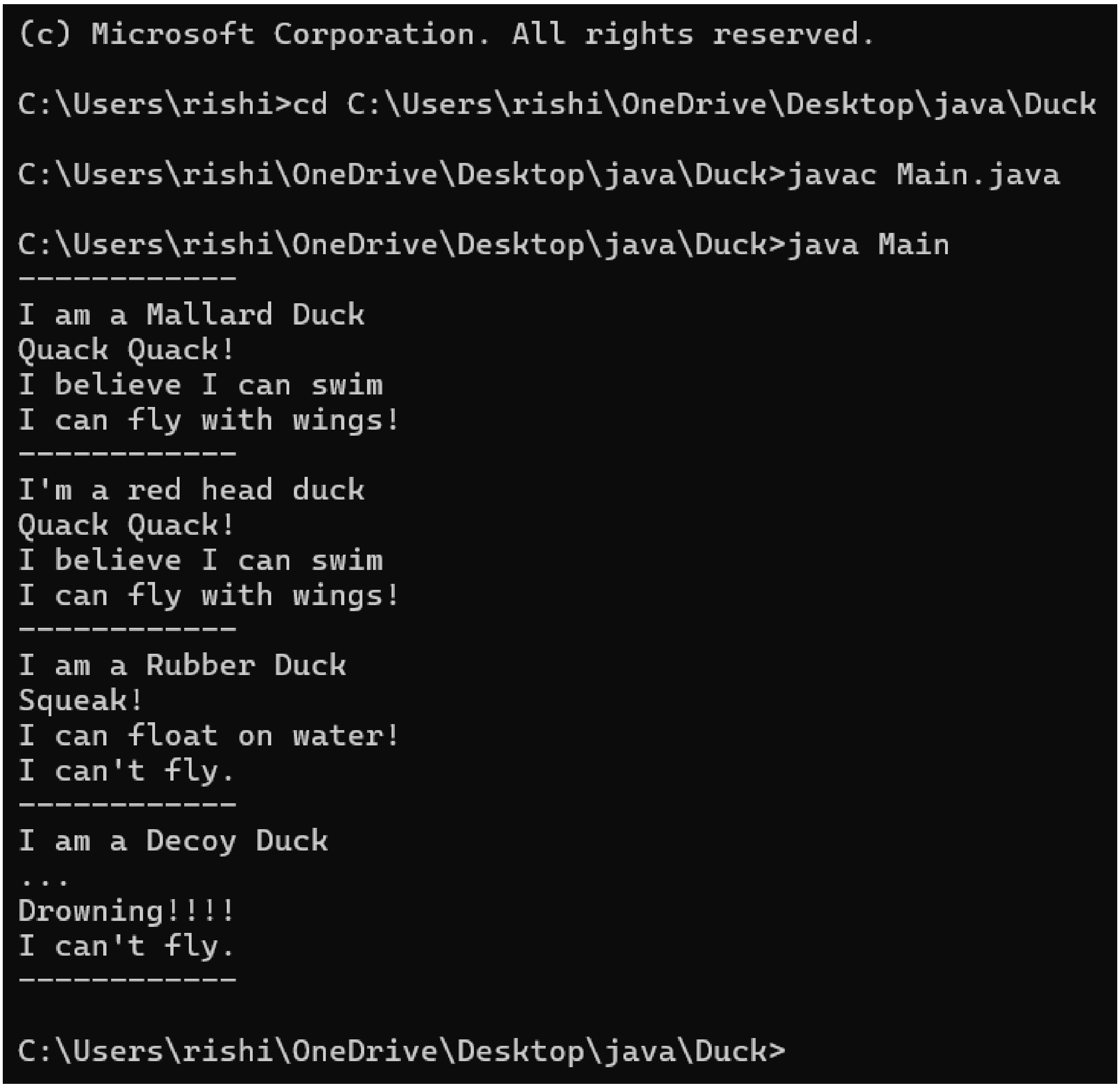


```
public interface SwimBehaviour {
    public void swim();
}
```



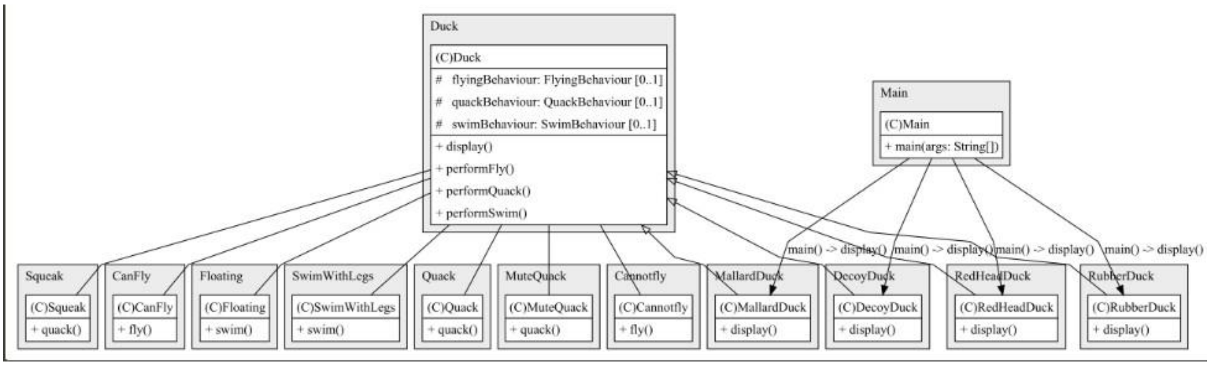
```
public class SwimWithLegs implements SwimBehaviour {
    @Override
    public void swim() {
    System.out.println("I believe I can swim");
    }
}
```

Output:



```
(c) Microsoft Corporation. All rights reserved.

C:\Users\rishi>cd C:\Users\rishi\OneDrive\Desktop\java\Duck

C:\Users\rishi\OneDrive\Desktop\java\Duck>javac Main.java

C:\Users\rishi\OneDrive\Desktop\java\Duck>java Main
------------
I am a Mallard Duck
Quack Quack!
I believe I can swim
I can fly with wings!
------------
I'm a red head duck
Quack Quack!
I believe I can swim
I can fly with wings!
------------
I am a Rubber Duck
Squeak!
I can float on water!
I can't fly.
------------
I am a Decoy Duck
...
Drowning!!!!
I can't fly.
------------

C:\Users\rishi\OneDrive\Desktop\java\Duck>
```

Class Diagram:



Github Repo Link:
https://github.com/RishiSelam/Duck_java