Inheritance Example: Product Inventory

For this inheritance example, I use motor trade products to create the product inventory. Therefore, Product class will act as the parent class that store the product's description and price.

```
package inheritance;

/**
 *
 * @author My-Hp
 */
class Product {
    private String description;
    private double price;

    public Product()
    {
        description = "";
        price = 0.0;
    }
    public Product(String description, double price)
    {
        this.description = description;
        this.price = price;
    }

    public String getDescription()
    {
        return description;
    }

    public double getPrice()
    {
        return price;
    }


}
```

Figure 1: Product Class

The sub product that I choose for this example is Vehicle and Motor Oil. Class MotorVehicle and MotorOil act as the subclasses of Product class and inherit the methods.

```
package inheritance;

/**
 *
 * @author My-Hp
 */
class MotorVehicle extends Product {
    private String MotorType;

    public MotorVehicle(String d, double p, String MotorType )
    {
        super(d,p);
        this.MotorType = MotorType;
    }

    public String getMotorType()
    {
        return MotorType;
    }

    public String display()
    {
        return("Description = "+ getDescription() + "\nType = "+ MotorType + "\nPrice = RM " + getPrice() + "\n");
    }
}
```

Figure 2: MotorVehicle Class

```
package inheritance;

/**
 *
 * @author My-Hp
 */
class MotorOil extends Product {
    private String type;
    private String viscosity;

    public MotorOil(String d, double p, String type, String viscosity)
    {
        super(d,p);
        this.type =type;
        this.viscosity = viscosity;
    }

    public String getType()
    {
        return type;
    }

    public String getViscosity()
    {
        return viscosity;
    }

    public String display()
    {
        return("Description = "+ getDescription() + "\nType = " + type + "\nViscosity Grade = " + viscosity +"\nPrice = RM " + getPrice()+ "\n");
    }

}
```

Figure 3: MotorOil Class

The MotorOil and MotorVehicle object created and displayed through Inheritance Class inside main() method.

```java
package inheritance;

import java.util.ArrayList;

/**
 *
 * @author My-Hp
 *
 * inheritance example for Motor Trade products
 */
public class Inheritance {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {

        ArrayList<MotorOil> oilList = new ArrayList<MotorOil>();

        MotorVehicle product1 = new MotorVehicle("Proton", 56000, "Car");

        System.out.println("====MOTOR TRADE PRODUCTS INFO====\n");
        System.out.println("--Vehicles--\n");
        System.out.println(product1.display());

        MotorOil product2 = new MotorOil("Shell",120, "Premium Conventional Oil", "10W-30");
        MotorOil product3 = new MotorOil("Shell",170, "Semi Synthetic Oil", "10W-40");

        oilList.add(product2);
        oilList.add(product3);


        System.out.println("--Motor Oils--\n");
        for(int i = 0; i<oilList.size();i++)
        {
            System.out.println((oilList.get(i)).display());
        }

    }

}
```

Figure 4: Main Class

The output shows by Figure 5 below.

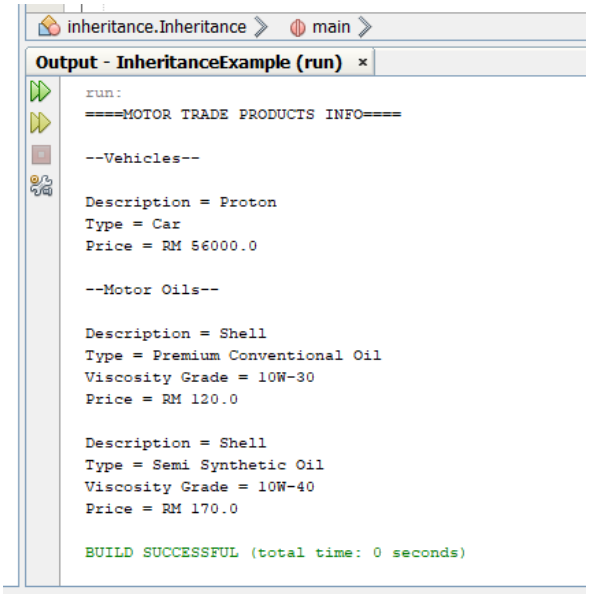

Figure 5: Output