Name: MD. Atif Rahman Rudro

ID: IT-22002

Practice Abstraction

```
import java.io.PrintStream;

interface Machine {
    PrintStream o=System.out;
    void initialEngine();
    void finalEngine();
}

abstract class Transport {
    String trademark;
    int manufYear;
    int engineCapacity;
    public Transport(String trademark, int
            manufYear, int engineCapacity) {
        this.trademark=trademark;
        this.manufYear=manufYear;
        this.engineCapacity=engineCapacity;
    }
    abstract void propel();
```

```
public void exhibitInfo(){
    Machine.o.println("Trademark : " + trademark
        + ", Manufacture Year: " + manufYear + ",
          Engine Capacity: " +engineCapacity);
  }
}

class Car extends Transport implement Machine{
  public Car(String trademark, int manufYear,
             int engineCapacity){
      super(trademark, manufYear, engine
            Capacity);
  }
  @Override
  public void InitialEngine(){
                          engine
      o.println(" Car ~~Engine~~ Is starting...");
  }
  @Override
  public void finalEngine(){
    o.println(" Car engine is ending...");
}
@Override
public void propel(){
        o.println("Car is being driven...");
 }
}
class Bike extends Transport implements Machine{
     public Bike(String trademark, int manufYear,
                int engineCapacity){
            super(trademark, manufYear, engineCapacity);
     }
 @Override
 public void initialEngine(){
        o.println("Bike engine is starting...");
  }
@Override
public void finalEngine(){
        o.println("Bike engine is ending...");
    }
    @Override
    public void propel(){
        o.println("Bike is being rode...");
    }

    class Boat extends Transport implement Machine{
        public Boat(String trademark, int manufYear,
                    int engineCapacity){
            super(trademark, manufYear, engineCapacity);
        }
    @Override
    public void initialEngine(){
        o.println("Boat engine is starting...");
    }
    @Override
    public void finalEngine(){
        o.println("Boat engine is ending...");
    }
```

```
@ Override
    public void propel(){
        o.println("Boat is being sailed...");
    }
}

public class Abstraction{
    public static void main(String[] args){

    Machine.o.println("I am Md. Atif Rahman Rudro.
                    My ID is IT-22002");

    Machine.o.println();
                      car
    Transport ~~Car~~ = new Car("X-Corolla", 2002, 1000);
    car.exhibitInfo();
    ((Machine) car).initialEngine();
    Car.propel();
    ((Machine)car).finalEngine();
    ~~Syste~~ Machine.o.println();
Transport bike=new Bike ("Hero-Honda", 2001,
                                  150);

bike.exhibitInfo();
((Machine) bike).initialEngine();
bike.propel();
((Machine) bike).finalEngine();
Machine.o.println();

Transport boat=new Boat("Osprey", 2002, 1500);
boat.exhibitInfo();
((Machine) boat).initialEngine();
boat.propel();
((Machine) boat).finalEngine();
}
}
```

Output:

```
I am Md. Atif Rahman Rudro. My ID is: IT-22002
Trademark: X-Corolla, Manufacter Year: 2002,
                    Engine Capacity: 1000
```

```
Car engine is starting...
Car is being driven...
Car engine is ending...

Trademark: Hero-Honda, Manufacture Year: 2001,
            Engine Capacity:

Bike engine is starting...
Bike is being rode...
Bike engine is ending...

Trademark: Osprey, Manufacture Year: 2002, Engine
            Capacity: 1000

Boat engine is starting...
Boat is being sailed...
Boat engine is ending...
```