Adapter Design Pattern :

```
class Volt {
   private int volts;
   public Volt(int v) {
      this.volts = v;
   }

   public int getVolts() {
      return volts;
   }

   public void setVolts(int volts) {
      this.volts = volts;
   }
}

class Socket {
   public Volt getVolt() {
      return new Volt(120);
   }
}

interface SocketAdapter {
   Volt get120Volt();

   Volt get12Volt();

   Volt get3Volt();
}
```

```
class SocketClassAdapter extends Socket implements SocketAdapter {

    public Volt get120Volt() {
        return getVolt();
    }

    public Volt get12Volt() {
        Volt v = getVolt();
        return convertVolt(v, 10);
    }

    public Volt get3Volt() {
        Volt v = getVolt();
        return convertVolt(v, 40);
    }

    private Volt convertVolt(Volt v, int i) {
        return new Volt(v.getVolts() / i);
    }
}

public class Main {
    public static void main(String[] args) {
        SocketAdapter socketAdapter = new SocketClassAdapter();

        Volt v120 = socketAdapter.get120Volt();
        Volt v12 = socketAdapter.get12Volt();
        Volt v3 = socketAdapter.get3Volt();

        System.out.println("The Output of 120V: " + v120.getVolts());
        System.out.println("The Output of 12V: " + v12.getVolts());
```

```
        System.out.println("The Output of 3V: " + v3.getVolts());
    }
}
```